

DevOrSysAdminScripts

Laurent Frédéric Bernard François Lyaudet

<https://lyaudet.eu/laurent/>

Laurent.Lyaudet@gmail.com

February 27, 2026

Abstract

Some “small” useful scripts for enhancing development or system-administrators tasks

Current version: 2026-02-27

Current number of commits: 413

Current git SHA1: a84a49dbdc1063155f05d697f645c75f219e86b1

Code lines: 8039 total lines, 7301 not empty lines, 738 empty lines.

Contents

1	Files tree	4
2	Listing of files	7
2.1	build_and_checks_dependencies/build_and_checks.exec.sh.tpl . . .	8
2.2	build_and_checks_dependencies/build_md_from_printable_md.exec.sh	10
2.3	build_and_checks_dependencies/check_shell_scripts_beginnings.libr.sh	13
2.4	build_and_checks_dependencies/check_shell_scripts_indentation.libr.sh	14
2.5	build_and_checks_dependencies/check_URLs.libr.sh	17
2.6	build_and_checks_dependencies/common_build_and_checks.exec.sh	19
2.7	build_and_checks_dependencies/common_build_and_checks.exec.sh.tpl	34
2.8	build_and_checks_dependencies/comparisons.libr.sh	49
2.9	build_and_checks_dependencies/create_PDF.exec.sh	53
2.10	build_and_checks_dependencies/generate_from_template.libr.sh	64
2.11	build_and_checks_dependencies/get_common_text_glob_patterns.libr.sh	66
2.12	build_and_checks_dependencies/grammar_and_spell_check.libr.sh	68
2.13	build_and_checks_dependencies/licenses_templates/build_licenses_templates.exec.sh	70

2.14	build_and_checks_dependencies/licenses_templates/license_file_header_AGPLv3+.tpl	78
2.15	build_and_checks_dependencies/licenses_templates/license_file_header_GPLv3+.tpl	79
2.16	build_and_checks_dependencies/licenses_templates/license_file_header_LGPLv3+.tpl	79
2.17	build_and_checks_dependencies/lines_counts.libr.sh	80
2.18	build_and_checks_dependencies/lines_filters.libr.sh	82
2.19	build_and_checks_dependencies/lines_maps.libr.sh	89
2.20	build_and_checks_dependencies/overwrite_if_not_equal.libr.sh	91
2.21	build_and_checks_dependencies/python_black_complement.libr.sh	93
2.22	build_and_checks_dependencies/python_isort_complement.libr.sh	94
2.23	build_and_checks_dependencies/split_score.exec.php	95
2.24	build_and_checks_dependencies/split_score.libr.php	96
2.25	build_and_checks_dependencies/strings_functions.libr.sh	98
2.26	build_and_checks_dependencies/too_long_code_lines.libr.sh	105
2.27	build_and_checks_dependencies/update_common_build_and_checks.exec.sh	108
2.28	build_and_checks_dependencies/update_or_check_files_names_listing.exec.sh	110
2.29	build_and_checks_variables/.gitignore	114
2.30	build_and_checks_variables/bandit.ini	114
2.31	build_and_checks_variables/bandit_baseline.json	114
2.32	build_and_checks_variables/black.toml	114
2.33	build_and_checks_variables/DevOrSysAdminScripts.html.tpl	114
2.34	build_and_checks_variables/DevOrSysAdminScripts.tex.tpl	119
2.35	build_and_checks_variables/files_listing.tex.tpl	123
2.36	build_and_checks_variables/files_names_listing.txt	138
2.37	build_and_checks_variables/licenses_notes.txt	140
2.38	build_and_checks_variables/phpmd_baseline.xml	141
2.39	build_and_checks_variables/pylintrc.toml	141
2.40	build_and_checks_variables/repository_data.txt	142
2.41	build_and_checks_variables/ruff.toml	144
2.42	build_and_checks_variables/shellcheck.ini	145
2.43	build_and_checks_variables/shellcheck_notes.txt	146
2.44	build_and_checks.exec.sh	147
2.45	COPYING	149
2.46	COPYING.LESSER	164
2.47	git_helpers/git_checks.libr.sh	167
2.48	git_helpers/post-commit	170
2.49	git_helpers/post_commit_renaming_comment.libr.sh	171
2.50	git_helpers/pre-commit	179
2.51	IMAP_transfert.py	180
2.52	LiensUtiles.txt	186
2.53	README.md.tpl	186
2.54	unicode.libr.php	189

2.55	upgrade_build_and_checks.exec.sh	195
2.56	wget_sha512.libr.sh	197

1 Files tree

```
. └── build_and_checks_dependencies
    ├── build_and_checks.exec.sh.tpl
    ├── build_md_from_printable_md.exec.sh
    ├── check_shell_scripts_beginnings.libr.sh
    ├── check_shell_scripts_indentation.libr.sh
    ├── check_URLs.libr.sh
    ├── common_build_and_checks.exec.sh
    ├── common_build_and_checks.exec.sh.tpl
    ├── comparisons.libr.sh
    ├── create_PDF.exec.sh
    ├── generate_from_template.libr.sh
    ├── get_common_text_glob_patterns.libr.sh
    ├── grammar_and_spell_check.libr.sh
    ├── licenses_templates
    │   ├── build_licenses_templates.exec.sh
    │   ├── license_file_header GPLv3+.tpl
    │   ├── license_file_header GPLv3+.tpl
    │   └── license_file_header GPLv3+.tpl
    ├── lines_counts.libr.sh
    ├── lines_filters.libr.sh
    ├── lines_maps.libr.sh
    ├── overwrite_if_not_equal.libr.sh
    ├── python_black_complement.libr.sh
    ├── python_isort_complement.libr.sh
    ├── split_score.exec.php
    ├── split_score.libr.php
    ├── strings_functions.libr.sh
    ├── too_long_code_lines.libr.sh
    ├── update_common_build_and_checks.exec.sh
    └── update_or_check_files_names_listing.exec.sh
    └── build_and_checks.exec.sh
    └── build_and_checks_variables
        ├── bandit_baseline.json
        ├── bandit.ini
        ├── black.toml
        ├── DevOrSysAdminScripts.html.tpl
        ├── DevOrSysAdminScripts.tex
        ├── DevOrSysAdminScripts.tex.tpl
        ├── files_listing.tex.tpl
        ├── files_names_listing.txt
        ├── .gitignore
        ├── licenses_notes.txt
        └── phpmd_baseline.xml
```

```
    └── pylintrc.toml
    └── repository_data.txt
    └── ruff.toml
    └── shellcheck.ini
    └── shellcheck_notes.txt
    └── COPYING
    └── COPYING.LESSER
    └── DevOrSysAdminScripts.html
    └── DevOrSysAdminScripts.pdf
    └── git_helpers
        ├── git_checks.libr.sh
        ├── post-commit
        ├── post_commit_renaming_comment.libr.sh
        └── pre-commit
    └── IMAP_transfert.py
    └── LiensUtiles.txt
    └── README.html
    └── README.md
    └── README.md.tpl
    └── unicode.libr.php
    └── upgrade_build_and_checks.exec.sh
    └── wget_sha512.libr.sh
```

5 directories, 61 files

```
[ 12K Feb 27 22:39] ./
└── [4.0K Feb 27 22:39] build_and_checks_dependencies/
    ├── [3.7K Feb 27 20:35] build_and_checks.exec.sh.tpl*
    ├── [3.9K Feb 23 19:39] build_md_from_printable_md.exec.sh*
    ├── [2.1K Feb 15 11:34] check_shell_scripts_beginnings.libr.sh
    ├── [4.0K Feb 24 17:34] check_shell_scripts_indentation.libr.sh
    ├── [2.9K Feb 23 17:21] check_URLs.libr.sh
    ├── [ 27K Feb 27 22:24] common_build_and_checks.exec.sh*
    ├── [ 23K Feb 27 21:22] common_build_and_checks.exec.sh.tpl
    ├── [4.7K Feb 23 17:34] comparisons.libr.sh
    ├── [ 17K Feb 27 22:23] create_PDF.exec.sh*
    ├── [3.3K Feb 19 18:48] generate_from_template.libr.sh
    ├── [1.9K Feb 19 17:18] get_common_text_glob_patterns.libr.sh
    ├── [2.2K Feb 27 19:33] grammar_and_spell_check.libr.sh
    └── [4.0K Feb 27 20:41] licenses_templates/
        ├── [ 13K Feb 27 20:41] build_licenses_templates.exec.sh*
        ├── [ 728 Dec 30 19:32] license_file_header_AGPLv3+.tpl
        ├── [ 707 Dec 27 21:48] license_file_header_GPLv3+.tpl
        └── [ 728 Dec 27 21:48] license_file_header_LGPLv3+.tpl
    └── [2.7K Feb 19 18:44] lines_counts.libr.sh
    └── [8.8K Feb 27 21:26] lines_filters.libr.sh
```

```
    └── [2.7K Feb 19 16:16]  lines_maps.libr.sh
    └── [1.8K Feb 26 16:58]  overwrite_if_not_equal.libr.sh
    └── [1.4K Jan 24 18:32]  python_black_complement.libr.sh
    └── [1.8K Feb 23 18:13]  python_isort_complement.libr.sh
    └── [1.6K Feb 3 17:57]   split_score.exec.php
    └── [1.9K Feb 3 20:11]   split_score.libr.php
    └── [ 12K Feb 26 15:37]  strings_functions.libr.sh
    └── [3.2K Feb 26 21:22]  too_long_code_lines.libr.sh
    └── [4.1K Feb 24 17:35]  update_common_build_and_checks.exec.sh*
    └── [4.8K Feb 26 22:35]  *
    └── update_or_check_files_names_listing.exec.sh**
[3.9K Feb 27 22:24]  build_and_checks.exec.sh*
[4.0K Feb 27 22:39]  build_and_checks_variables/
    ├── [ 15 Jan 5 20:00]  bandit_baseline.json
    ├── [ 25 Jan 5 18:49]  bandit.ini
    ├── [ 30 Feb 27 16:27]  black.toml
    ├── [5.3K Feb 27 22:39]  DevOrSysAdminScripts.html.tpl
    ├── [ 23K Feb 27 22:31]  DevOrSysAdminScripts.tex
    ├── [3.9K Feb 27 22:28]  DevOrSysAdminScripts.tex.tpl
    ├── [ 13K Feb 27 16:27]  files_listing.tex.tpl
    ├── [4.0K Feb 24 20:40]  files_names_listing.txt
    ├── [ 24 Jan 15 22:26]  .gitignore
    ├── [1.2K Feb 14 15:04]  licenses_notes.txt
    ├── [1001 Feb 26 15:37]  phpmd_baseline.xml
    ├── [ 678 Feb 24 22:38]  pylintrc.toml
    ├── [3.8K Feb 27 16:27]  repository_data.txt
    ├── [1.4K Feb 24 23:25]  ruff.toml
    ├── [ 167 Feb 19 18:39]  shellcheck.ini
    └── [2.1K Feb 19 19:36]  shellcheck_notes.txt
[ 34K May 6 2024]  COPYING
[7.5K Dec 27 21:48]  COPYING.LESSER
[309K Feb 27 22:31]  DevOrSysAdminScripts.html
[298K Feb 27 22:31]  DevOrSysAdminScripts.pdf
[4.0K Feb 27 20:20]  git_helpers/
    ├── [3.5K Feb 19 17:00]  git_checks.libr.sh
    ├── [1.8K Feb 27 20:20]  post-commit
    ├── [ 13K Feb 26 19:35]  post_commit_renaming_comment.libr.sh
    └── [1.4K Feb 19 18:42]  pre-commit
[8.3K Feb 27 16:27]  IMAP_transfert.py
[ 830 Dec 29 19:10]  LiensUtiles.txt
[7.2K Feb 23 16:51]  README.html
[3.5K Feb 16 17:08]  README.md
[3.6K Feb 16 16:59]  README.md.tpl
[9.1K Feb 26 15:05]  unicode.libr.php
[1.4K Feb 3 14:19]   upgrade_build_and_checks.exec.sh*
[1.4K Feb 19 14:41]  wget_sha512.libr.sh
```

5 directories, 61 files

2 Listing of files

This repository is dual licensed. All the Bash scripts source code is covered by GPLv3+. All the PHP scripts source code is covered by LGPLv3+.

The text of the licenses is available at: <https://www.gnu.org/licenses/>. The git repository of this source code is also available at: <https://github.com/LLyaudet/DevOrSysAdminScripts/>. Clone it with this URL: `git@github.com:LLyaudet/DevOrSysAdminScripts.git`.

Here is a listing of the dependencies of the code in this repository:

- The following dependencies are GPLv3+ licensed:
basename, bash (and its builtins), cat, chmod, cp, cut, date, diff, dirname, find, grep, head, mkdir, mv, realpath, rm, sed, sha512sum, shellcheck, sort, tail, wc, wget, xargs
- The following dependencies are GPLv2+ licensed:
more, pandoc, pdftex, pdflatex, tree
- The following dependencies are GPLv2 licensed:
git, pylint
- The following dependencies are LGPLv2.1+ licensed:
notify-send
- The following dependencies are MIT licensed:
black, isort, mypy, ruff
- The following dependencies are Apache-2.0 licensed:
bandit
- The following dependencies are BSD-3-Clause licensed:
phpmd
- php is licensed under PHP License v3.01
- tex is licensed under Knuth license
- pcregrep has a custom license from Cambridge University

Remark: Some commands listed under GPLv3+ are POSIX and can have different implementations under another license; for example, uutils coreutils is MIT licensed, whilst GNU coreutils is GPLv3+ licensed.

2.1 build_and_checks_dependencies/build_and_checks.exec.sh.tpl

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # The file "build_and_checks.exec.sh.tpl" was renamed from
25 # "build_and_checks.sh.tpl" to "build_and_checks.exec.sh.tpl".
26 # The file "build_and_checks.exec.sh" generated from the file
27 # "build_and_checks.sh.tpl" or "build_and_checks.exec.sh.tpl"
28 # was renamed from
29 # "build_and_checks.sh" to "build_and_checks.exec.sh".
30
31 build_and_checks(){
32     local LFBFL_working_directory="."
33     if [[ -n "$1" ]]; then
34         LFBFL_working_directory="$1"
35     fi
36     readonly LFBFL_working_directory
37
38     local LFBFL_verbose=""
39     if [[ "$*" == *--verbose* ]]; then
40         echo "$0 $*"
41         LFBFL_verbose="--verbose"
42     fi
43     readonly LFBFL_verbose
44 }
```

```

45     source ./wget_sha512.libr.sh
46
47     LFBFL_subdir="build_and_checks_dependencies"
48     mkdir -p "${LFBFL_subdir}/licenses_templates"
49     mkdir -p "build_and_checks_variables/temp"
50     if ! [[ -f "build_and_checks_variables/.gitignore" ]]; then
51         {
52             echo "temp/"
53             echo ""
54         } > "build_and_checks_variables/.gitignore"
55     fi
56
57     # LFBFL_dependencies_raw_content_download_URL
58     local LFBFL_dependencies_URL
59     LFBFL_dependencies_URL="https://raw.githubusercontent.com/LLyaudet/"
60     LFBFL_dependencies_URL+="DevOrSysAdminScripts/main/${LFBFL_subdir}"
61     readonly LFBFL_dependencies_URL
62
63     declare -r LFBFL_common_file_name="common_build_and_checks.exec.sh"
64     # LFBFL_script_download_URL
65     declare -r \
66         LFBFL_script="${LFBFL_dependencies_URL}/${LFBFL_common_file_name}"
67     declare -r \
68         LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_common_file_name}"
69     @sha512_common_build_and_checks.exec.sh@
70     wget_sha512 "${LFBFL_file_path}" "${LFBFL_script}" \
71         "${LFBFL_correct_sha512}" "${LFBFL_verbose}" \
72         chmod +x "./${LFBFL_file_path}"
73
74     "${LFBFL_file_path}" "${LFBFL_working_directory}" \
75     "${LFBFL_dependencies_URL}" "${LFBFL_verbose}"
76 }
77
78 if [[ "$*" == *--fixed_point_build* ]]; then
79     echo "--fixed_point_build"
80     source "./build_and_checks_dependencies/lines_filters.libr.sh"
81     LFBFL_working_directory=."
82     if [[ -n "$1" ]]; then
83         LFBFL_working_directory="$1"
84     fi
85     LFBFL_variables_directory="${LFBFL_working_directory}/"
86     LFBFL_variables_directory+="build_and_checks_variables"
87     LFBFL_data_file_name="${LFBFL_variables_directory}/"
88     LFBFL_data_file_name+="repository_data.txt"
89     LFBFL_repository_name=""
90     grep_variable "${LFBFL_data_file_name}" repository_name\

```

```

91     --result-variable-prefix="LFBFL_"\ \
92     --replace-line-returns-by="" \
93 # Touching the 3 following files first let us iterate
94 # upgrade_build_and_checks 2 times instead of 3 to reach
95 # a fixed point.
96 touch "${LFBFL_variables_directory}/${LFBFL_repository_name}.tex"
97 touch "${LFBFL_working_directory}/${LFBFL_repository_name}.pdf"
98 touch "${LFBFL_working_directory}/${LFBFL_repository_name}.html"
99 # The two iterations of build_and_checks needs to complete
100 # during the same minute to have a fixed point.
101 build_and_checks "$@"
102 # build_and_checks "$@" second iteration below
103 # Then calling this script without --fixed_point_build should not
104 # yield any new superficial modifications.
105 fi
106
107 build_and_checks "$@"

```

2.2 build_and_checks_dependencies/build_md_from_printable_md.exec.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "build_md_from_printable_md.sh"
25 # to "build_md_from_printable_md.exec.sh".
26

```

```

27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=overwrite_if_not_equal.libr.sh
29 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
30
31 build_md_from_printable_md() {
32     declare -i LFBFL_verbose=0
33     if [[ "$*" == *--verbose* ]]; then
34         echo "$0 $*"
35         LFBFL_verbose=1
36     fi
37     readonly LFBFL_verbose
38
39     # Remove line returns here to keep lines short.
40     local LFBFL_sed_expression
41     LFBFL_sed_expression='s/(\[[a-zA-Z0-9:-]*\]: [^\n\]*)\n/\1/Mg'
42     LFBFL_sed_expression+=';s/(<http[^\\n\\]*\\n)\\1/Mg'
43     LFBFL_sed_expression+=';s/(- <http[^\\n\\]*\\n)\\1/Mg'
44     readonly LFBFL_sed_expression
45     # Since Markdown can contain code, it is limited to Markdown URLs.
46     # Hence it is way more complicated than 's/\\n//Mg' expression
47     # in other files.
48     # Another downside is that more than one pass is needed.
49
50     declare -i LFBFL_cd_result
51     pushd .
52     if [[ -n "$1" ]];
53     then
54         if [[ LFBFL_verbose -eq 1 ]]; then
55             echo "Moving to directory: $1"
56         fi
57         cd "$1" || {
58             LFBFL_cd_result=$?
59             echo "build_md_from_printable_md no such directory"
60             # shellcheck disable=SC2248
61             return ${LFBFL_cd_result}
62         }
63     fi
64
65     local LFBFL_base_name="README"
66     if [[ -n "$2" ]];
67     then
68         if [[ LFBFL_verbose -eq 1 ]]; then
69             echo "Searching md file: $2"
70         fi
71         LFBFL_base_name="$2"
72     fi

```

```

73     readonly LFBFL_base_name
74
75     if [[ -f "${LFBFL_base_name}.md.tpl" ]];
76     then
77         sed -Ez -e "${LFBFL_sed_expression}" \
78             -e "${LFBFL_sed_expression}" \
79             -e "${LFBFL_sed_expression}" \
80             -e "${LFBFL_sed_expression}" \
81             "${LFBFL_base_name}.md.tpl" \
82             > "${LFBFL_base_name}.md.temp"
83     overwrite_if_not_equal "${LFBFL_base_name}.md" \
84             "${LFBFL_base_name}.md.temp"
85   else
86     echo "No file ${LFBFL_base_name}.md.tpl"
87   fi
88
89   pandoc --from markdown --to html --standalone \
90       --shift-heading-level-by=-1 "${LFBFL_base_name}.md" \
91       -o "${LFBFL_base_name}.html.temp"
92   # Correcting pandoc HTML "by hand".
93   # https://github.com/jgm/pandoc/issues/10957
94   # With the most open-minded answer found on Internet ; ) XD.
95   # This second sed expression is what I need,
96   # but it may be unsuitable for you.
97   # Other problems:
98   # https://github.com/jgm/pandoc/issues/11484
99   local LFBFL_prewrap="s~(code \{(\n|[^}]*)\}(\n|\})"
100  LFBFL_prewrap+="~\1      white-space: pre-wrap;\n\3~M"
101  readonly LFBFL_prewrap
102  sed -i -E -e 's~( <meta .*) />~\1>~g' \
103      -e 's~<html .*>~<html lang="en">~' \
104      -e '/ code\{white-space: pre-wrap;\}/d' \
105      "${LFBFL_base_name}.html.temp"
106  sed -i -Ez -e 's~(<img(\n|^a-z0-9)(\n|^>)* ) />~\1>~Mg' \
107      -e "${LFBFL_prewrap}" \
108      "${LFBFL_base_name}.html.temp"
109  overwrite_if_not_equal "${LFBFL_base_name}.html" \
110      "${LFBFL_base_name}.html.temp"
111
112 declare -i LFBFL_popd_result
113 popd || {
114   LFBFL_popd_result=$?
115   echo "build_md_from_printable_md no popd"
116   # shellcheck disable=SC2248
117   return ${LFBFL_popd_result}
118 }

```

```

119     # java -Xss128M -jar "../jar_venv/vnu.jar" --exit-zero-always\
120     #   --verbose ${LFBFL_base_name}.html"
121 }
122
123 build_md_from_printable_md "$@"

```

2.3 build_and_checks_dependencies/check_shell_scripts_beginnings.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "check_shell_scripts_beginnings.sh"
25 # to "check_shell_scripts_beginnings.libr.sh".
26
27 declare -gr LFBFL_SHELL_SCRIPT_BEGINNING="#!/usr/bin/env bash"
28
29 check_one_shell_script_beginning(){
30     declare -r LFBFL_file_name=$(basename "$1")
31     if [[ "${LFBFL_file_name}" =~ license_file_header_.*\\.sh ]]; then
32         return 0
33     fi
34     diff <(head -n 1 "$1") <(echo '#!/usr/bin/env bash')
35     # -----
36     # is of course slower than
37     # -----
38     # diff <(head -n 1 "$1") <(echo '#!/usr/bin/env bash')

```

```

39      # -----
40      # is of course slower than
41      #
42      declare -r LFBFL_head=$(head -n 1 "$1")
43      [[ "${LFBFL_head}" == "${LFBFL_SCRIPT_BEGINNING}" ]] \
44          || echo "$1:File has wrong shell script beginning."
45  }
46
47 check_shell_scripts_beginnings() {
48     shopt -s globstar
49     local LFBFL_file_name
50     local LFBFL_head
51     for LFBFL_file_name in **/*.sh; do
52         LFBFL_head=$(head -n 1 "${LFBFL_file_name}")
53         [[ "${LFBFL_head}" == "${LFBFL_SCRIPT_BEGINNING}" ]] || \
54             echo "${LFBFL_file_name}:File has wrong shell script beginning."
55     done
56 }
```

2.4 build_and_checks_dependencies/check_shell_scripts_indentation.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "check_shell_scripts_beginnings.sh"
25 # to "check_shell_scripts_beginnings.libr.sh".
```

```

26
27 check_one_shell_script_indentation() {
28     shopt -s lastpipe
29
30     declare -i LFBFL_file_with_error=0
31     local LFBFL_some_line
32     local LFBFL_previous_line="workeduntilsomeasshole"
33     # created a command named "workeduntilsomeasshole"
34     # and called it without neither indentation, nor parameters
35     # in a .sh script XD.
36     # Manual justification of the lines L-4 and L-3 was a serendipity.
37     # I just tried a joke and was lucky ;) :).
38     local LFBFL_line_end
39     local LFBFL_previous_line_end
40     declare -i LFBFL_line_length
41     declare -i LFBFL_line_end_length
42     declare -i LFBFL_line_start_length
43     local LFBFL_spacing
44     declare -i LFBFL_offset
45     local LFBFL_substring1
46     local LFBFL_substring2
47     declare -r LFBFL_pattern='^ *-e '
48     # This grep would have been enough without some sed commands.
49     # shellcheck disable=SC2312
50     grep -EHn -B 1 '^(\ )*' (^ |$)' "$1" \
51         | while read -r LFBFL_some_line;
52     do
53         if [[ "${LFBFL_some_line}" == "--" ]]; then
54             # When switching between code fragments, we reinit.
55             LFBFL_previous_line="workeduntilsomeasshole"
56             continue
57         fi
58         # Note how it's funny that this loop always work because we check
59         # shell scripts beginning in another script and thus we always
60         # have a line before.
61         if [[ "${LFBFL_previous_line}" == "workeduntilsomeasshole" ]]; then
62             LFBFL_previous_line="${LFBFL_some_line}"
63             continue
64         fi
65         # Remove the name of the file and then the line number
66         LFBFL_line_end=${LFBFL_some_line#*:}
67         LFBFL_line_end=${LFBFL_line_end#*:}
68         # But for the previous line '-' or ':' is used instead of ':'
69         # and filenames with '-' are much more frequent than with ':...'
70         LFBFL_line_length=${#LFBFL_some_line}
71

```

```

72     LFBFL_line_end_length=${#LFBFL_line_end}
73     LFBFL_line_start_length=$(((
74         LFBFL_line_length-LFBFL_line_end_length
75     )))
76     LFBFL_offset=$((LFBFL_line_start_length-2))
77     # Checking if increasing number of digits in the line numbers
78     # (99 -> 100).
79     LFBFL_substring1=${LFBFL_previous_line:${LFBFL_offset}:1}
80     # Nice puzzle below :) Look for comments above :)
81     case ${LFBFL_substring1} in
82         -) ((LFBFL_offset+=1));;
83         :) ((LFBFL_offset+=1));;
84         [0-9]) ((LFBFL_offset+=2));;
85         *) echo ':-(';;
86     esac
87     LFBFL_previous_line_end=${LFBFL_previous_line:${LFBFL_offset}}
88     if [[ "${LFBFL_line_end}" =~ ${LFBFL_pattern} ]]; then
89         # Line starts with spacing and -e
90         # Keep max spaces
91         LFBFL_spacing=${LFBFL_line_end%%-*}
92         LFBFL_offset=${#LFBFL_spacing}
93         LFBFL_substring1=${LFBFL_line_end:${LFBFL_offset}:3}
94         LFBFL_substring2=${LFBFL_previous_line_end:${LFBFL_offset}:3}
95         if [[ "${LFBFL_substring1}" != "${LFBFL_substring2}" ]]; then
96             echo "${LFBFL_some_line}"
97             LFBFL_file_with_error=1
98         fi
99     else
100         echo "${LFBFL_some_line}"
101         LFBFL_file_with_error=1
102     fi
103     LFBFL_previous_line="${LFBFL_some_line}"
104 done
105
106 if [[ LFBFL_file_with_error -eq 1 ]]; then
107     echo "$1:File has some lines with odd number of spaces."
108 fi
109 }
110
111 check_shell_scripts_indentation() {
112     shopt -s globstar
113     local LFBFL_file_name
114     for LFBFL_file_name in **/*.sh; do
115         check_one_shell_script_indentation "${LFBFL_file_name}"
116     done
117 }

```

2.5 build_and_checks_dependencies/check_URLs.libr.sh

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "check_URLs.sh" to "check_URLs.libr.sh".
25 #
26 LFBFL_subdir="build_and_checks_dependencies"
27 # shellcheck source=get_common_text_glob_patterns.libr.sh
28 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
29 # shellcheck source=lines_filters.libr.sh
30 source "./${LFBFL_subdir}/lines_filters.libr.sh"
31 #
32 check_URLs() {
33     declare -i LFBFL_verbose=0
34     if [[ "$*" == *--verbose* ]]; then
35         echo "$0 $*"
36         LFBFL_verbose=1
37     fi
38     readonly LFBFL_verbose
39     if [[ ! -o pipefail ]]; then
40         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
41         set -o pipefail
42         trap 'set +o pipefail' RETURN
43     fi
44 }
```

```

45
46     get_COMMON_TEXT_FILES_GLOB_PATTERNS
47
48     declare -Ar LFBFL_substitutions=
49         ["http://www.gnu.org/licenses/"]="https://www.gnu.org/licenses/"
50     )
51
52     local LFBFL_pattern
53     local LFBFL_file_name
54     local LFBFL_base_file_name
55     local LFBFL_substitution
56     local LFBFL_substitution2
57     for LFBFL_pattern in "${COMMON_TEXT_FILES_GLOB_PATTERNS[@]}"; do
58         [[ LFBFL_verbose -eq 0 ]]\
59             || echo "Iterating on pattern: ${LFBFL_pattern}"
60         find . -type f -name "${LFBFL_pattern}" -printf '%P\n'\
61             | xargs grep -H 'http:'\
62             | grep -v '| xargs grep -H 'htt"\''\
63             | grep -vP "[\'\"]http(:[^\'\"]*)[\'\"].*[\'\"]https\\1[\'\"]"
64     # Last grep just above will remove false positives from
65     # substitutions that fit on one line.
66     find . -type f -name "${LFBFL_pattern}" -printf '%P\n'\
67         | relevant_find\
68         | not_main_html_find\
69         | while read -r LFBFL_file_name;
70     do
71         [[ -f "${LFBFL_file_name}" ]] || continue
72         LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
73         [[ "${LFBFL_base_file_name}" != "check_URLs.libr.sh" ]]\
74             || continue
75         [[ LFBFL_verbose -eq 0 ]]\
76             || echo "Handling the file: ${LFBFL_file_name}"
77         for LFBFL_substitution in "${!LFBFL_substitutions[@]}"; do
78             LFBFL_substitution2=\
79             ${LFBFL_substitutions[$LFBFL_substitution]}
80             if grep -q "${LFBFL_substitution}" "${LFBFL_file_name}"; then
81                 sed -i "s|${LFBFL_substitution}|${LFBFL_substitution2}|g"\'
82                 "${LFBFL_file_name}"
83             fi
84         done
85     done
86 done
87 }
```

2.6 build_and_checks_dependencies/common_build_and_checks.exec.sh

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # The file "common_build_and_checks.exec.sh.tpl" was renamed from
25 # "common_build_and_checks.sh.tpl" to
26 # "common_build_and_checks.exec.sh.tpl".
27 # The file "common_build_and_checks.exec.sh" generated from the file
28 # "common_build_and_checks.sh.tpl"
29 # or "common_build_and_checks.exec.sh.tpl"
30 # was renamed from
31 # "common_build_and_checks.sh" to "common_build_and_checks.exec.sh".
32 #
33 common_build_and_checks() {
34     # $1 LFBFL_working_directory
35     # $2 LFBFL_dependencies_URL
36     # $3 optional --verbose
37     declare -r LFBFL_working_directory="$1"
38     # declare -r LFBFL_dependencies_URL="$2" too long
39     declare -r LFBFL_start_URL="$2"
40     local LFBFL_verbose=""
41     if [[ "$*" == *--verbose* ]]; then
42         echo "$0 $*"
43         LFBFL_verbose="--verbose"
44     fi
```

```

45   readonly LFBFL_verbose
46
47   if [[ ! -o pipefail ]]; then
48     [[ "${LFBFL_verbose}" == "--verbose" ]] \
49       && echo "pipefail option activated"
50     set -o pipefail
51     trap 'set +o pipefail' RETURN
52   fi
53
54   source ./wget_sha512.libr.sh
55
56   LFBFL_subdir="build_and_checks_dependencies"
57   local LFBFL_file_name
58   local LFBFL_URL
59   local LFBFL_file_path
60
61   LFBFL_file_name="build_md_from_printable_md.exec.sh"
62   # LFBFL_script_download_URL
63   LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
64   LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
65   local LFBFL_correct_sha512
66   LFBFL_correct_sha512='5e6560b6f4d0792ab065e5ee5746560d68aa8e0609ff'
67   LFBFL_correct_sha512+=`953fabd5e4bc740c972956375b043f1556dee03aa49a'
68   LFBFL_correct_sha512+=`3a6edf10bd6e2586141baf435ce35d78df2213f'
69   wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
70     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
71   chmod +x "./${LFBFL_file_path}"
72
73   LFBFL_file_name="check_shell_scripts_beginnings.libr.sh"
74   LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
75   LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
76   local LFBFL_correct_sha512
77   LFBFL_correct_sha512='e0cdfc2d653daf9f15a3447f40e41fe03a332ac4054c4'
78   LFBFL_correct_sha512+=`0950efc0b9f58912791068c081babf7a7a9e72bacd57'
79   LFBFL_correct_sha512+=`8e600bbb5e0c51466e60e91e5b6dda6173b088b'
80   wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
81     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
82
83   LFBFL_file_name="check_shell_scripts_indentation.libr.sh"
84   LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
85   LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
86   local LFBFL_correct_sha512
87   LFBFL_correct_sha512='d473359eaef8164f8e7ab85fcdb8812b3ce46e2731ef96'
88   LFBFL_correct_sha512+=`ff46df693890131abf064c9f2b1beacbabd50aad778f'
89   LFBFL_correct_sha512+=`6d3b9b669bc0469f868f1ae69660fd4b89990f1'
90   wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \

```

```

91      "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
92
93  LFBFL_file_name="check_URLs.libr.sh"
94  LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
95  LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
96  local LFBFL_correct_sha512
97  LFBFL_correct_sha512='bd3d085710ce3ee75034daea6286adc21b0309cf3004a'
98  LFBFL_correct_sha512+='41b65af1878bed263c392dd8159716996420abb03cb8'
99  LFBFL_correct_sha512+='34d9b91c8859bdf8455c5b9d85d9b8d80fec062'
100 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
101     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
102
103 LFBFL_file_name="comparisons.libr.sh"
104 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
105 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
106 local LFBFL_correct_sha512
107 LFBFL_correct_sha512='1abcc547d6baf661c339489967108fd744321419e8e6d'
108 LFBFL_correct_sha512+='2a25896280a23243106dc72be30a4f400caf6789e9be'
109 LFBFL_correct_sha512+='740223b9e12d6155bb931aa94bb946455c8c5c'
110 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
111     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
112
113 LFBFL_file_name="create_PDF.exec.sh"
114 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
115 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
116 local LFBFL_correct_sha512
117 LFBFL_correct_sha512='485cdee460a381180f9a1ab5ca8ebf3c5d25a3660f99f'
118 LFBFL_correct_sha512+='7edd8854ea85e43b111afb3d4d44d683744632a774d9'
119 LFBFL_correct_sha512+='f92317744d72fed37389627c3496b688eec12dc'
120 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
121     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
122 chmod +x "./${LFBFL_file_path}"
123
124 LFBFL_file_name="generate_from_template.libr.sh"
125 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
126 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
127 local LFBFL_correct_sha512
128 LFBFL_correct_sha512='5779c3d2c64d0bb403525c410e37b0b06fcf1ef97d503'
129 LFBFL_correct_sha512+='21fd27345545a8ebea020b2d3d809ac5c1bc66f9cebe'
130 LFBFL_correct_sha512+='a8e8ea63aa539eaa179cf09a3860239c96df8e9'
131 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
132     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
133
134 LFBFL_file_name="get_common_text_glob_patterns.libr.sh"
135 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
136 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"

```

```

137 local LFBFL_correct_sha512
138 LFBFL_correct_sha512='4e0915efcc9a4088b1f79c9c7efbf7da5d68e8e1b21ff'
139 LFBFL_correct_sha512+='feb4746c9d468be80affa30babf35938e15f9161db3e'
140 LFBFL_correct_sha512+='3f6d95c913b86638563d7bd2995dc58a26f8e3a'
141 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
142     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
143
144 LFBFL_file_name="grammar_and_spell_check.libr.sh"
145 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
146 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
147 local LFBFL_correct_sha512
148 LFBFL_correct_sha512='a47f723b8e10a924c131f3779d3985758cb9933816f5f'
149 LFBFL_correct_sha512+='9b37b4518d6ab94453b71cd67adcd4659fe4757207de'
150 LFBFL_correct_sha512+='055c1aca88bd16dbc034497a36a6b5eaec4e55d'
151 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
152     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
153
154 # /licenses_templates/ -----
155 declare -r LFBFL_subdir2="${LFBFL_subdir}/licenses_templates"
156 declare -r LFBFL_start_URL2="${LFBFL_start_URL}/licenses_templates"
157 LFBFL_file_name="build_licenses_templates.exec.sh"
158 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
159 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
160 local LFBFL_correct_sha512
161 LFBFL_correct_sha512='5e8f15180c1d5661daa8ef4a77800a835a99cb02eaelf'
162 LFBFL_correct_sha512+='edee9f223a8433f9309ed7bbc353a0429813ddad0153'
163 LFBFL_correct_sha512+='d0b7d39b4e877514c9996d65d7b5fb24117a032'
164 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
165     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
166 chmod +x "./${LFBFL_file_path}"
167
168 LFBFL_file_name="license_file_header GPLv3+.tpl"
169 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
170 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
171 local LFBFL_correct_sha512
172 LFBFL_correct_sha512='9f9aa56d9b90d5003a12c5ec3370ac31aab74fc4a2524'
173 LFBFL_correct_sha512+='2c5fbcb0197b0abb8242f62dafc76eaf60faa5da173'
174 LFBFL_correct_sha512+='05924b31b64b037083532fcfd9f98445000e2d0'
175 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
176     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
177
178 LFBFL_file_name="license_file_header GPLv3+.tpl"
179 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
180 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
181 local LFBFL_correct_sha512
182 LFBFL_correct_sha512='2bdef9a729a5a9ca5ed87ef8a518877c5ee6b31b45336'

```

```

183 LFBFL_correct_sha512+='08660316a4e595fb31465a4847f4d53a2f7264118bd4'
184 LFBFL_correct_sha512+='eab7c1d1892e00bc1861455ed02898f194db067'
185 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
186     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
187
188 LFBFL_file_name="license_file_header_LGPLv3+.tpl"
189 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
190 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
191 local LFBFL_correct_sha512
192 LFBFL_correct_sha512='265fd0d086b48ab798f35a072335a4976e4eda8c256fe'
193 LFBFL_correct_sha512+='bba6c77c98ca86cdd844c6ac150c3445c80debbb3fc2'
194 LFBFL_correct_sha512+='574b5fceeeaf79ec734a759bdef5fe798067c62'
195 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
196     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
197 # -----
198
199 LFBFL_file_name="lines_counts.libr.sh"
200 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
201 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
202 local LFBFL_correct_sha512
203 LFBFL_correct_sha512='d8c22af747f1e255b773edbf705030d0064f711dd5d9'
204 LFBFL_correct_sha512+='4d8cb51049310f0fad2495103dfb7cd200500ebcc85d'
205 LFBFL_correct_sha512+='884f087a2d54fe9a47bc5c5e58b68963ee432a8'
206 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
207     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
208
209 LFBFL_file_name="lines_filters.libr.sh"
210 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
211 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
212 local LFBFL_correct_sha512
213 LFBFL_correct_sha512='1a5a7a4903168a61135b4cbd1d6ac65c65a93b6873130'
214 LFBFL_correct_sha512+='8115edfb0ff4dfe64321c7b6b1fe15a1a32aff292736'
215 LFBFL_correct_sha512+='a74645974ca01f0162e5e9f753910281e3576f0'
216 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
217     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
218
219 LFBFL_file_name="lines_maps.libr.sh"
220 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
221 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
222 local LFBFL_correct_sha512
223 LFBFL_correct_sha512='25c3e5082951260295b5f0e87ba1f2c978efdea1e8926'
224 LFBFL_correct_sha512+='072ce4d2062dc2a47bb92b9098f081e553eeae5d9154'
225 LFBFL_correct_sha512+='fdc50c9d88ae5365b776a23eb6e026b884a8374'
226 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
227     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
228

```

```

229 LFBFL_file_name="overwrite_if_not_equal.libr.sh"
230 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
231 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
232 local LFBFL_correct_sha512
233 LFBFL_correct_sha512='2781a564d264bae84c787321481dc83ae43a1ce324ee6'
234 LFBFL_correct_sha512+='d5ceb73d5a98607fad5fe9ff8d03979b898f7cd7c712'
235 LFBFL_correct_sha512+='8215d26e76874d56aa67fb63fa9de776617d24a'
236 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
237     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
238
239 LFBFL_file_name="python_black_complement.libr.sh"
240 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
241 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
242 local LFBFL_correct_sha512
243 LFBFL_correct_sha512='e28c6e0cfffe6b8eebef26bddfb440f0303d0dcc3987'
244 LFBFL_correct_sha512+='6d498d95b9d589e0800b8381124936e5103b372f509c'
245 LFBFL_correct_sha512+='aeb6213e909710a2ec1e345e076a44f0006fdbb'
246 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
247     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
248
249 LFBFL_file_name="python_isort_complement.libr.sh"
250 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
251 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
252 local LFBFL_correct_sha512
253 LFBFL_correct_sha512='97a9fa46b975d81c68f910d5a7d1efcd476c1a4be6570'
254 LFBFL_correct_sha512+='37066f5178cd5761d1079d86bd266a83000c37149327'
255 LFBFL_correct_sha512+='3b8d7647da5dea4c962ea3a4c3643cb4e07b778'
256 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
257     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
258
259 LFBFL_file_name="split_score.exec.php"
260 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
261 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
262 local LFBFL_correct_sha512
263 LFBFL_correct_sha512='2f7ff107a43716109b1acbd7d89870d05577af00a981a'
264 LFBFL_correct_sha512+='05f162a054c96387fb99544f8225c333b1212f74326c'
265 LFBFL_correct_sha512+='f9ec6196e796f6257dfbba9347ac8c57b4f09db'
266 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
267     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
268
269 LFBFL_file_name="split_score.libr.php"
270 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
271 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
272 local LFBFL_correct_sha512
273 LFBFL_correct_sha512='1413898a4fb4f31ca17993fd6110f7e1f32adf31c22c4'
274 LFBFL_correct_sha512+='7fa46a31d3c1511a3112a611e04fac19bb838d4476d9'

```

```

275 LFBFL_correct_sha512+='ecf77a28dd48d3d2732df7392c50e38262852d7'
276 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
277     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
278
279 LFBFL_file_name="strings_functions.libr.sh"
280 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
281 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
282 local LFBFL_correct_sha512
283 LFBFL_correct_sha512='f29cce1e27d2b9483716326938b09b4e9cffdc23f5a39'
284 LFBFL_correct_sha512+='1516dd6b1514903b3484b38e9c585c85ff389e5d90a0'
285 LFBFL_correct_sha512+='2ffda748385aac01a078dc8aae06f78c08579b6'
286 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
287     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
288
289 LFBFL_file_name="too_long_code_lines.libr.sh"
290 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
291 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
292 local LFBFL_correct_sha512
293 LFBFL_correct_sha512='433858315341224e32d3e90c6f0c611ba17f88b3d3cd7'
294 LFBFL_correct_sha512+='53296c1294c4b838e8b59031e5395bdd16ad898a9030'
295 LFBFL_correct_sha512+='4c89b03779abcf7f68e418b832eb56505f92cf'
296 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
297     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
298
299 LFBFL_file_name="update_or_check_files_names_listing.exec.sh"
300 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
301 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
302 local LFBFL_correct_sha512
303 LFBFL_correct_sha512='bf330ba727703e9763a04bd7a05f7ad037531f75f420b'
304 LFBFL_correct_sha512+='90595d4d8b886310170bfb2f6c423b8d7fb5ab3b5394'
305 LFBFL_correct_sha512+='413b3b17d2afcde53019c0ce25730716e77aba8'
306 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
307     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
308 chmod +x "./${LFBFL_file_path}"
309
310 shopt -s globstar
311 # shellcheck source=check_shell_scripts_beginnings.libr.sh
312 source "./${LFBFL_subdir}/check_shell_scripts_beginnings.libr.sh"
313 # shellcheck source=check_shell_scripts_indentation.libr.sh
314 source "./${LFBFL_subdir}/check_shell_scripts_indentation.libr.sh"
315 # shellcheck source=check_URLs.libr.sh
316 source "./${LFBFL_subdir}/check_URLs.libr.sh"
317 # shellcheck source=comparisons.libr.sh
318 source "./${LFBFL_subdir}/comparisons.libr.sh"
319 # shellcheck source=generate_from_template.libr.sh
320 source "./${LFBFL_subdir}/generate_from_template.libr.sh"

```

```

321 # shellcheck source=get_common_text_glob_patterns.libr.sh
322 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
323 # shellcheck source=grammar_and_spell_check.libr.sh
324 source "./${LFBFL_subdir}/grammar_and_spell_check.libr.sh"
325 # shellcheck source=lines_counts.libr.sh
326 source "./${LFBFL_subdir}/lines_counts.libr.sh"
327 # shellcheck source=lines_filters.libr.sh
328 source "./${LFBFL_subdir}/lines_filters.libr.sh"
329 # shellcheck source=lines_maps.libr.sh
330 source "./${LFBFL_subdir}/lines_maps.libr.sh"
331 # shellcheck source=overwrite_if_not_equal.libr.sh
332 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
333 # shellcheck source=python_black_complement.libr.sh
334 source "./${LFBFL_subdir}/python_black_complement.libr.sh"
335 # shellcheck source=python_isort_complement.libr.sh
336 source "./${LFBFL_subdir}/python_isort_complement.libr.sh"
337 # shellcheck source=strings_functions.libr.sh
338 source "./${LFBFL_subdir}/strings_functions.libr.sh"
339 # shellcheck source=too_long_code_lines.libr.sh
340 source "./${LFBFL_subdir}/too_long_code_lines.libr.sh"

341
342 local LFBFL_data_file_name=\
343 "build_and_checks_variables/repository_data.txt"
344
345 declare -i LFBFL_max_line_length
346 grep_variable "${LFBFL_data_file_name}" max_line_length\
347 --result-variable-prefix="LFBFL_"
348
349 local LFBFL_update_max_length="s/^line-length = [0-9]*$"
350 LFBFL_update_max_length+="/line-length = ${LFBFL_max_line_length}/"
351 sed -E -e "${LFBFL_update_max_length}" \
352 build_and_checks_variables/black.toml\
353 > build_and_checks_variables/black.toml.temp
354 overwrite_if_not_equal build_and_checks_variables/black.toml\
355 build_and_checks_variables/black.toml.temp
356
357 declare -i LFBFL_upgrade_venvs=0
358 declare -r LFBFL_upgrade_venvs_ts_file=\
359 "build_and_checks_variables/upgrade_venvs_ts"
360 declare -i LFBFL_upgrade_venvs_ts
361 declare -i LFBFL_current_ts
362 local LFBFL_upgrade_venvs_answer
363
364 local upgrade_venvs_time_interval_in_seconds=""
365 grep_variable "${LFBFL_data_file_name}" \
366 upgrade_venvs_time_interval_in_seconds

```

```

367 case ${upgrade_venvs_time_interval_in_seconds} in
368     wat) upgrade_venvs_time_interval_in_seconds=${RANDOM};;
369     watyouwant?) upgrade_venvs_time_interval_in_seconds=${SRANDOM};;
370     *) echo "No wat for you?";; #TempsDeCerveauDisponible XD SC2249 ;;
371 esac
372
373 if [[ -f "${LFBFL_upgrade_venvs_ts_file}" ]]; then
374     LFBFL_upgrade_venvs_ts=$(
375         stat -c %Y "${LFBFL_upgrade_venvs_ts_file}"
376     )
377     LFBFL_current_ts=$((date +%s)
378     ((LFBFL_current_ts--=LFBFL_upgrade_venvs_ts))
379     ((LFBFL_current_ts--=upgrade_venvs_time_interval_in_seconds)))
380     if [[ LFBFL_current_ts -gt 0 ]]; then
381         LFBFL_upgrade_venvs=1
382     fi
383 else
384     LFBFL_upgrade_venvs=1
385 fi
386 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
387     local LFBFL2_upgrade_venvs=""
388     grep_variable "${LFBFL_data_file_name}" upgrade_venvs \
389     --result-variable-prefix="LFBFL2_"
390     if [[ "${LFBFL2_upgrade_venvs}" != "auto" ]]; then
391         read -r -n 1 -t 10 \
392             -p "Upgrade venvs and composer global? [Y/n]" \
393             LFBFL_upgrade_venvs_answer
394         if [[ "${LFBFL_upgrade_venvs_answer}" == "n" ]]; then
395             LFBFL_upgrade_venvs=0
396         fi
397     fi
398 fi
399
400 echo "Building license headers"
401 "./${LFBFL_subdir2}/build_licenses_templates.exec.sh" \
402     "${LFBFL_working_directory}" "${LFBFL_verbose}"
403
404 echo "Building README.md"
405 "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
406     "${LFBFL_working_directory}" "README" "${LFBFL_verbose}"
407
408 echo "Building other MarkDown files"
409 local LFBFL_some_directory
410 declare -r LFBFL_readme="${LFBFL_working_directory}/README.md.tpl"
411 find "${LFBFL_working_directory}" -name "*.md.tpl" \
412     | relevant_find\

```

```

413     | while read -r LFBFL_file_path;
414 do
415     [[ "${LFBFL_file_path}" == "${LFBFL_readme}" ]] && continue
416     echo "Found template ${LFBFL_file_path}"
417     LFBFL_some_directory=$(dirname "${LFBFL_file_path}")
418     LFBFL_file_name=$(basename "${LFBFL_file_path}")
419     LFBFL_file_name=${LFBFL_file_name%.md.tpl}
420     "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
421         "${LFBFL_some_directory}" "${LFBFL_file_name}" \
422         "${LFBFL_verbose}"
423 done
424
425 declare -i LFBFL_cd_result
426 pushd .
427 cd "${LFBFL_working_directory}" || {
428     LFBFL_cd_result=$?
429     echo "common_build_and_checks.exec.sh no such directory"
430     # shellcheck disable=SC2248
431     return ${LFBFL_cd_result}
432 }
433
434 echo "Running shellcheck"
435 find . -name "*.sh" \
436     | relevant_find \
437     | while read -r LFBFL_file_path;
438 do
439     shellcheck --rcfile=build_and_checks_variables/shellcheck.ini \
440         "${LFBFL_file_path}"
441 done
442
443 echo "---Python---"
444 echo "Running isort"
445 local LFBFL_isort_venv=""
446 grep_variable "${LFBFL_data_file_name}" isort_venv \
447     --result-variable-prefix="LFBFL_" \
448     --replace-line-returns-by=""
449 if [[ -n "${LFBFL_isort_venv}" ]]; then
450     if command -v deactivate
451     then
452         deactivate
453     fi
454     # shellcheck disable=SC1090,SC1091
455     source "${LFBFL_isort_venv}/bin/activate"
456 fi
457 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
458     pip install --upgrade isort

```

```

459     fi
460     isort .
461     if [[ -n "${LFBFL_isort_venv}" ]]; then
462         deactivate
463     fi
464     python_isort_complement
465
466     echo "Running black"
467     local LFBFL_black_venv=""
468     grep_variable "${LFBFL_data_file_name}" black_venv\
469         --result-variable-prefix="LFBFL_"\
470         --replace-line-returns-by=""
471     if [[ -n "${LFBFL_black_venv}" ]]; then
472         if command -v deactivate
473             then
474                 deactivate
475             fi
476             # shellcheck disable=SC1090,SC1091
477             source "${LFBFL_black_venv}/bin/activate"
478         fi
479     if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
480         pip install --upgrade black
481     fi
482     black --config build_and_checks_variables/black.toml .
483     if [[ -n "${LFBFL_black_venv}" ]]; then
484         deactivate
485     fi
486     python_black_complement
487
488     local LFBFL_mypy_venv=""
489     grep_variable "${LFBFL_data_file_name}" mypy_venv\
490         --result-variable-prefix="LFBFL_"\
491         --replace-line-returns-by=""
492     if [[ -n "${LFBFL_mypy_venv}" ]]; then
493         if command -v deactivate
494             then
495                 deactivate
496             fi
497             # shellcheck disable=SC1090,SC1091
498             source "${LFBFL_mypy_venv}/bin/activate"
499         fi
500     if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
501         pip install --upgrade mypy
502     fi
503     shopt -s lastpipe
504     local LFBFL_directory_path

```

```

505 declare -i LFBFL_no_toml=1
506 find . -name "pyproject.toml" \
507   | relevant_find \
508   | while read -r LFBFL_file_path;
509 do
510   if grep -q "Typing :: Typed" "${LFBFL_file_path}"; then
511     echo "Running mypy"
512     LFBFL_directory_path=$(dirname "${LFBFL_file_path}")
513     mypy "${LFBFL_directory_path}"
514   fi
515   LFBFL_no_toml=0
516 done
517 if [[ LFBFL_no_toml -eq 1 ]]; then
518   echo "Running mypy"
519   mypy .
520 fi
521 if [[ -n "${LFBFL_mypy_venv}" ]]; then
522   deactivate
523 fi
524
525 echo "Running bandit"
526 LFBFL_bandit_venv=""
527 grep_variable "${LFBFL_data_file_name}" bandit_venv \
528   --result-variable-prefix="LFBFL_" \
529   --replace-line-returns-by=""
530 if [[ -n "${LFBFL_bandit_venv}" ]]; then
531   if command -v deactivate
532   then
533     deactivate
534   fi
535   # shellcheck disable=SC1090,SC1091
536   source "${LFBFL_bandit_venv}/bin/activate"
537 fi
538 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
539   pip install --upgrade bandit
540 fi
541 bandit --ini build_and_checks_variables/bandit.ini \
542   -b build_and_checks_variables/bandit_baseline.json \
543   -r .
544 # Saving new baseline in temp if necessary.
545 bandit --ini build_and_checks_variables/bandit.ini \
546   -f json -o build_and_checks_variables/temp/bandit_baseline.json \
547   -r .
548 if [[ -n "${LFBFL_bandit_venv}" ]]; then
549   deactivate
550 fi

```

```

551
552     echo "Running pylint"
553     LFBFL_pylint_venv=""
554     grep_variable "${LFBFL_data_file_name}" pylint_venv\
555         --result-variable-prefix="LFBFL_"\
556         --replace-line-returns-by=""
557     if [[ -n "${LFBFL_pylint_venv}" ]]; then
558         if command -v deactivate
559             then
560                 deactivate
561             fi
562             # shellcheck disable=SC1090,SC1091
563             source "${LFBFL_pylint_venv}/bin/activate"
564         fi
565         if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
566             pip install --upgrade pylint
567         fi
568         pylint --rcfile build_and_checks_variables/pylintrc.toml\
569             --recursive=y .
570         if [[ -n "${LFBFL_pylint_venv}" ]]; then
571             deactivate
572         fi
573
574     echo "Running ruff"
575     LFBFL_ruff_venv=""
576     grep_variable "${LFBFL_data_file_name}" ruff_venv\
577         --result-variable-prefix="LFBFL_"\
578         --replace-line-returns-by=""
579     if [[ -n "${LFBFL_ruff_venv}" ]]; then
580         if command -v deactivate
581             then
582                 deactivate
583             fi
584             # shellcheck disable=SC1090,SC1091
585             source "${LFBFL_ruff_venv}/bin/activate"
586         fi
587         if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
588             pip install --upgrade ruff
589         fi
590         ruff check --config build_and_checks_variables/ruff.toml
591         if [[ -n "${LFBFL_ruff_venv}" ]]; then
592             deactivate
593         fi
594         echo "---Python end---"
595
596     echo "---PHP---"

```

```

597 echo "Running PHPMD"
598 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
599   composer global require phpmd/phpmd
600 fi
601
602 local LFBFL_phpmd_baseline="build_and_checks_variables/"
603 LFBFL_phpmd_baseline+="phpmd_baseline.xml"
604 readonly LFBFL_phpmd_baseline
605
606 local LFBFL_temp_phpmd_baseline="build_and_checks_variables/temp/"
607 LFBFL_temp_phpmd_baseline+="phpmd_baseline.xml"
608 readonly LFBFL_temp_phpmd_baseline
609
610 local LFBFL_phpmd_rulesets="cleancode,codesize,controversial,"
611 LFBFL_phpmd_rulesets+="design,naming,unusedcode"
612 readonly LFBFL_phpmd_rulesets
613
614 phpmd --color --baseline-file "${LFBFL_phpmd_baseline}" \
615   . text "${LFBFL_phpmd_rulesets}"
616 # Saving new baseline in temp if necessary.
617 rm "${LFBFL_temp_phpmd_baseline}"
618 phpmd --color --generate-baseline \
619   --baseline-file "${LFBFL_temp_phpmd_baseline}" \
620   . text "${LFBFL_phpmd_rulesets}"
621 echo "---PHP end---"
622
623 echo "----JS----"
624 LFBFL_npm_lint_directories=""
625 grep_variable "${LFBFL_data_file_name}" npm_lint_directories \
626   --result-variable-prefix="LFBFL_"
627 if [[ -n "${LFBFL_npm_lint_directories}" ]]; then
628   echo "Running ESLint"
629   local LFBFL_JS_directory
630   echo "${LFBFL_npm_lint_directories}" \
631     | sed -e 's/\n/\n/g' \
632     | while read -r LFBFL_JS_directory;
633   do
634     (cd "${LFBFL_JS_directory}" && npm run lint)
635   done
636 fi
637 echo "----JS end---"
638
639 echo "Analyzing too long lines"
640 # shellcheck disable=SC2248
641 too_long_code_lines ${LFBFL_max_line_length} "$@"
642

```

```

643 echo "Analyzing shell scripts beginnings"
644 check_shell_scripts_beginnings\
645     | relevant_grep
646
647 echo "Analyzing shell scripts indentation"
648 check_shell_scripts_indentation\
649     | relevant_grep
650
651 echo "Analyzing URLs"
652 check_URLs "$@" \
653     | relevant_grep
654
655 echo "Analyzing strange characters: hover over in doubt"
656 # shellcheck disable=SC1111
657 LFBFL_usual_characters="\x00-\x7Fàâéèëïïôûç©“”└─|«»"
658 grep --exclude-dir .git --binary-files=without-match --color=always\
659     -nPr "[^${LFBFL_usual_characters}]" .
660
661 echo "Checking listed files"
662 "./${LFBFL_subdir}/update_or_check_files_names_listing.exec.sh"
663
664 echo "Creating the PDF file of the listing of the source code"
665 "./${LFBFL_subdir}/create_PDF.exec.sh" "${LFBFL_verbose}"
666
667 if [[ -f "build_and_checks_variables/post_build.sh" ]]; then
668     ./build_and_checks_variables/post_build.sh
669 fi
670
671 # -----
672 # It is at the end that all the HTML files are generated and that
673 # we can check there are no mistakes.
674 echo "Running Nu W3C HTML CSS and SVG validator"
675 local upgrade_vnu_jar_time_interval_in_seconds=""
676 grep_variable "${LFBFL_data_file_name}" \
677     upgrade_vnu_jar_time_interval_in_seconds
678 if [[ "${upgrade_vnu_jar_time_interval_in_seconds}" == "wat" ]];
679 then
680     upgrade_vnu_jar_time_interval_in_seconds=${RANDOM}
681 fi
682
683 declare -i LFBFL_vnu_jar_ts
684 local LFBFL_vnu_jar_path=""
685 grep_variable "${LFBFL_data_file_name}" vnu_jar_path\
686     --result-variable-prefix="LFBFL_"\ \
687     --replace-line-returns-by=""
688 if [[ -f "${LFBFL_vnu_jar_path}" ]]; then

```

```

689 LFBFL_vnu_jar_ts=$(stat -c %Y "${LFBFL_vnu_jar_path}")
690 LFBFL_current_ts=$(date +%s)
691 ((LFBFL_current_ts==LFBFL_vnu_jar_ts))
692 ((LFBFL_current_ts==upgrade_vnu_jar_time_interval_in_seconds))
693 if [[ LFBFL_current_ts -gt 0 ]]; then
694     echo $'!/\\ATTENTION: Nu W3C validator is too old./!\\\''
695     echo "Go get the latest version here:"
696     echo "https://github.com/validator/validator/releases"
697 fi
698 declare -r LFBFL_files_for_Nu=$((
699     find . -iregex ".*\.(css|html|svg|xhtml)\" \
700     | relevant_find
701 ))
702 declare -a LFBFL_files_for_Nu2
703 mapfile -t LFBFL_files_for_Nu2 <<< "${LFBFL_files_for_Nu}"
704 readonly LFBFL_files_for_Nu2
705 java -Xss128M -jar "${LFBFL_vnu_jar_path}" --exit-zero-always\
706     --verbose "${LFBFL_files_for_Nu2[@]}@"
707 fi
708 # -----
709 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
710     touch "${LFBFL_upgrade_venvs_ts_file}"
711 fi
712
713 declare -i LFBFL_popd_result
714 popd || {
715     LFBFL_popd_result=$?
716     echo "common_build_and_checks.exec.sh no popd"
717     # shellcheck disable=SC2248
718     return ${LFBFL_popd_result}
719 }
720 }
721 }
722
723 common_build_and_checks "$@"

```

2.7 build_and_checks_dependencies/common_build_and_checks.exec.sh.tpl

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,

```

```

9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24 # The file "common_build_and_checks.exec.sh.tpl" was renamed from
25 # "common_build_and_checks.sh.tpl" to
26 # "common_build_and_checks.exec.sh.tpl".
27 # The file "common_build_and_checks.exec.sh" generated from the file
28 # "common_build_and_checks.sh.tpl"
29 # or "common_build_and_checks.exec.sh.tpl"
30 # was renamed from
31 # "common_build_and_checks.sh" to "common_build_and_checks.exec.sh".
32
33 common_build_and_checks() {
34     # $1 LFBFL_working_directory
35     # $2 LFBFL_dependencies_URL
36     # $3 optional --verbose
37     declare -r LFBFL_working_directory="$1"
38     # declare -r LFBFL_dependencies_URL="$2" too long
39     declare -r LFBFL_start_URL="$2"
40     local LFBFL_verbose=""
41     if [[ "$*" == *--verbose* ]]; then
42         echo "$0 $*"
43         LFBFL_verbose="--verbose"
44     fi
45     readonly LFBFL_verbose
46
47     if [[ ! -o pipefail ]]; then
48         [[ "${LFBFL_verbose}" == "--verbose" ]] \
49             && echo "pipefail option activated"
50         set -o pipefail
51         trap 'set +o pipefail' RETURN
52     fi
53
54     source ./wget_sha512.libr.sh

```

```

55
56     LFBFL_subdir="build_and_checks_dependencies"
57     local LFBFL_file_name
58     local LFBFL_URL
59     local LFBFL_file_path
60
61     LFBFL_file_name="build_md_from_printable_md.exec.sh"
62     # LFBFL_script_download_URL
63     LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
64     LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
65     @sha512_build_md_from_printable_md.exec.sh@
66     wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
67         "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
68     chmod +x "./${LFBFL_file_path}"
69
70     LFBFL_file_name="check_shell_scripts_beginnings.libr.sh"
71     LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
72     LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
73     @sha512_check_shell_scripts_beginnings.libr.sh@
74     wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
75         "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
76
77     LFBFL_file_name="check_shell_scripts_indentation.libr.sh"
78     LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
79     LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
80     @sha512_check_shell_scripts_indentation.libr.sh@
81     wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
82         "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
83
84     LFBFL_file_name="check_URLs.libr.sh"
85     LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
86     LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
87     @sha512_check_URLs.libr.sh@
88     wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
89         "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
90
91     LFBFL_file_name="comparisons.libr.sh"
92     LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
93     LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
94     @sha512_comparisons.libr.sh@
95     wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
96         "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
97
98     LFBFL_file_name="create_PDF.exec.sh"
99     LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
100    LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"

```

```

101 @sha512_create_PDF.exec.sh@
102 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
103   "${LFBFL_correct_sha512}" "${LFBFL_verbose}" \
104 chmod +x "./${LFBFL_file_path}"
105
106 LFBFL_file_name="generate_from_template.libr.sh"
107 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
108 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
109 @sha512_generate_from_template.libr.sh@
110 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
111   "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
112
113 LFBFL_file_name="get_common_text_glob_patterns.libr.sh"
114 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
115 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
116 @sha512_get_common_text_glob_patterns.libr.sh@
117 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
118   "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
119
120 LFBFL_file_name="grammar_and_spell_check.libr.sh"
121 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
122 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
123 @sha512_grammar_and_spell_check.libr.sh@
124 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
125   "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
126
127 # /licenses_templates/ -----
128 declare -r LFBFL_subdir2="${LFBFL_subdir}/licenses_templates"
129 declare -r LFBFL_start_URL2="${LFBFL_start_URL}/licenses_templates"
130 LFBFL_file_name="build_licenses_templates.exec.sh"
131 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
132 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
133 @sha512_build_licenses_templates.exec.sh@
134 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
135   "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
136 chmod +x "./${LFBFL_file_path}"
137
138 LFBFL_file_name="license_file_header GPLv3+.tpl"
139 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
140 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
141 @sha512_license_file_header_GPLv3+.tpl@
142 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
143   "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
144
145 LFBFL_file_name="license_file_header GPLv3+.tpl"
146 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"

```

```

147 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}""
148 @sha512_license_file_header_GPLv3+.tpl@
149 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
150     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
151
152 LFBFL_file_name="license_file_header_LGPLv3+.tpl"
153 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
154 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}""
155 @sha512_license_file_header_LGPLv3+.tpl@
156 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
157     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
158 # -----
159
160 LFBFL_file_name="lines_counts.libr.sh"
161 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
162 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}""
163 @sha512_lines_counts.libr.sh@
164 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
165     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
166
167 LFBFL_file_name="lines_filters.libr.sh"
168 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
169 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}""
170 @sha512_lines_filters.libr.sh@
171 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
172     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
173
174 LFBFL_file_name="lines_maps.libr.sh"
175 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
176 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}""
177 @sha512_lines_maps.libr.sh@
178 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
179     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
180
181 LFBFL_file_name="overwrite_if_not_equal.libr.sh"
182 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
183 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}""
184 @sha512_overwrite_if_not_equal.libr.sh@
185 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
186     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
187
188 LFBFL_file_name="python_black_complement.libr.sh"
189 LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
190 LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}""
191 @sha512_python_black_complement.libr.sh@
192 wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \

```

```

193      "${LFBFL_correct_sha512}" "${LFBFL_verbose}""
194
195      LFBFL_file_name="python_isort_complement.libr.sh"
196      LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
197      LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
198      @sha512_python_isort_complement.libr.sh@
199      wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
200          "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
201
202      LFBFL_file_name="split_score.exec.php"
203      LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
204      LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
205      @sha512_split_score.exec.php@
206      wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
207          "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
208
209      LFBFL_file_name="split_score.libr.php"
210      LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
211      LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
212      @sha512_split_score.libr.php@
213      wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
214          "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
215
216      LFBFL_file_name="strings_functions.libr.sh"
217      LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
218      LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
219      @sha512_strings_functions.libr.sh@
220      wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
221          "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
222
223      LFBFL_file_name="too_long_code_lines.libr.sh"
224      LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
225      LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
226      @sha512_too_long_code_lines.libr.sh@
227      wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
228          "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
229
230      LFBFL_file_name="update_or_check_files_names_listing.exec.sh"
231      LFBFL_URL="${LFBFL_start_URL}/${LFBFL_file_name}"
232      LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_file_name}"
233      @sha512_update_or_check_files_names_listing.exec.sh@
234      wget_sha512 "${LFBFL_file_path}" "${LFBFL_URL}" \
235          "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
236      chmod +x "./${LFBFL_file_path}"
237
238      shopt -s globstar

```

```

239 # shellcheck source=check_shell_scripts_beginnings.libr.sh
240 source "./${LFBFL_subdir}/check_shell_scripts_beginnings.libr.sh"
241 # shellcheck source=check_shell_scripts_indentation.libr.sh
242 source "./${LFBFL_subdir}/check_shell_scripts_indentation.libr.sh"
243 # shellcheck source=check_URLs.libr.sh
244 source "./${LFBFL_subdir}/check_URLs.libr.sh"
245 # shellcheck source=comparisons.libr.sh
246 source "./${LFBFL_subdir}/comparisons.libr.sh"
247 # shellcheck source=generate_from_template.libr.sh
248 source "./${LFBFL_subdir}/generate_from_template.libr.sh"
249 # shellcheck source=get_common_text_glob_patterns.libr.sh
250 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
251 # shellcheck source=grammar_and_spell_check.libr.sh
252 source "./${LFBFL_subdir}/grammar_and_spell_check.libr.sh"
253 # shellcheck source=lines_counts.libr.sh
254 source "./${LFBFL_subdir}/lines_counts.libr.sh"
255 # shellcheck source=lines_filters.libr.sh
256 source "./${LFBFL_subdir}/lines_filters.libr.sh"
257 # shellcheck source=lines_maps.libr.sh
258 source "./${LFBFL_subdir}/lines_maps.libr.sh"
259 # shellcheck source=overwrite_if_not_equal.libr.sh
260 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
261 # shellcheck source=python_black_complement.libr.sh
262 source "./${LFBFL_subdir}/python_black_complement.libr.sh"
263 # shellcheck source=python_isort_complement.libr.sh
264 source "./${LFBFL_subdir}/python_isort_complement.libr.sh"
265 # shellcheck source=strings_functions.libr.sh
266 source "./${LFBFL_subdir}/strings_functions.libr.sh"
267 # shellcheck source=too_long_code_lines.libr.sh
268 source "./${LFBFL_subdir}/too_long_code_lines.libr.sh"
269
270 local LFBFL_data_file_name=\
271 "build_and_checks_variables/repository_data.txt"
272
273 declare -i LFBFL_max_line_length
274 grep_variable "${LFBFL_data_file_name}" max_line_length\
275 --result-variable-prefix="LFBFL_"
276
277 local LFBFL_update_max_length="s/^line-length = [0-9]*$"
278 LFBFL_update_max_length+="/line-length = ${LFBFL_max_line_length}/"
279 sed -E -e "${LFBFL_update_max_length}" \
280     build_and_checks_variables/black.toml\
281     > build_and_checks_variables/black.toml.temp
282 overwrite_if_not_equal build_and_checks_variables/black.toml\
283     build_and_checks_variables/black.toml.temp
284

```

```

285 declare -i LFBFL_upgrade_venvs=0
286 declare -r LFBFL_upgrade_venvs_ts_file=\"
287 "build_and_checks_variables/upgrade_venvs_ts"
288 declare -i LFBFL_upgrade_venvs_ts
289 declare -i LFBFL_current_ts
290 local LFBFL_upgrade_venvs_answer
291
292 local upgrade_venvs_time_interval_in_seconds=""
293 grep_variable "${LFBFL_data_file_name}" \
294     upgrade_venvs_time_interval_in_seconds
295 case ${upgrade_venvs_time_interval_in_seconds} in
296     wat) upgrade_venvs_time_interval_in_seconds=${RANDOM};;
297     watyouwant?) upgrade_venvs_time_interval_in_seconds=${SRANDOM};;
298     *) echo "No wat for you?";; #TempsDeCerveauDisponible XD SC2249 ;;
299 esac
300
301 if [[ -f "${LFBFL_upgrade_venvs_ts_file}" ]]; then
302     LFBFL_upgrade_venvs_ts=$(
303         stat -c %Y "${LFBFL_upgrade_venvs_ts_file}"
304     )
305     LFBFL_current_ts=$(date +%s)
306     ((LFBFL_current_ts--=LFBFL_upgrade_venvs_ts))
307     ((LFBFL_current_ts--=upgrade_venvs_time_interval_in_seconds))
308     if [[ LFBFL_current_ts -gt 0 ]]; then
309         LFBFL_upgrade_venvs=1
310     fi
311 else
312     LFBFL_upgrade_venvs=1
313 fi
314 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
315     local LFBFL2_upgrade_venvs=""
316     grep_variable "${LFBFL_data_file_name}" upgrade_venvs \
317         --result-variable-prefix="LFBFL2_"
318     if [[ "${LFBFL2_upgrade_venvs}" != "auto" ]]; then
319         read -r -n 1 -t 10 \
320             -p "Upgrade venvs and composer global? [Y/n]" \
321             LFBFL_upgrade_venvs_answer
322         if [[ "${LFBFL_upgrade_venvs_answer}" == "n" ]]; then
323             LFBFL_upgrade_venvs=0
324         fi
325     fi
326 fi
327
328 echo "Building license headers"
329 "./${LFBFL_subdir2}/build_licenses_templates.exec.sh" \
330     "${LFBFL_working_directory}" "${LFBFL_verbose}"

```

```

331
332 echo "Building README.md"
333 "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
334     "${LFBFL_working_directory}" "README" "${LFBFL_verbose}"
335
336 echo "Building other MarkDown files"
337 local LFBFL_some_directory
338 declare -r LFBFL_readme="${LFBFL_working_directory}/README.md.tpl"
339 find "${LFBFL_working_directory}" -name "*.md.tpl" \
340     | relevant_find \
341     | while read -r LFBFL_file_path;
342 do
343     [[ "${LFBFL_file_path}" == "${LFBFL_readme}" ]] && continue
344     echo "Found template ${LFBFL_file_path}"
345     LFBFL_some_directory=$(dirname "${LFBFL_file_path}")
346     LFBFL_file_name=$(basename "${LFBFL_file_path}")
347     LFBFL_file_name=${LFBFL_file_name%.md.tpl}
348     "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
349         "${LFBFL_some_directory}" "${LFBFL_file_name}" \
350         "${LFBFL_verbose}"
351 done
352
353 declare -i LFBFL_cd_result
354 pushd .
355 cd "${LFBFL_working_directory}" || {
356     LFBFL_cd_result=$?
357     echo "common_build_and_checks.exec.sh no such directory"
358     # shellcheck disable=SC2248
359     return ${LFBFL_cd_result}
360 }
361
362 echo "Running shellcheck"
363 find . -name "*.sh" \
364     | relevant_find \
365     | while read -r LFBFL_file_path;
366 do
367     shellcheck --rcfile=build_and_checks_variables/shellcheck.ini \
368         "${LFBFL_file_path}"
369 done
370
371 echo "---Python---"
372 echo "Running isort"
373 local LFBFL_isort_venv=""
374 grep_variable "${LFBFL_data_file_name}" isort_venv \
375     --result-variable-prefix="LFBFL_" \
376     --replace-line-returns-by=""

```

```

377 if [[ -n "${LFBFL_isort_venv}" ]]; then
378     if command -v deactivate
379     then
380         deactivate
381     fi
382     # shellcheck disable=SC1090,SC1091
383     source "${LFBFL_isort_venv}/bin/activate"
384 fi
385 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
386     pip install --upgrade isort
387 fi
388 isort .
389 if [[ -n "${LFBFL_isort_venv}" ]]; then
390     deactivate
391 fi
392 python_isort_complement
393
394 echo "Running black"
395 local LFBFL_black_venv=""
396 grep_variable "${LFBFL_data_file_name}" black_venv\
397     --result-variable-prefix="LFBFL_"\
398     --replace-line-returns-by=""
399 if [[ -n "${LFBFL_black_venv}" ]]; then
400     if command -v deactivate
401     then
402         deactivate
403     fi
404     # shellcheck disable=SC1090,SC1091
405     source "${LFBFL_black_venv}/bin/activate"
406 fi
407 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
408     pip install --upgrade black
409 fi
410 black --config build_and_checks_variables/black.toml .
411 if [[ -n "${LFBFL_black_venv}" ]]; then
412     deactivate
413 fi
414 python_black_complement
415
416 local LFBFL_mypy_venv=""
417 grep_variable "${LFBFL_data_file_name}" mypy_venv\
418     --result-variable-prefix="LFBFL_"\
419     --replace-line-returns-by=""
420 if [[ -n "${LFBFL_mypy_venv}" ]]; then
421     if command -v deactivate
422     then

```

```

423         deactivate
424     fi
425     # shellcheck disable=SC1090,SC1091
426     source "${LFBFL_mypy_venv}/bin/activate"
427   fi
428   if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
429     pip install --upgrade mypy
430   fi
431   shopt -s lastpipe
432   local LFBFL_directory_path
433   declare -i LFBFL_no_toml=1
434   find . -name "pyproject.toml" \
435     | relevant_find \
436     | while read -r LFBFL_file_path;
437   do
438     if grep -q "Typing :: Typed" "${LFBFL_file_path}"; then
439       echo "Running mypy"
440       LFBFL_directory_path=$(dirname "${LFBFL_file_path}")
441       mypy "${LFBFL_directory_path}"
442     fi
443     LFBFL_no_toml=0
444   done
445   if [[ LFBFL_no_toml -eq 1 ]]; then
446     echo "Running mypy"
447     mypy .
448   fi
449   if [[ -n "${LFBFL_mypy_venv}" ]]; then
450     deactivate
451   fi
452
453   echo "Running bandit"
454   LFBFL_bandit_venv=""
455   grep_variable "${LFBFL_data_file_name}" bandit_venv \
456     --result-variable-prefix="LFBFL_" \
457     --replace-line-returns-by=""
458   if [[ -n "${LFBFL_bandit_venv}" ]]; then
459     if command -v deactivate
460     then
461       deactivate
462     fi
463     # shellcheck disable=SC1090,SC1091
464     source "${LFBFL_bandit_venv}/bin/activate"
465   fi
466   if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
467     pip install --upgrade bandit
468   fi

```

```

469 bandit --ini build_and_checks_variables/bandit.ini \
470   -b build_and_checks_variables/bandit_baseline.json \
471   -r .
472 # Saving new baseline in temp if necessary.
473 bandit --ini build_and_checks_variables/bandit.ini \
474   -f json -o build_and_checks_variables/temp/bandit_baseline.json \
475   -r .
476 if [[ -n "${LFBFL_bandit_venv}" ]]; then
477   deactivate
478 fi
479
480 echo "Running pylint"
481 LFBFL_pylint_venv=""
482 grep_variable "${LFBFL_data_file_name}" pylint_venv \
483   --result-variable-prefix="LFBFL_" \
484   --replace-line-returns-by=""
485 if [[ -n "${LFBFL_pylint_venv}" ]]; then
486   if command -v deactivate
487     then
488       deactivate
489     fi
490   # shellcheck disable=SC1090,SC1091
491   source "${LFBFL_pylint_venv}/bin/activate"
492 fi
493 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
494   pip install --upgrade pylint
495 fi
496 pylint --rcfile build_and_checks_variables/pylintrc.toml \
497   --recursive=y .
498 if [[ -n "${LFBFL_pylint_venv}" ]]; then
499   deactivate
500 fi
501
502 echo "Running ruff"
503 LFBFL_ruff_venv=""
504 grep_variable "${LFBFL_data_file_name}" ruff_venv \
505   --result-variable-prefix="LFBFL_" \
506   --replace-line-returns-by=""
507 if [[ -n "${LFBFL_ruff_venv}" ]]; then
508   if command -v deactivate
509     then
510       deactivate
511     fi
512   # shellcheck disable=SC1090,SC1091
513   source "${LFBFL_ruff_venv}/bin/activate"
514 fi

```

```

515 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
516     pip install --upgrade ruff
517 fi
518 ruff check --config build_and_checks_variables/ruff.toml
519 if [[ -n "${LFBFL_ruff_venv}" ]]; then
520     deactivate
521 fi
522 echo "---Python end---"
523
524 echo "---PHP---"
525 echo "Running PHPMD"
526 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
527     composer global require phpmd/phpmd
528 fi
529
530 local LFBFL_phpmd_baseline="build_and_checks_variables/"
531 LFBFL_phpmd_baseline+="phpmd_baseline.xml"
532 readonly LFBFL_phpmd_baseline
533
534 local LFBFL_temp_phpmd_baseline="build_and_checks_variables/temp/"
535 LFBFL_temp_phpmd_baseline+="phpmd_baseline.xml"
536 readonly LFBFL_temp_phpmd_baseline
537
538 local LFBFL_phpmd_rulesets="cleancode,codesize,controversial,"
539 LFBFL_phpmd_rulesets+="design,naming,unusedcode"
540 readonly LFBFL_phpmd_rulesets
541
542 phpmd --color --baseline-file "${LFBFL_phpmd_baseline}" \
543     . text "${LFBFL_phpmd_rulesets}"
544 # Saving new baseline in temp if necessary.
545 rm "${LFBFL_temp_phpmd_baseline}"
546 phpmd --color --generate-baseline \
547     --baseline-file "${LFBFL_temp_phpmd_baseline}" \
548     . text "${LFBFL_phpmd_rulesets}"
549 echo "---PHP end---"
550
551 echo "---JS---"
552 LFBFL_npm_lint_directories=""
553 grep_variable "${LFBFL_data_file_name}" npm_lint_directories\
554     --result-variable-prefix="LFBFL_"
555 if [[ -n "${LFBFL_npm_lint_directories}" ]]; then
556     echo "Running ESLint"
557     local LFBFL_JS_directory
558     echo "${LFBFL_npm_lint_directories}"\
559         | sed -e 's/\n/g'\
560         | while read -r LFBFL_JS_directory;

```

```

561     do
562         (cd "${LFBFL_JS_directory}" && npm run lint)
563     done
564 fi
565 echo "---JS end---"
566
567 echo "Analyzing too long lines"
568 # shellcheck disable=SC2248
569 too_long_code_lines ${LFBFL_max_line_length} "$@"
570
571 echo "Analyzing shell scripts beginnings"
572 check_shell_scripts_beginnings\
573     | relevant_grep
574
575 echo "Analyzing shell scripts indentation"
576 check_shell_scripts_indentation\
577     | relevant_grep
578
579 echo "Analyzing URLs"
580 check_URLs "$@" \
581     | relevant_grep
582
583 echo "Analyzing strange characters: hover over in doubt"
584 # shellcheck disable=SC1111
585 LFBFL_usual_characters="\x00-\x7Fàâéèëïïôûç©“”└┘|«»"
586 grep --exclude-dir .git --binary-files=without-match --color=always\
587     -nPr "[^${LFBFL_usual_characters}]" .
588
589 echo "Checking listed files"
590 "./${LFBFL_subdir}/update_or_check_files_names_listing.exec.sh"
591
592 echo "Creating the PDF file of the listing of the source code"
593 "./${LFBFL_subdir}/create_PDF.exec.sh" "${LFBFL_verbose}"
594
595 if [[ -f "build_and_checks_variables/post_build.sh" ]]; then
596     ./build_and_checks_variables/post_build.sh
597 fi
598
599 # -----
600 # It is at the end that all the HTML files are generated and that
601 # we can check there are no mistakes.
602 echo "Running Nu W3C HTML CSS and SVG validator"
603 local upgrade_vnu_jar_time_interval_in_seconds=""
604 grep_variable "${LFBFL_data_file_name}" \
605     upgrade_vnu_jar_time_interval_in_seconds
606 if [[ "${upgrade_vnu_jar_time_interval_in_seconds}" == "wat" ]];
```

```

607     then
608         upgrade_vnu_jar_time_interval_in_seconds=${RANDOM}
609     fi
610
611     declare -i LFBFL_vnu_jar_ts
612     local LFBFL_vnu_jar_path=""
613     grep_variable "${LFBFL_data_file_name}" vnu_jar_path\
614         --result-variable-prefix="LFBFL_"\
615         --replace-line-returns-by=""
616     if [[ -f "${LFBFL_vnu_jar_path}" ]]; then
617         LFBFL_vnu_jar_ts=$(stat -c %Y "${LFBFL_vnu_jar_path}")
618         LFBFL_current_ts=$(date +s)
619         ((LFBFL_current_ts-=LFBFL_vnu_jar_ts))
620         ((LFBFL_current_ts-=upgrade_vnu_jar_time_interval_in_seconds))
621         if [[ LFBFL_current_ts -gt 0 ]]; then
622             echo $'!/\\ATTENTION: Nu W3C validator is too old./!\\\''
623             echo "Go get the latest version here:"
624             echo "https://github.com/validator/validator/releases"
625         fi
626         declare -r LFBFL_files_for_Nu=$(
627             find . -iregex ".*\.(css|html|svg|xhtml)" \
628                 | relevant_find
629         )
630         declare -a LFBFL_files_for_Nu2
631         mapfile -t LFBFL_files_for_Nu2 <<< "${LFBFL_files_for_Nu}"
632         readonly LFBFL_files_for_Nu2
633         java -Xss128M -jar "${LFBFL_vnu_jar_path}" --exit-zero-always\
634             --verbose "${LFBFL_files_for_Nu2[@]}"
635     fi
636     # -----
637
638     if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
639         touch "${LFBFL_upgrade_venvs_ts_file}"
640     fi
641
642     declare -i LFBFL_popd_result
643     popd || {
644         LFBFL_popd_result=$?
645         echo "common_build_and_checks.exec.sh no popd"
646         # shellcheck disable=SC2248
647         return ${LFBFL_popd_result}
648     }
649 }
650
651 common_build_and_checks "$@"

```

2.8 build_and_checks_dependencies/comparisons.libr.sh

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # The file "comparisons.sh" was renamed to "comparisons.libr.sh".
25
26 equal() {
27     # Returns 1 if all the arguments strings are equal.
28     # Renvoie 1 dans $? si toutes les chaînes en arguments sont égales.
29     # J'ai rerepensé au 3-way-merge
30     # et à mon article sur le principe de première différence
31     # appliqué à la largeur modulaire, de clique ou de rang avec
32     # l'idée d'un principe de première différence ternaire.
33     declare -i LFBFL_first=1
34     local LFBFL_current
35     local LFBFL_arg
36     for LFBFL_arg in "$@"; do
37         if [[ LFBFL_first -eq 1 ]]; then
38             LFBFL_current="${LFBFL_arg}"
39             LFBFL_first=0
40             continue
41         fi
42         if [[ "${LFBFL_current}" != "${LFBFL_arg}" ]]; then
43             return 0
44         fi
```

```

45     done
46     return 1
47 }
48
49 some_distinct() {
50     ! equal "$@"
51 }
52
53 all_distinct() {
54     # Returns 1 if all the arguments strings are distinct.
55     declare -ir LFBFL_ij_max=$#
56     declare -i LFBFL_i
57     declare -i LFBFL_j
58     for ((LFBFL_i=1; LFBFL_i<=LFBFL_ij_max; ++LFBFL_i)); do
59         for ((LFBFL_j=1; LFBFL_j<=LFBFL_ij_max; ++LFBFL_j)); do
60             [[ LFBFL_i -eq LFBFL_j ]] && continue
61             [[ "${!LFBFL_i}" == "${!LFBFL_j}" ]] && return 0
62         done
63     done
64     return 1
65 }
66
67 all_distinct2() {
68     # Returns 1 if all the arguments strings are distinct.
69     local LFBFL_element
70     local LFBFL_previous_element
71     if [[ $# -eq 0 ]]; then
72         return 1
73     fi
74     # Not setting pipefail since the result would still be incorrect.
75     # shellcheck disable=SC2312
76     printf "%s\n" "$@" \
77         | sort\
78         | {
79             read -r LFBFL_previous_element\
80             && while read -r LFBFL_element; do
81                 if [[ "${LFBFL_previous_element}" == "${LFBFL_element}" ]];
82                 then
83                     return 0
84                 fi
85                 LFBFL_previous_element="${LFBFL_element}"
86             done
87             return 1
88         }
89     }
90

```

```

91 all_distinct3() {
92     # Returns 1 if all the arguments strings are distinct.
93     # This is the fastest of all 3; but note that version 2 is faster
94     # than this one if a call to sort is added.
95     if [[ $# -eq 0 ]]; then
96         return 1
97     fi
98     # Not setting pipefail since the result would still be incorrect.
99     # shellcheck disable=SC2162,SC2312
100    printf "%s\n" "$@" \
101        | uniq -d\
102        | read
103    }
104
105 some_equal() {
106     ! all_distinct "$@"
107 }
108
109 some_equal2() {
110     ! all_distinct2 "$@"
111 }
112
113 some_equal3() {
114     ! all_distinct3 "$@"
115 }
116
117 max() {
118     # Returns the maximum string among the arguments.
119     # Renvoie la chaîne la plus grande parmi celles proposées.
120     # $1=$sort command with the flags you want to be applied.
121     # $ max sort a b c
122     # C
123     if [[ $# -eq 0 ]]; then
124         return 1
125     fi
126     if [[ ! -o pipefail ]]; then
127         set -o pipefail
128         trap 'set +o pipefail' RETURN
129     fi
130     printf "%s\n" "${@:2}" \
131         | eval "$1 -r" \
132         | head -1
133    }
134
135 min() {
136     # Returns the minimum string among the arguments.

```

```

137 # Renvoie la chaîne la plus petite parmi celles proposées.
138 # $1=$sort command with the flags you want to be applied.
139 # $ min sort a b c
140 # a
141 if [[ $# -eq 0 ]]; then
142     return 1
143 fi
144 if [[ ! -o pipefail ]]; then
145     set -o pipefail
146     trap 'set +o pipefail' RETURN
147 fi
148 printf "%s\n" "${@:2}" \
149     | eval "$1" \
150     | head -1
151 }
152
153 is_substring(){
154     # $1=$string
155     # $2=$substring
156     declare -r LFBFL_var_1=$((
157         echo "$1" \
158         | sed -e 's/\\/\&/g' -e "s/'/\\\\\&/g"
159     ))
160     declare -r LFBFL_var_2=$((
161         echo "$2" \
162         | sed -e 's/\\/\&/g' -e "s/'/\\\\\&/g"
163     ))
164     local LFBFL_var_3="\$a = '$LFBFL_var_1';"
165     LFBFL_var_3+=" \$b = preg_quote('$LFBFL_var_2');"
166     LFBFL_var_3+=" \$c = '/'.addcslashes(\$b, '/').'/';"
167     LFBFL_var_3+=" die(1 - preg_match(\$c, \$a));"
168     readonly LFBFL_var_3
169     php -r "{$LFBFL_var_3}"
170     return $?
171 }
172
173 is_subfile(){
174     # $1=$file_name
175     # $2=$sub_file_name
176     local LFBFL_var_1="\$a = file_get_contents('$1');"
177     LFBFL_var_1+=" \$b = preg_quote(file_get_contents('$2'));"'
178     LFBFL_var_1+=" \$c = '/'.addcslashes(\$b, '/').'/';"
179     LFBFL_var_1+=" die(1 - preg_match(\$c, \$a));"
180     readonly LFBFL_var_1
181     php -r "{$LFBFL_var_1}"
182     return $?

```

```

183 }
184
185 alias min_from_int_sort="min 'sort --numeric-sort'"
186 alias max_from_int_sort="max 'sort --numeric-sort'"

```

2.9 build_and_checks_dependencies/create_PDF.exec.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "create_PDF.sh" to "create_PDF.exec.sh".
25
26 LFBFL_subdir="build_and_checks_dependencies"
27 # shellcheck source=generate_from_template.libr.sh
28 source "./${LFBFL_subdir}/generate_from_template.libr.sh"
29 # shellcheck source=lines_counts.libr.sh
30 source "./${LFBFL_subdir}/lines_counts.libr.sh"
31 # shellcheck source=lines_filters.libr.sh
32 source "./${LFBFL_subdir}/lines_filters.libr.sh"
33 # shellcheck source=lines_maps.libr.sh
34 source "./${LFBFL_subdir}/lines_maps.libr.sh"
35 # shellcheck source=overwrite_if_not_equal.libr.sh
36 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
37 # shellcheck source=strings_functions.libr.sh
38 source "./${LFBFL_subdir}/strings_functions.libr.sh"
39

```

```

40  create_PDF() {
41      declare -i LFBFL_verbose=0
42      if [[ "$*" == *--verbose* ]]; then
43          echo "$0 $*"
44          LFBFL_verbose=1
45      fi
46      readonly LFBFL_verbose
47
48      if [[ ! -o pipefail ]]; then
49          [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
50          set -o pipefail
51          trap 'set +o pipefail' RETURN
52      fi
53
54      local LFBFL_subdir2="build_and_checks_variables"
55      local LFBFL_data_file_name="${LFBFL_subdir2}/repository_data.txt"
56      local LFBFL_repository_name=""
57      grep_variable "${LFBFL_data_file_name}" repository_name \
58          --result-variable-prefix="LFBFL_" \
59          --replace-line-returns-by=""
60
61      local LFBFL_repository_web_url=""
62      grep_variable "${LFBFL_data_file_name}" repository_web_url \
63          --result-variable-prefix="LFBFL_" \
64          --replace-line-returns-by=""
65
66      local LFBFL_repository_git_url=""
67      grep_variable "${LFBFL_data_file_name}" repository_git_url \
68          --result-variable-prefix="LFBFL_" \
69          --replace-line-returns-by=""
70
71      local LFBFL_abstract=""
72      grep_variable "${LFBFL_data_file_name}" abstract \
73          --result-variable-prefix="LFBFL_"
74
75      local LFBFL_acknowledgments=""
76      grep_variable "${LFBFL_data_file_name}" acknowledgments \
77          --result-variable-prefix="LFBFL_"
78
79      local LFBFL_author_email=""
80      grep_variable "${LFBFL_data_file_name}" author_email \
81          --result-variable-prefix="LFBFL_" \
82          --replace-line-returns-by=""
83
84      local LFBFL_author_full_name=""
85      grep_variable "${LFBFL_data_file_name}" author_full_name \

```

```

86      --result-variable-prefix="LFBFL_"\ \
87      --replace-line-returns-by=" "
88
89  local LFBFL_author_website=""
90  grep_variable "${LFBFL_data_file_name}" author_website\
91      --result-variable-prefix="LFBFL_"\ \
92      --replace-line-returns-by=" "
93
94  declare -i LFBFL_max_line_length
95  grep_variable "${LFBFL_data_file_name}" max_line_length\
96      --result-variable-prefix="LFBFL_"
97
98  declare -r LFBFL_current_date=$(date -I"date")
99
100 declare -r LFBFL_current_git_SHA1=$(git rev-parse HEAD)
101
102 declare -ir LFBFL_number_of_commits=$(
103     git shortlog\
104     | space_starting_lines\
105     | wc -l
106 )
107
108 declare -ir LFBFL_all_code_lines_count=$(code_lines_count_all "$@")
109 declare -ir LFBFL_not_empty_code_lines_count=$(
110     code_lines_count_not_empty "$@"
111 )
112 declare -ir LFBFL_empty_code_lines_count=$(
113     code_lines_count_empty "$@"
114 )
115
116 local LFBFL_number_of_lines="${LFBFL_all_code_lines_count}"
117 LFBFL_number_of_lines+=" total lines,"
118 LFBFL_number_of_lines+=" ${LFBFL_not_empty_code_lines_count}"
119 LFBFL_number_of_lines+=" not empty lines,"
120 LFBFL_number_of_lines+=" ${LFBFL_empty_code_lines_count}"
121 LFBFL_number_of_lines+=" empty lines."
122 readonly LFBFL_number_of_lines
123
124 # see https://gitlab.com/OldManProgrammer/unix-tree/-/issues
125 # ?show=eyJpaWQioI0MyIsImZ1bGxfcGF0aCI6Ik9sZE1hb1Byb2dyYWlt
126 # ZXIvdW5peC10cmVlIiwiaWQiOjE4NTE0NTgzOH0%3D
127 tree -a --gitignore\
128     -I "node_modules/"\
129     -I "__pycache__/"\
130     -I ".mypy_cache/"\
131     -I ".ruff_cache/"

```

```

132     -I ".git/" \
133     | replace_non_ascii_spaces \
134     > "${LFBFL_subdir2}/temp/current_tree_light.txt"
135
136     tree -a -DFh --gitignore \
137         -I "node_modules/" \
138         -I "__pycache__/" \
139         -I ".mypy_cache/" \
140         -I ".ruff_cache/" \
141         -I ".git/" \
142         | replace_non_ascii_spaces \
143         > "${LFBFL_subdir2}/temp/current_tree.txt.temp"
144
145     local LFBFL_temp_files_listing="./${LFBFL_subdir2}/temp/"
146     LFBFL_temp_files_listing+="files_listing.tex.tpl.temp"
147     readonly LFBFL_temp_files_listing
148     : > "${LFBFL_temp_files_listing}"
149     local LFBFL_temp_files_listing2="./${LFBFL_subdir2}/temp/"
150     LFBFL_temp_files_listing2+="files_listing.html.tpl.temp"
151     readonly LFBFL_temp_files_listing2
152     : > "${LFBFL_temp_files_listing2}"
153     # HTML <li> elements, hence "lis".
154     local LFBFL_temp_files_lis
155     LFBFL_temp_files_lis="./${LFBFL_subdir2}/temp/files_lis.html.tpl"
156     readonly LFBFL_temp_files_lis
157     : > "${LFBFL_temp_files_lis}"
158     # shellcheck disable=SC2248
159     get_split_score_simple 1 ${LFBFL_max_line_length} /
160     declare -r LFBFL_score_command="${get_split_score_result}"
161     local LFBFL_score_command_properties="${get_split_score_result2}"
162     readonly LFBFL_score_command_properties
163     # shellcheck disable=SC2248
164     get_split_score_simple 1 ${LFBFL_max_line_length} ':'
165     declare -r LFBFL_score_command2="${get_split_score_result}"
166     local LFBFL_score_command_properties2="${get_split_score_result2}"
167     readonly LFBFL_score_command_properties2
168     declare -r LFBFL_suffix='%'
169     declare -i LFBFL_i=0
170     local LFBFL_cleaned_path1
171     local LFBFL_cleaned_path2
172     local LFBFL_new_lines
173     local LFBFL_new_lines2
174     local LFBFL_new_lines3
175     # Remove line returns here to keep lines short.
176     sed -Ez 's/\n//Mg' "./${LFBFL_subdir2}/files_names_listing.txt" \
177         | grep -v '^//\'
```

```

178     | while read -r LFBFL_file_name;
179 do
180     echo "Listing file for tex/HTML : ${LFBFL_file_name}"
181     # echo "Listing file for tex : ${LFBFL_file_name}"
182     # LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
183     LFBFL_cleaned_path1="${LFBFL_file_name}/_/\_\_"
184     LFBFL_cleaned_path2=(
185         echo "${LFBFL_file_name}"\
186         | sed -e 's/\//:/g' -e 's/\././g'
187     )
188     LFBFL_new_lines=" ${LFBFL_cleaned_path1}"
189     if [[ ${#LFBFL_new_lines} -gt LFBFL_max_line_length ]]; then
190         # shellcheck disable=SC2248
191         repeated_split_last_line "${LFBFL_new_lines}"\
192             "\n"\
193             ${LFBFL_max_line_length}\
194             "${LFBFL_suffix}"\
195             "${LFBFL_score_command}"\
196             "${LFBFL_score_command_properties}"\
197             "\\\\""
198     LFBFL_new_lines=${repeated_split_last_line_result}
199 fi
200 LFBFL_new_lines2=" ${LFBFL_cleaned_path2}"
201 if [[ ${#LFBFL_new_lines2} -gt LFBFL_max_line_length ]]; then
202     # shellcheck disable=SC2248
203     repeated_split_last_line "${LFBFL_new_lines2}"\
204         "\n"\
205         ${LFBFL_max_line_length}\
206         "${LFBFL_suffix}"\
207         "${LFBFL_score_command2}"\
208         "${LFBFL_score_command_properties2}"
209     LFBFL_new_lines2=${repeated_split_last_line_result}
210 fi
211 LFBFL_new_lines3="${LFBFL_file_name}"
212 if [[ ${#LFBFL_new_lines3} -gt LFBFL_max_line_length ]]; then
213     # shellcheck disable=SC2248
214     repeated_split_last_line "${LFBFL_new_lines3}"\
215         "\n"\
216         ${LFBFL_max_line_length}\
217         "${LFBFL_suffix}"\
218         "${LFBFL_score_command}"\
219         "${LFBFL_score_command_properties}"
220     LFBFL_new_lines3=${repeated_split_last_line_result}
221 fi
222 {
223     echo "\subsection{"

```

```

224     echo "  ${LFBFL_file_name}" \
225         | sed -e "s|  ${LFBFL_file_name}|${LFBFL_new_lines}|g" \
226             -e 's/_/\\_/_g'
227     echo "}"
228     echo "\label{"
229     echo "  ${LFBFL_cleaned_path2}" \
230         | sed -e "s|  ${LFBFL_cleaned_path2}|${LFBFL_new_lines2}|g"
231     echo "}"
232     echo ""
233     echo "\VerbatimInput[numbers=left,xleftmargin=-5mm] {"
234     echo "  ${LFBFL_file_name}" \
235         | sed -e "s|  ${LFBFL_file_name}|${LFBFL_new_lines3}|g"
236     echo "}"
237     echo ""
238     echo ""
239 } >> "${LFBFL_temp_files_listing}"

240
241 # echo "Listing file for HTML : ${LFBFL_file_name}"
242 ((++LFBFL_i))
243 LFBFL_new_lines="${LFBFL_file_name}"
244 if [[ ${#LFBFL_file_name} -gt LFBFL_max_line_length ]]; then
245     # shellcheck disable=SC2248
246     repeated_split_last_line "${LFBFL_new_lines}" \
247         "-->" \
248         ${LFBFL_max_line_length}\ \
249         "<!--"\ \
250         "${LFBFL_score_command}"\ \
251         "${LFBFL_score_command_properties}"\ \
252         LFBFL_new_lines=${repeated_split_last_line_result}
253 fi
254 {
255     echo "<h3 id=\"subsection2.${LFBFL_i}\">2.${LFBFL_i}"
256     # shellcheck disable=SC2001
257     echo "${LFBFL_file_name}"\
258         | sed -e "s|${LFBFL_file_name}|${LFBFL_new_lines}|g"
259     echo "</h3>"
260     echo "<pre class=\"numbered_lines\">"
261     sed -e 's/&/\&/g' -e 's/</\</g' \
262         < "${LFBFL_file_name}"
263     echo "</pre>"
264     echo ""
265     echo ""
266 } >> "${LFBFL_temp_files_listing2}"
267 {
268     echo "      <li><a href=\"#subsection2.${LFBFL_i}\">"
269     # shellcheck disable=SC2001

```

```

270     echo "${LFBFL_file_name}" \
271         | sed -e "s|${LFBFL_file_name}|${LFBFL_new_lines}|g"
272     echo "          </a></li>" 
273 } >> "${LFBFL_temp_files_lis}"
274 done
275 overwrite_if_not_equal "./${LFBFL_subdir2}/files_listing.tex.tpl" \
276     "${LFBFL_temp_files_listing}"
277 sed -i -e "s|\n</pre>|</pre>|g" "${LFBFL_temp_files_listing2}"
278 overwrite_if_not_equal \
279     "./${LFBFL_subdir2}/temp/files_listing.html.tpl" \
280     "${LFBFL_temp_files_listing2}"
281
282 # We verify if some lines are beyond max_line_length characters
283 # in current_tree_light.txt and current_tree.txt.
284 declare -ar LFBFL_trees=( 
285     "current_tree_light.txt"
286     "current_tree.txt.temp"
287 )
288 local LFBFL_tree_path
289 local LFBFL_line
290 local LFBFL_prefix
291 local LFBFL_line_start
292 declare -ir LFBFL_overlength=$((LFBFL_max_line_length+1))
293 for LFBFL_tree in "${LFBFL_trees[@]}"; do
294     LFBFL_tree_path="${LFBFL_subdir2}/temp/${LFBFL_tree}"
295     # shellcheck disable=SC2248
296     grep '\.'"${LFBFL_overlength}"' ' "${LFBFL_tree_path}" \
297         | while read -r LFBFL_line;
298 do
299     # echo "LFBFL_line: ${LFBFL_line}"
300     LFBFL_prefix=$( 
301         echo "${LFBFL_line}" \
302             | sed -E -e 's/(.*)-[^-]+$/\1/g' -e 's/[^\ ]+$/\ /'
303     )
304     LFBFL_prefix+="| "
305     # echo "LFBFL_prefix: ${LFBFL_prefix}"
306     LFBFL_file_name=$( 
307         echo "${LFBFL_line}" \
308             | sed -E 's|.* ([a-zA-Z0-9\._/-]+).*$|\1|g'
309     )
310     # echo "LFBFL_file_name: ${LFBFL_file_name}"
311     LFBFL_line_start=$( 
312         echo "${LFBFL_line}" \
313             | sed -E -e "s/(.*)[\ ]*${LFBFL_file_name}/\1/g" \
314                 -e 's/ *$/\ /'
315     )

```

```

316     # echo "LFBFL_line_start: ${LFBFL_line_start}"
317     LFBFL_line=$(
318         echo "${LFBFL_line}" \
319         | sed -E -e 's/\\/[\\]/g' -e 's/\\]/\\\\]/g'
320     )
321     LFBFL_new_lines="${LFBFL_prefix}${LFBFL_file_name}"
322     if [[ ${#LFBFL_new_lines} -gt LFBFL_max_line_length ]]; then
323         # shellcheck disable=SC2248
324         repeated_split_last_line "${LFBFL_new_lines}" \
325             "${LFBFL_prefix}" \
326             ${LFBFL_max_line_length} \
327             " " \
328             " " \
329             " "
330         LFBFL_new_lines=${repeated_split_last_line_result}
331     fi
332     sed -i -e \
333         "s/${LFBFL_line}/${LFBFL_line_start}\n${LFBFL_new_lines}/g" \
334         "${LFBFL_tree_path}"
335     done
336 done
337
338 overwrite_if_not_equal "${LFBFL_subdir2}/temp/current_tree.txt" \
339     "${LFBFL_subdir2}/temp/current_tree.txt.temp" 1 1
340
341 declare -r LFBFL_tex_path_start=\
342     "./${LFBFL_subdir2}/temp/${LFBFL_repository_name}.tex"
343 declare -r LFBFL_html_path_start=\
344     "./${LFBFL_subdir2}/temp/${LFBFL_repository_name}.html"
345
346 if [[ -f "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex.tpl" ]]; then
347     # If there is a template, we init the process from it.
348     cp "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex.tpl" \
349         "${LFBFL_tex_path_start}.1"
350     elif [[ -f "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex" ]]; then
351         # Otherwise if there is a tex file, we init the process from it.
352         cp "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex" \
353             "${LFBFL_tex_path_start}.1"
354     else
355         echo "Neither .tex.tpl, nor .tex in ./${LFBFL_subdir2}/"
356     fi
357
358     # Same logic with repository HTML file.
359     if [[ -f "./${LFBFL_subdir2}/${LFBFL_repository_name}.html.tpl" ]];
```

```

362     then
363         cp "./${LFBFL_subdir2}/${LFBFL_repository_name}.html.tpl" \
364             "${LFBFL_html_path_start}.1"
365     elif [[ -f "./${LFBFL_repository_name}.html" ]]; then
366         cp "./${LFBFL_repository_name}.html" "${LFBFL_html_path_start}.1"
367     else
368         echo "Neither .html.tpl in ./${LFBFL_subdir2}/, nor .html in ./"
369     fi
370
371     # shellcheck disable=SC2001
372     echo "${LFBFL_abstract}" \
373         | sed -e 's/\n/g' \
374         > "./${LFBFL_subdir2}/temp/abstract_temp"
375
376     # shellcheck disable=SC2001
377     echo "${LFBFL_acknowledgments}" \
378         | sed -e 's/\n/g' \
379         > "./${LFBFL_subdir2}/temp/acknowledgments_temp"
380
381     # But the filling still occurs, in case the dev want to refill
382     # part of the tex/html file that he modified with @token@
383     # using some computed results.
384     # Tex filling:
385     if [[ -f "${LFBFL_tex_path_start}.1" ]]; then
386         sed -i -e "s|@repository_name@|${LFBFL_repository_name}|g" \
387             -e "s|@repository_web_url@|${LFBFL_repository_web_url}|g" \
388             -e "s|@repository_git_url@|${LFBFL_repository_git_url}|g" \
389             -e "s|@author_email@|${LFBFL_author_email}|g" \
390             -e "s|@author_full_name@|${LFBFL_author_full_name}|g" \
391             -e "s|@author_website@|${LFBFL_author_website}|g" \
392             -e "s|@current_date@|${LFBFL_current_date}|g" \
393             -e "s|@current_git_SHA1@|${LFBFL_current_git_SHA1}|g" \
394             -e "s|@number_of_commits@|${LFBFL_number_of_commits}|g" \
395             -e "s|@number_of_lines@|${LFBFL_number_of_lines}|g" \
396         "${LFBFL_tex_path_start}.1"
397
398         insert_file_at_token "${LFBFL_tex_path_start}.1" @abstract@ \
399             "./${LFBFL_subdir2}/temp/abstract_temp" \
400             "${LFBFL_tex_path_start}.2"
401
402         insert_file_at_token "${LFBFL_tex_path_start}.2" \
403             @acknowledgments@ \
404             "./${LFBFL_subdir2}/temp/acknowledgments_temp" \
405             "${LFBFL_tex_path_start}.3"
406
407         pushd "./${LFBFL_subdir2}/temp/" || (echo "pushd failed" && exit)

```

```

408     sed -i \
409         -e '/@current_tree_light@/{r current_tree_light.txt' -e 'd}' \
410         -e '/@current_tree@/{r current_tree.txt' -e 'd}' \
411         "${LFBFL_repository_name}.tex.3"
412     popd || (echo "popd failed" && exit)
413
414     insert_file_at_token "${LFBFL_tex_path_start}.3" \
415         @files_listing_VerbatimInput@ \
416         "./${LFBFL_subdir2}/files_listing.tex.tpl" \
417         "${LFBFL_tex_path_start}.4"
418
419     overwrite_if_not_equal \
420         "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex" \
421         "${LFBFL_tex_path_start}.4" 1
422 fi
423
424 # HTML filling:
425 declare -i LFBFL_HTML_updated=0
426 if [[ -f "${LFBFL_html_path_start}.1" ]]; then
427     sed -i -e "s|@repository_name@|${LFBFL_repository_name}|g" \
428             -e "s|@repository_web_url@|${LFBFL_repository_web_url}|g" \
429             -e "s|@repository_git_url@|${LFBFL_repository_git_url}|g" \
430             -e "s|@author_email@|${LFBFL_author_email}|g" \
431             -e "s|@author_full_name@|${LFBFL_author_full_name}|g" \
432             -e "s|@author_website@|${LFBFL_author_website}|g" \
433             -e "s|@current_date@|${LFBFL_current_date}|g" \
434             -e "s|@current_git_SHA1@|${LFBFL_current_git_SHA1}|g" \
435             -e "s|@number_of_commits@|${LFBFL_number_of_commits}|g" \
436             -e "s|@number_of_lines@|${LFBFL_number_of_lines}|g" \
437         "${LFBFL_html_path_start}.1"
438
439     insert_file_at_token "${LFBFL_html_path_start}.1" @abstract@ \
440         "./${LFBFL_subdir2}/temp/abstract_temp" \
441         "${LFBFL_html_path_start}.2"
442
443     insert_file_at_token "${LFBFL_html_path_start}.2" \
444         @acknowledgments@ \
445         "./${LFBFL_subdir2}/temp/acknowledgments_temp" \
446         "${LFBFL_html_path_start}.3"
447
448     insert_file_at_token "${LFBFL_html_path_start}.3" \
449         @files_lis@ "${LFBFL_temp_files_lis}" \
450         "${LFBFL_html_path_start}.4"
451
452     pushd "./${LFBFL_subdir2}/temp/" || (echo "pushd failed" && exit)
453     sed -i \

```

```

454     -e '/@current_tree_light@/{r current_tree_light.txt' -e 'd}' '\
455     -e '/@current_tree@/{r current_tree.txt' -e 'd}' '\
456     "${LFBFL_repository_name}.html.4"
457 popd || (echo "popd failed" && exit)
458
459     insert_file_at_token "${LFBFL_html_path_start}.4" \
460         @files_listing_HTMLPreInput@\
461         "./${LFBFL_subdir2}/temp/files_listing.html.tpl" \
462         "${LFBFL_html_path_start}.5"
463
464     overwrite_if_not_equal "./${LFBFL_repository_name}.html" \
465         "${LFBFL_html_path_start}.5" 1
466     LFBFL_HTML_updated=$?
467 fi
468
469 # The HTML contains everything including listings with source code.
470 # It it doesn't get updated,
471 # then it is useless to recompute the PDF.
472 if [[ LFBFL_HTML_updated -eq 0 ]]; then
473     return
474 fi
475
476 if [[ LFBFL_verbose -eq 1 ]]; then
477     for ((i=0; i<3; i++)); do
478         pdflatex "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex"
479         done
480     else
481         for ((i=0; i<3; i++)); do
482             pdflatex "./${LFBFL_subdir2}/${LFBFL_repository_name}.tex" \
483                 > /dev/null
484             done
485     fi
486
487 LFBFL_files_to_temp=(\
488     "${LFBFL_repository_name}.aux" \
489     "${LFBFL_repository_name}.log" \
490     "${LFBFL_repository_name}.out" \
491     "${LFBFL_repository_name}.toc"
492 )
493 for LFBFL_file_name in "${LFBFL_files_to_temp[@]}"; do
494     mv "${LFBFL_file_name}" "${LFBFL_subdir2}/temp/"
495 done
496 }
497
498 create_PDF "$@"

```

2.10 build_and_checks_dependencies/generate_from_template.libr.sh

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "generate_from_template.sh" to
25 # "generate_from_template.libr.sh".
26
27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=lines_filters.libr.sh
29 source "./${LFBFL_subdir}/lines_filters.libr.sh"
30 # shellcheck source=overwrite_if_not_equal.libr.sh
31 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
32
33 generate_from_template_with_block_comments() {
34     # $1=base_file_name
35     # $2=target_file_name
36     # $3=enter_block_comment
37     # $4=exit_block_comment
38     local LFBFL_temp
39     LFBFL_temp="$2.generate_from_template_with_block_comments.temp"
40     readonly LFBFL_temp
41     {
42         echo "$3"
43         cat "$1"
44         echo "$4"
```

```

45     } > "${LFBFL_temp}"
46     overwrite_if_not_equal "$2" "${LFBFL_temp}"
47   }
48
49   generate_from_template_with_line_comments() {
50     # $1=base_file_name
51     # $2=target_file_name
52     # $3=line_comment_prefix
53     # $4=optional_post_processing
54     # $5=file_prefix
55     local LFBFL_temp
56     LFBFL_temp="$2.generate_from_template_with_line_comments.temp"
57     readonly LFBFL_temp
58     if [[ -n "$5" ]]; then
59       echo "$5" \
60         > "${LFBFL_temp}"
61     fi
62     sed -e "s/^/$3/g" "$1" \
63       >> "${LFBFL_temp}"
64     if [[ -n "$4" ]]; then
65       eval "$4"
66     fi
67     overwrite_if_not_equal "$2" "${LFBFL_temp}"
68   }
69
70   split_file_in_two() {
71     # $1=$file_name
72     # $2=$token assume that token is the only thing on his line.
73     # $3=$file_name_part1
74     # $4=$file_name_part2
75     declare -ir LFBFL_line_number=$(((
76       grep -n "$2" "$1" \
77         | head --lines=1 \
78         | cut -f 1 -d ':'
79     )))
80     declare -ir LFBFL_line_count=$(ll_wc -l -n "$1")
81     declare -ir LFBFL_lines_after=$(((
82       LFBFL_line_count - LFBFL_line_number
83     )))
84     head --lines="$((LFBFL_line_number - 1))" "$1" \
85       > "$3"
86     tail --lines="${LFBFL_lines_after}" "$1" \
87       > "$4"
88   }
89
90   insert_file_at_token() {

```

```

1 # $1=$file_name
2 # $2=$token assume that token is the only thing on his line.
3 # $3=$file_name_to_insert
4 # $4=$file_name_result
5 declare -r LFBFL_start_file_name="$1.insert_file_at_token1.temp"
6 declare -r LFBFL_end_file_name="$1.insert_file_at_token2.temp"
7 local LFBFL_result_file_name
8 if [[ -n "$4" ]]; then
9     LFBFL_result_file_name="$4"
10 else
11     LFBFL_result_file_name="$1.insert_file_at_token3.temp"
12 fi
13 readonly LFBFL_result_file_name
14 split_file_in_two "$1" "$2" "${LFBFL_start_file_name}" \
15     "${LFBFL_end_file_name}"
16 cat "${LFBFL_start_file_name}" "$3" "${LFBFL_end_file_name}" \
17     > "${LFBFL_result_file_name}"
18 rm "${LFBFL_start_file_name}" "${LFBFL_end_file_name}"
19 if [[ -z "$4" ]]; then
20     overwrite_if_not_equal "$1" "${LFBFL_result_file_name}"
21 fi
22 }

```

2.11 build_and_checks_dependencies/get_common_text_glob_patterns.libr.sh

```

1#!/usr/bin/env bash
2# This file is part of DevOrSysAdminScripts library.
3#
4# DevOrSysAdminScripts is free software:
5# you can redistribute it and/or modify it under the terms
6# of the GNU General Public License
7# as published by the Free Software Foundation,
8# either version 3 of the License,
9# or (at your option) any later version.
10#
11# DevOrSysAdminScripts is distributed in the hope
12# that it will be useful,
13# but WITHOUT ANY WARRANTY;
14# without even the implied warranty of
15# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16# See the GNU General Public License for more details.
17#
18# You should have received a copy of
19# the GNU General Public License
20# along with DevOrSysAdminScripts.
21# If not, see <https://www.gnu.org/licenses/>.

```

```

22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "get_common_text_glob_patterns.sh"
25 # to "get_common_text_glob_patterns.libr.sh".
26
27 get_COMMON_TEXT_PATHS_GLOB_PATTERNS() {
28     if [[ -n "${COMMON_TEXT_PATHS_GLOB_PATTERNS}" ]]; then
29         return
30     fi
31
32     declare -agr COMMON_TEXT_PATHS_GLOB_PATTERNS=(
33         '**/*.c'
34         '**/*.css'
35         '**/*.h'
36         '**/*.htm'
37         '**/*.html'
38         '**/*.js'
39         '**/*.json'
40         '**/*.md'
41         '**/*.php'
42         '**/*.py'
43         '**/*.sh'
44         '**/*.sql'
45         '**/*.tex'
46         '**/*.toml'
47         '**/*.tpl'
48         '**/*.ts'
49         '**/*.txt'
50         '**/*.yml'
51         '**/COPYING'
52         '**/COPYING.LESSER'
53         '**/pre-commit'
54         '**/post-commit'
55         '**/.gitignore'
56         '**/py.typed'
57     )
58 }
59
60
61 get_COMMON_TEXT_FILES_GLOB_PATTERNS() {
62     if [[ -n "${COMMON_TEXT_FILES_GLOB_PATTERNS}" ]]; then
63         return
64     fi
65
66     declare -agr COMMON_TEXT_FILES_GLOB_PATTERNS=(
67         '*.c'

```

```

68     ' *.css'
69     ' *.h'
70     ' *.htm'
71     ' *.html'
72     ' *.js'
73     ' *.json'
74     ' *.md'
75     ' *.php'
76     ' *.py'
77     ' *.sh'
78     ' *.sql'
79     ' *.tex'
80     ' *.toml'
81     ' *.tpl'
82     ' *.ts'
83     ' *.txt'
84     ' *.yml'
85     ' COPYING'
86     ' COPYING.LESSER'
87     ' pre-commit'
88     ' post-commit'
89     ' .gitignore'
90     ' py.typed'
91 )
92 }

```

2.12 build_and_checks_dependencies/grammar_and_spell_check.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of

```

```

19  # the GNU General Public License
20  # along with DevOrSysAdminScripts.
21  # If not, see <https://www.gnu.org/licenses/>.
22  #
23  # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24  # This file was renamed from "grammar_and_spell_check.sh"
25  # to "grammar_and_spell_check.libr.sh".
26
27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=get_common_text_glob_patterns.libr.sh
29 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
30 # shellcheck source=lines_filters.libr.sh
31 source "./${LFBFL_subdir}/lines_filters.libr.sh"
32
33 grammar_and_spell_check() {
34     declare -i LFBFL_verbose=0
35     if [[ "$*" == *--verbose* ]]; then
36         echo "$0 $*"
37         LFBFL_verbose=1
38     fi
39     readonly LFBFL_verbose
40
41     if [[ ! -o pipefail ]]; then
42         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
43         set -o pipefail
44         trap 'set +o pipefail' RETURN
45     fi
46
47     get_COMMON_TEXT_FILES_GLOB_PATTERNS
48     local LFBFL_grammar_or_spell_checker_command=""
49     grep_variable "$1" grammar_or_spell_checker_command\
50         --result-variable-prefix="LFBFL_"\
51         --replace-line-returns-by="""
52     declare -r LFBFL_command="${LFBFL_grammar_or_spell_checker_command}"
53     if [[ -z "${LFBFL_command}" ]]; then
54         return
55     fi
56     local LFBFL_file_path
57     local LFBFL_pattern
58     local LFBFL_eval_string
59     for LFBFL_pattern in "${COMMON_TEXT_FILES_GLOB_PATTERNS[@]}"; do
60         [[ LFBFL_verbose -eq 0 ]]\
61             || echo "Iterating on pattern: ${LFBFL_pattern}"
62         find . -type f -name "${LFBFL_pattern}" -printf '%P\n'\
63             | while read -r LFBFL_file_path;
64     do

```

```

65     LFBFL_eval_string="${LFBFL_command} ${LFBFL_file_path}"
66     eval "${LFBFL_eval_string}"
67   done
68 done
69 }

```

2.13 build_and_checks_dependencies/licenses_templates/build_licenses_templates.exec.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "build_licenses_templates.sh"
25 # to "build_licenses_templates.exec.sh".
26
27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=comparisons.libr.sh
29 source "./${LFBFL_subdir}/comparisons.libr.sh"
30 # shellcheck source=generate_from_template.libr.sh
31 source "./${LFBFL_subdir}/generate_from_template.libr.sh"
32 # shellcheck source=lines_filters.libr.sh
33 source "./${LFBFL_subdir}/lines_filters.libr.sh"
34 # shellcheck source=overwrite_if_not_equal.libr.sh
35 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
36
37 build_licenses_templates() {
38   local LFBFL_working_directory=$1

```

```

39     if [[ -z "$1" ]]; then
40         LFBFL_working_directory="."
41     fi
42     readonly LFBFL_working_directory
43     declare -i LFBFL_verbose=0
44     if [[ "$*" == *--verbose* ]]; then
45         echo "$0 $*"
46         LFBFL_verbose=1
47     fi
48     readonly LFBFL_verbose
49
50     if [[ ! -o pipefail ]]; then
51         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
52         set -o pipefail
53         trap 'set +o pipefail' RETURN
54     fi
55
56     # Source
57     local LFBFL_license_source_subdir
58     LFBFL_license_source_subdir="./${LFBFL_subdir}/licenses_templates/"
59     readonly LFBFL_license_source_subdir
60     local LFBFL_source_license_prefix="${LFBFL_license_source_subdir}"
61     LFBFL_source_license_prefix+="license_file_header_"
62     readonly LFBFL_source_license_prefix
63     # Target
64     local LFBFL_license_target_subdir
65     LFBFL_license_target_subdir="${LFBFL_working_directory}/"
66     LFBFL_license_target_subdir="build_and_checks_variables/temp/"
67     readonly LFBFL_license_target_subdir
68     local LFBFL_target_license_prefix="${LFBFL_license_target_subdir}"
69     LFBFL_target_license_prefix+="license_file_header_"
70     readonly LFBFL_target_license_prefix
71
72     declare -ar LFBFL_licenses=(
73         "AGPLv3+"
74         "GPLv3+"
75         "LGPLv3+"
76     )
77
78     # -----
79     # Various language definitions for text file formats
80     declare -ar LFBFL_block_comment_languages=(
81         "c"
82         "py"
83     )
84     declare -ar LFBFL_block_comment_enters=(
```

```

85      '/ *'
86      ' """'
87  )
88 declare -ar LFBFL_block_comment_exits=(
89      ' */'
90      ' """'
91  )
92 equal "${#LFBFL_block_comment_languages[@]}"\ \
93     "${#LFBFL_block_comment_enters[@]}"\ \
94     "${#LFBFL_block_comment_exits[@]}"
95 declare -i LFBFL_equal_result=$?
96 if [[ LFBFL_equal_result -eq 0 ]]; then
97     echo $'/!\`Problem with the definitions of languages arrays 1/!\\''
98 fi
99
100 declare -ar LFBFL_line_comment_languages=(
101     "sh"
102     "tex"
103  )
104 declare -ar LFBFL_line_comment_prefixes=(
105     '# '
106     '% '
107  )
108 equal "${#LFBFL_line_comment_languages[@]}"\ \
109     "${#LFBFL_line_comment_prefixes[@]}"
110 LFBFL_equal_result=$?
111 if [[ LFBFL_equal_result -eq 0 ]]; then
112     echo $'/!\`Problem with the definitions of languages arrays 2/!\\''
113 fi
114 # -----
115
116 # -----
117 # We generate the license header templates adapted to various text
118 # files formats.
119 # Kamoulox do endfor, Jacques Beauregard XD
120 # (C'est toujours mieux que l'inverse :) !)
121 # for do endfor -> for do done (Fort Doux Donne) .
122 local LFBFL_some_license
123 local LFBFL_license_prefix
124 local LFBFL_license_prefix2
125 local LFBFL_extension
126 local LFBFL_license_file_name
127 local LFBFL_enter_string
128 local LFBFL_exit_string
129 local LFBFL_temp2
130 local LFBFL_prefix_string

```

```

131 local LFBFL_intermediate_file_name
132 local LFBFL_file_prefix
133 declare -i LFBFL_generate_from_template_result
134 for LFBFL_some_license in "${LFBFL_licenses[@]}"; do
135     LFBFL_license_prefix="${LFBFL_source_license_prefix}"
136     LFBFL_license_prefix+="${LFBFL_some_license}"
137     LFBFL_license_prefix2="${LFBFL_target_license_prefix}"
138     LFBFL_license_prefix2+="${LFBFL_some_license}"
139     for ((i=0; i<${#LFBFL_block_comment_languages[@]}; i++)); do
140         LFBFL_extension=${LFBFL_block_comment_languages[i]}
141         LFBFL_license_file_name="${LFBFL_license_prefix2}"
142         LFBFL_license_file_name+=".${LFBFL_extension}"
143         LFBFL_enter_string=${LFBFL_block_comment_enters[i]}
144         LFBFL_exit_string=${LFBFL_block_comment_exits[i]}
145         generate_from_template_with_block_comments \
146             "${LFBFL_license_prefix}.tpl" \
147             "${LFBFL_license_file_name}.tpl" \
148             "${LFBFL_enter_string}" "${LFBFL_exit_string}"
149         LFBFL_generate_from_template_result=$?
150         if [[ LFBFL_verbose -eq 1 ]]; then
151             if [[ LFBFL_generate_from_template_result -eq 1 ]]; then
152                 echo \
153                     "License template ${LFBFL_license_file_name}.tpl updated"
154             elif [[ LFBFL_generate_from_template_result -eq 2 ]]; then
155                 echo \
156                     "License template ${LFBFL_license_file_name}.tpl created"
157             fi
158         fi
159     done
160     LFBFL_temp2=".generate_from_template_with_line_comments.temp"
161     for ((i=0; i<${#LFBFL_line_comment_languages[@]}; i++)); do
162         LFBFL_extension=${LFBFL_line_comment_languages[i]}
163         LFBFL_license_file_name="${LFBFL_license_prefix2}"
164         LFBFL_license_file_name+=".${LFBFL_extension}"
165         LFBFL_prefix_string=${LFBFL_line_comment_prefixes[i]}
166         LFBFL_intermediate_file_name="${LFBFL_license_file_name}.tpl"
167         LFBFL_intermediate_file_name+=".${LFBFL_temp2}"
168         LFBFL_file_prefix=""
169         if [[ "${LFBFL_extension}" == "sh" ]]; then
170             LFBFL_file_prefix="#!/usr/bin/env bash"
171         fi
172         generate_from_template_with_line_comments \
173             "${LFBFL_license_prefix}.tpl" \
174             "${LFBFL_license_file_name}.tpl" \
175             "${LFBFL_prefix_string}" \
176             "sed -Ei -e 's/\s*\$/g' '${LFBFL_intermediate_file_name}'"

```

```

177         "${LFBFL_file_prefix}"
178     LFBFL_generate_from_template_result=$?
179     if [[ LFBFL_verbose -eq 1 ]]; then
180         if [[ LFBFL_generate_from_template_result -eq 1 ]]; then
181             echo\
182                 "License template ${LFBFL_license_file_name}.tpl updated"
183             elif [[ LFBFL_generate_from_template_result -eq 2 ]]; then
184                 echo\
185                     "License template ${LFBFL_license_file_name}.tpl created"
186             fi
187         fi
188     done
189 done
190 # -----
191
192 # Now that base license templates generated languages license
193 # templates in working directory, we can go there.
194 declare -i LFBFL_cd_result
195 pushd .
196 cd "${LFBFL_working_directory}" || {
197     LFBFL_cd_result=$?
198     echo "build_licenses_templates no such directory"
199     # shellcheck disable=SC2248
200     return ${LFBFL_cd_result}
201 }
202
203 local LFBFL_data_file_name="build_and_checks_variables/"
204 LFBFL_data_file_name+="repository_data.txt"
205 readonly LFBFL_data_file_name
206 local LFBFL_repository_name=""
207 grep_variable "${LFBFL_data_file_name}" repository_name\
208     --result-variable-prefix="LFBFL_"\
209     --replace-line-returns-by="
210 local LFBFL_license=""
211 grep_variable "${LFBFL_data_file_name}" license\
212     --result-variable-prefix="LFBFL_"
213 local LFBFL_license2=""
214 grep_variable "${LFBFL_data_file_name}" license2\
215     --result-variable-prefix="LFBFL_"
216 local LFBFL_author_full_name=""
217 grep_variable "${LFBFL_data_file_name}" author_full_name\
218     --result-variable-prefix="LFBFL_"\
219     --replace-line-returns-by=" "
220
221 LFBFL_license_prefix="build_and_checks_variables/temp/"
222 LFBFL_license_prefix+="license_file_header_"

```

```

223 readonly LFBFL_license_prefix
224 LFBFL_license_prefix2="${LFBFL_license_prefix}${LFBFL_license}"
225 readonly LFBFL_license_prefix2
226 declare -r\
227     LFBFL_license_prefix3="${LFBFL_license_prefix}${LFBFL_license2}"
228
229 # First year according to current state of git repository.
230 declare -ir LFBFL_first_year=$(
231     git log\
232     | grep 'Date:'\
233     | cut -f 8 -d ' '\
234     | tail -1
235 )
236 # Last year according to current state of git repository.
237 declare -ir LFBFL_last_year=$(
238     git log\
239     | grep 'Date:'\
240     | cut -f 8 -d ' '\
241     | head -1
242 )
243 local LFBFL_copyright_string
244 LFBFL_copyright_string="${LFBFL_first_year}-${LFBFL_last_year}"
245 LFBFL_copyright_string+=" ${LFBFL_author_full_name}"
246 readonly LFBFL_copyright_string
247
248 prepare_filled_license_file_block() {
249     # $1=license_file_name
250     sed -e "s/@repository_name@/${LFBFL_repository_name}/g" \
251         -e "s/@copyright_string@/${LFBFL_copyright_string}/g" \
252             "$1.tpl" \
253             > "$1.temp"
254     overwrite_if_not_equal "$1" "$1.temp"
255     declare -r LFBFL_result=$?
256     if [[ LFBFL_verbose -eq 1 ]]; then
257         if [[ LFBFL_result -eq 1 ]]; then
258             echo "License template $1 updated"
259         elif [[ LFBFL_result -eq 2 ]]; then
260             echo "License template $1 created"
261         fi
262     fi
263     head --lines=-1 "$1" \
264         | tail --lines=+2 \
265         > "$1.temp"
266     # Strange world: tail --lines=-1 was working
267     # to exclude first line.
268     # I tested it months ago, and saw with my eyes the start and end

```

```

269 # block comments removed with:
270 # head --lines=-1 "$1" | tail --lines=-1.
271 # Now, man page says to use --lines=+2 instead.
272 # Welcome in the world where AI change your environment and your
273 # documentation on the fly...
274 }
275
276 declare -Ar LFBFL_all_block_comment_languages=(
277     ["c"]="c"
278     ["php"]="c"
279     ["py"]="py"
280 )
281
282 local LFBFL_key
283 local LFBFL_dest
284 local LFBFL_license_file_name2
285 declare -i LFBFL_not_subfile
286 declare -i LFBFL_not_subfile2
287 for LFBFL_key in "${!LFBFL_all_block_comment_languages[@]}"; do
288     LFBFL_dest=${LFBFL_all_block_comment_languages[$LFBFL_key]}
289     LFBFL_license_file_name="${LFBFL_license_prefix2}.${LFBFL_dest}"
290     prepare_filled_license_file_block "${LFBFL_license_file_name}"
291     if [[ -n "${LFBFL_license2}" ]]; then
292         LFBFL_license_file_name2="${LFBFL_license_prefix3}"
293         LFBFL_license_file_name2+=".${LFBFL_dest}"
294         prepare_filled_license_file_block "${LFBFL_license_file_name2}"
295     fi
296     find . -type f -name "*.${LFBFL_key}" -printf '%P\n' \
297         | relevant_find \
298         | while read -r LFBFL_file_name;
299     do
300         is_subfile "${LFBFL_file_name}" \
301             "${LFBFL_license_file_name}.temp"
302         LFBFL_not_subfile=$?
303         LFBFL_not_subfile2=1
304         if [[ -n "${LFBFL_license2}" ]]; then
305             is_subfile "${LFBFL_file_name}" \
306                 "${LFBFL_license_file_name2}.temp"
307             LFBFL_not_subfile2=$?
308         fi
309         if [[ LFBFL_not_subfile -ge 1 && LFBFL_not_subfile2 -ge 1 ]]; then
310             echo "File ${LFBFL_file_name} has no/wrong license header?"
311             fi
312         done
313         rm "${LFBFL_license_file_name}.temp"

```

```

315     if [[ -n "${LFBFL_license2}" ]]; then
316         rm "${LFBFL_license_file_name2}.temp"
317     fi
318 done
319
320 prepare_filled_license_file_line(){
321     # $1=LFBFL_license_file_name
322     sed -e "s/@repository_name@/${LFBFL_repository_name}/g" \
323           -e "s/@copyright_string@/${LFBFL_copyright_string}/g" \
324           "$1.tpl"\ \
325         > "$1.temp"
326     overwrite_if_not_equal "$1" "$1.temp"
327     declare -r LFBFL_result=$?
328     if [[ LFBFL_verbose -eq 1 ]]; then
329         if [[ LFBFL_result -eq 1 ]]; then
330             echo "License template $1 updated"
331         elif [[ LFBFL_result -eq 2 ]]; then
332             echo "License template $1 created"
333         fi
334     fi
335 }
336
337 declare -Ar LFBFL_all_line_comment_languages=(
338     ["sh"]="sh"
339     ["tex"]="tex"
340 )
341
342 for LFBFL_key in "${!LFBFL_all_line_comment_languages[@]}"; do
343     LFBFL_dest=${LFBFL_all_line_comment_languages[$LFBFL_key]}
344     LFBFL_license_file_name="${LFBFL_license_prefix2}.${LFBFL_dest}"
345     prepare_filled_license_file_line "${LFBFL_license_file_name}"
346     if [[ -n "${LFBFL_license2}" ]]; then
347         LFBFL_license_file_name2="${LFBFL_license_prefix3}"
348         LFBFL_license_file_name2+=".${LFBFL_dest}"
349         prepare_filled_license_file_line "${LFBFL_license_file_name2}"
350     fi
351     find . -type f -name "*.${LFBFL_key}" -printf '%P\n' \
352         | relevant_find\
353         | while read -r LFBFL_file_name;
354     do
355         is_subfile "${LFBFL_file_name}" "${LFBFL_license_file_name}"
356         LFBFL_not_subfile=$?
357         LFBFL_not_subfile2=1
358         if [[ -n "${LFBFL_license2}" ]]; then
359             is_subfile "${LFBFL_file_name}" "${LFBFL_license_file_name2}"
360             LFBFL_not_subfile2=$?

```

```

361         fi
362         if [[ LFBFL_not_subfile -ge 1 && LFBFL_not_subfile2 -ge 1 ]];
363         then
364             echo "File ${LFBFL_file_name} has no/wrong license header?"
365             fi
366         done
367     done
368
369     declare -i LFBFL_popd_result
370     popd || {
371         LFBFL_popd_result=$?
372         echo "build_licenses_templates no popd"
373         # shellcheck disable=SC2248
374         return ${LFBFL_popd_result}
375     }
376 }
377
378 build_licenses_templates "$@"

```

2.14 build_and_checks_dependencies/licenses_templates/license_file_header_AGPLv3+.tpl

```

1 This file is part of @repository_name@ library.
2
3 @repository_name@ is free software:
4 you can redistribute it and/or modify it under the terms
5 of the GNU Affero General Public License
6 as published by the Free Software Foundation,
7 either version 3 of the License,
8 or (at your option) any later version.
9
10 @repository_name@ is distributed in the hope
11 that it will be useful,
12 but WITHOUT ANY WARRANTY;
13 without even the implied warranty of
14 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
15 See the GNU Affero General Public License for more details.
16
17 You should have received a copy of
18 the GNU Affero General Public License
19 along with @repository_name@.
20 If not, see <https://www.gnu.org/licenses/>.
21
22 ©Copyright @copyright_string@

```

2.15 build_and_checks_dependencies/licenses_templates/license_file_header_GPLv3+.tpl

```
1 This file is part of @repository_name@ library.  
2  
3 @repository_name@ is free software:  
4 you can redistribute it and/or modify it under the terms  
5 of the GNU General Public License  
6 as published by the Free Software Foundation,  
7 either version 3 of the License,  
8 or (at your option) any later version.  
9  
10 @repository_name@ is distributed in the hope  
11 that it will be useful,  
12 but WITHOUT ANY WARRANTY;  
13 without even the implied warranty of  
14 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  
15 See the GNU General Public License for more details.  
16  
17 You should have received a copy of  
18 the GNU General Public License  
19 along with @repository_name@.  
20 If not, see <https://www.gnu.org/licenses/>.  
21  
22 ©Copyright @copyright_string@
```

2.16 build_and_checks_dependencies/licenses_templates/license_file_header_LGPLv3+.tpl

```
1 This file is part of @repository_name@ library.  
2  
3 @repository_name@ is free software:  
4 you can redistribute it and/or modify it under the terms  
5 of the GNU Lesser General Public License  
6 as published by the Free Software Foundation,  
7 either version 3 of the License,  
8 or (at your option) any later version.  
9  
10 @repository_name@ is distributed in the hope  
11 that it will be useful,  
12 but WITHOUT ANY WARRANTY;  
13 without even the implied warranty of  
14 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  
15 See the GNU Lesser General Public License for more details.  
16  
17 You should have received a copy of  
18 the GNU Lesser General Public License  
19 along with @repository_name@.
```

```

20  If not, see <https://www.gnu.org/licenses/>.
21
22  ©Copyright @copyright_string@

```

2.17 build_and_checks_dependencies/lines_counts.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "lines_counts.sh" to
25 # "lines_counts.libr.sh".
26
27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=get_common_text_glob_patterns.libr.sh
29 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
30 # shellcheck source=lines_filters.libr.sh
31 source "./${LFBFL_subdir}/lines_filters.libr.sh"
32
33 all_code_lines(){
34     if [[ ! -o pipefail ]]; then
35         set -o pipefail
36         trap 'set +o pipefail' RETURN
37     fi
38     [[ "$*" != *--verbose* ]]
39     declare -ir LFBFL_verbose=$?
40     get_COMMON_TEXT_FILES_GLOB_PATTERNS

```

```

41 local LFBFL_pattern
42 for LFBFL_pattern in "${COMMON_TEXT_FILES_GLOB_PATTERNS[@]}"; do
43     [[ LFBFL_verbose -eq 1 ]]\
44     && echo "Iterating on pattern: ${LFBFL_pattern}"
45     find . -type f -name "${LFBFL_pattern}" -printf '%P\n'\
46     | xargs grep -H -v 'a(?!a)a'
47 done
48 }
49
50 all_self_code_lines(){
51     if [[ ! -o pipefail ]]; then
52         set -o pipefail
53         trap 'set +o pipefail' RETURN
54     fi
55     # below and another lines, see SC2145 false positive
56     all_code_lines "$@" \
57     | relevant_grep\
58     | not_license_grep\
59     | not_main_tex_grep\
60     | not_main_html_grep
61 }
62
63 all_self_empty_code_lines(){
64     if [[ ! -o pipefail ]]; then
65         set -o pipefail
66         trap 'set +o pipefail' RETURN
67     fi
68     all_self_code_lines "$@" \
69     | empty_lines_after_file_name
70 }
71
72 all_self_not_empty_code_lines(){
73     if [[ ! -o pipefail ]]; then
74         set -o pipefail
75         trap 'set +o pipefail' RETURN
76     fi
77     all_self_code_lines "$@" \
78     | not_empty_lines_after_file_name
79 }
80
81 code_lines_count_all(){
82     if [[ ! -o pipefail ]]; then
83         set -o pipefail
84         trap 'set +o pipefail' RETURN
85     fi
86     all_self_code_lines "$@" \

```

```

87         | wc -l
88     }
89
90     code_lines_count_empty() {
91         if [[ ! -o pipefail ]]; then
92             set -o pipefail
93             trap 'set +o pipefail' RETURN
94         fi
95         all_self_empty_code_lines "$@" \
96             | wc -l
97     }
98
99     code_lines_count_not_empty() {
100        if [[ ! -o pipefail ]]; then
101            set -o pipefail
102            trap 'set +o pipefail' RETURN
103        fi
104        all_self_not_empty_code_lines "$@" \
105            | wc -l
106    }

```

2.18 build_and_checks_dependencies/lines_filters.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet

```

```

24 # This file was renamed from "lines_filters.sh" to
25 # "lines_filters.libr.sh".
26
27 # The goal of this file is to have some common line filters
28 # for bash processing and pipes.
29 # As is the case for other .sh files in this repository,
30 # it can be copied somewhere in your home directory
31 # and sourced in your .bashrc.
32
33 ll_wc() {
34     # current options to wc:
35     # -c --bytes
36     # -m --chars
37     # -l --lines
38     # --files0-from # Incompatible with cat,
39     #                 # another thing intentional by ?
40     #                 # I don't see the meaning of "files0".
41     # -L --max-line-length
42     # -w --words
43     # --help
44     # --version
45     # New option by Laurent Lyraudet (and probably other before):
46     # -n --no-filenames display only number(s)
47     # I think I did that the first time many years ago (Teliae ?).
48     # Since then, they added it as ignored... (-n)
49     if [[ ! -o pipefail ]]; then
50         set -o pipefail
51         trap 'set +o pipefail' RETURN
52     fi
53     declare -a ll_wc_var_args=()
54     declare -i ll_wc_var_number_only=0
55     declare -i ll_wc_var_i=0
56     for ll_wc_var_arg in "$@"; do
57         # echo "$ll_wc_var_arg"
58         if [[ "${ll_wc_var_arg}" == "-n" ]]; then
59             ll_wc_var_number_only=1
60         elif [[ "${ll_wc_var_arg}" == "--no-filenames" ]]; then
61             ll_wc_var_number_only=1
62         elif [[ ll_wc_var_number_only -eq 0 ]]; then
63             ll_wc_var_args[ll_wc_var_i]="${ll_wc_var_arg}"
64             ll_wc_var_i=$((ll_wc_var_i + 1))
65         elif [[ "${ll_wc_var_arg}" =~ -* ]]; then
66             ll_wc_var_args[ll_wc_var_i]="${ll_wc_var_arg}"
67             ll_wc_var_i=$((ll_wc_var_i + 1))
68         fi
69     done

```

```

70  # typeset -p ll_wc_var_args
71  # echo "$ll_wc_var_number_only"
72  if [[ ll_wc_var_number_only -gt 0 ]]; then
73      # Too simple code, only the base use case is handled now.
74      # Convenient but incomplete.
75      # With more time,
76      # the simpler would be to submit a patch to GNU coreutils,
77      # unless blockade from assholes.
78      # Otherwise, it may be more simply handled in PHP or Python
79      # (argparse, etc.).
80      # You'll need to start by improving cat first,
81      # with --files0-from handling.
82      # wc does not display file_name when stream...
83      # hence cat of last arg "${!#}"
84      cat "${!#}"\
85          | wc "${ll_wc_var_args[@]}"

86  else
87      # echo "normal"
88      wc "${ll_wc_var_args[@]}"
89  fi
90 }

91
92  # I submitted a patch for wc :).
93  # echo "https://domain/list-path/2024-05/msg00013.html" | \
94  #     sed -e 's|domain|lists.gnu.org|' \
95  #         -e 's|list-path|archive/html/bug-coreutils|'
96  # I found another way to have a long URL fit on 70 characters.
97  # If it is merged, the following would be better:
98  # If --no-filenames option available,
99  # we redefine ll_wc command as an alias.
100 wc_help_text=$(wc --help)
101 if [[ "${wc_help_text}" == *"--no-filenames"* ]]; then
102     alias ll_wc='wc'
103 fi
104
105 # Trace avant le (2e ?) renommage
106 # de wget_sha512.sh à wget_sha512.libr.sh.
107 # $ ll_wc wget_sha512.sh
108 # 41 189 1224 wget_sha512.sh
109 # $ ll_wc -n wget_sha512.sh
110 # 41 189 1224
111 # $ ll_wc -l wget_sha512.sh
112 # 41 wget_sha512.sh
113 # $ ll_wc -l -n wget_sha512.sh
114 # 41
115

```

```

116  in_place_grep() {
117      declare -r LFBFL_temp="${!#}.in_place_grep.temp"
118      grep "$@" \
119          > "${LFBFL_temp}"
120      declare -ir LFBFL_lines_before=$(ll_wc -l -n "${!#}")
121      declare -ir LFBFL_lines_after=$(ll_wc -l -n "${LFBFL_temp}")
122      # echo "$LFBFL_lines_before"
123      # echo "$LFBFL_lines_after"
124      if [[ "${LFBFL_lines_before}" == "${LFBFL_lines_after}" ]]; then
125          rm "${LFBFL_temp}"
126          return
127      fi
128      mv "${LFBFL_temp}" "$!"
129  }
130
131  grep_variable() {
132      # $1=$file
133      # $2=$variable_name
134      # Options :
135      #   --replace-line-returns-by=""
136      #   --result-variable-prefix="LFBFL_" for example
137      declare -r LFBFL_regexp="(?:^$2=).*$"
138      local LFBFL_arg
139      # echo $LFBFL_regexp
140      local LFBFL_variable_value
141      if [[ "$*" == *--replace-line-returns-by=* ]]; then
142          local LFBFL_s_replace=""
143          for LFBFL_arg in "$@"; do
144              if [[ "${LFBFL_arg}" == --replace-line-returns-by=* ]]; then
145                  LFBFL_s_replace=${LFBFL_arg##--replace-line-returns-by=}
146                  break
147              fi
148          done
149          LFBFL_variable_value=$((
150              grep -oP "${LFBFL_regexp}" "$1" \
151              | sed -Ez -e 's/\n/'"${LFBFL_s_replace}"'/Mg'
152          )
153          # We remove the trailing replace string,
154          # since the intent is only to replace the intermediate \n.
155          if [[ -n "${LFBFL_s_replace}" ]]; then
156              LFBFL_variable_value=${LFBFL_variable_value%${LFBFL_s_replace}}
157          fi
158      else
159          LFBFL_variable_value=$(grep -oP "${LFBFL_regexp}" "$1")
160      fi
161      local LFBFL_prefix=""

```

```

162     for LFBFL_arg in "$@"; do
163         if [[ "${LFBFL_arg}" == --result-variable-prefix=* ]]; then
164             LFBFL_prefix=${LFBFL_arg#--result-variable-prefix=}
165             break
166         fi
167     done
168     declare -r LFBFL_variable_name="${LFBFL_prefix}$_2"
169     # echo "${LFBFL_variable_name} : ${LFBFL_variable_value}"
170     # The declare line may be dropped by a local or declare
171     # in outer scope.
172     # declare -g "${LFBFL_variable_name}"
173     # So the assignment must be done on another line.
174     printf -v "${LFBFL_variable_name}" "%s" "${LFBFL_variable_value}"
175     # echo "${!LFBFL_variable_name}"
176 }
177
178 empty_lines() {
179     grep '^$'
180 }
181
182 not_empty_lines() {
183     grep -v '^$'
184 }
185
186 space_starting_lines() {
187     grep '^[ ]'
188 }
189
190 not_space_starting_lines() {
191     grep '^[^ ]'
192 }
193
194 empty_lines_after_file_name() {
195     grep '^[:]\+:$'
196 }
197
198 not_empty_lines_after_file_name() {
199     grep -v '^[:]\+:$'
200 }
201
202 space_starting_lines_after_file_name() {
203     grep '^[:]\+: '
204 }
205
206 not_space_starting_lines_after_file_name() {
207     grep '^[:]\+:[^ ]'

```

```

208     }
209
210     not_JS_dependencies_find() {
211         grep -vE -e "^( |/ )node_modules/" \
212             -e "^( |/ )package-lock\.json$"
213     }
214
215     not_JS_dependencies_grep() {
216         grep -vE -e "^( [^:]++ / )?node_modules/" \
217             -e "^( [^:]++ / )?package-lock\.json:"
218     }
219
220     not_dependencies_find() {
221         not_JS_dependencies_find
222     }
223
224     not_dependencies_grep() {
225         not_JS_dependencies_grep
226     }
227
228     not_python_cache_find() {
229         grep -vE -e "^( |/ )__pycache__/" \
230             -e "^( |/ )\.mypy_cache/" \
231             -e "^( |/ )\.ruff_cache/"
232     }
233
234     not_python_cache_grep() {
235         grep -vE -e "^( [^:]++ / )?__pycache__/" \
236             -e "^( [^:]++ / )?\.\mypy_cache/" \
237             -e "^( [^:]++ / )?\.\ruff_cache/"
238     }
239
240     not_cache_find() {
241         not_python_cache_find
242     }
243
244     not_cache_grep() {
245         not_python_cache_grep
246     }
247
248     not_git_find() {
249         grep -vE "^( |/ )\.git/"
250     }
251
252     not_git_grep() {
253         grep -vE "^( [^:]++ / )?\.\git/"

```

```

254     }
255
256     not_archive_find() {
257         grep -vE "(\.gz|\.rar|\.tar|\.tgz|\.whl)$"
258     }
259
260     not_archive_grep() {
261         grep -vE "^[^:]*(\.gz|\.rar|\.tar|\.tgz|\.whl):"
262     }
263
264     not_license_find() {
265         grep -vE -e "(^|/)COPYING$" \
266             -e "^(^|/)COPYING\.LESSER$"
267     }
268
269     not_license_grep() {
270         grep -vE -e "^([:]+) ?COPYING:" \
271             -e "^([:]+) ?COPYING\.LESSER:"
272     }
273
274     not_main_tex_find() {
275         local LFBFL_repository_name=""
276         grep_variable build_and_checks_variables/repository_data.txt \
277             repository_name \
278             --result-variable-prefix="LFBFL_" \
279             --replace-line-returns-by=""
280         grep -vE "^(|/)${LFBFL_repository_name}\.tex$"
281     }
282
283     not_main_tex_grep() {
284         local LFBFL_repository_name=""
285         grep_variable build_and_checks_variables/repository_data.txt \
286             repository_name \
287             --result-variable-prefix="LFBFL_" \
288             --replace-line-returns-by=""
289         grep -vE "^([:]+) ?${LFBFL_repository_name}\.tex:"
290     }
291
292     not_main_html_find() {
293         local LFBFL_repository_name=""
294         grep_variable build_and_checks_variables/repository_data.txt \
295             repository_name \
296             --result-variable-prefix="LFBFL_" \
297             --replace-line-returns-by=""
298         grep -vE "^(|/)${LFBFL_repository_name}\.html$"
299     }

```

```

300
301 not_main_html_grep() {
302     local LFBFL_repository_name=""
303     grep_variable build_and_checks_variables/repository_data.txt \
304         repository_name \
305         --result-variable-prefix="LFBFL_" \
306         --replace-line-returns-by="" \
307     grep -vE "^([:]+/)?${LFBFL_repository_name}\.html:" \
308 }
309
310 not_temp_file_find() {
311     grep -vE "^(|/)build_and_checks_variables/temp/.*$"
312 }
313
314 not_temp_file_grep() {
315     grep -vE "^([:]+/)?build_and_checks_variables/temp/[^\n]*:"
316 }
317
318 relevant_find() {
319     if [[ ! -o pipefail ]]; then
320         set -o pipefail
321         trap 'set +o pipefail' RETURN
322     fi
323     not_dependencies_find \
324         | not_cache_find \
325         | not_git_find \
326         | not_archive_find \
327         | not_temp_file_find
328 }
329
330 relevant_grep() {
331     if [[ ! -o pipefail ]]; then
332         set -o pipefail
333         trap 'set +o pipefail' RETURN
334     fi
335     not_dependencies_grep \
336         | not_cache_grep \
337         | not_git_grep \
338         | not_archive_grep \
339         | not_temp_file_grep
340 }

```

2.19 build_and_checks_dependencies/lines_maps.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.

```

```

3  #
4  # DevOrSysAdminScripts is free software:
5  # you can redistribute it and/or modify it under the terms
6  # of the GNU General Public License
7  # as published by the Free Software Foundation,
8  # either version 3 of the License,
9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "lines_maps.sh" to "lines_maps.libr.sh".
25 #
26 # https://en.wikibooks.org/wiki/Unicode/Character_reference/2000–2FFF
27 # Some invisible unicode characters that are a problem:
28 # 2000–200F 2028–202F 205F 2060–206F
29 # [
30 #   '2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007',
31 #   '2008', '2009', '200A', '200B', '200C', '200D', '200E', '200F',
32 #   '2028', '2029', '202A', '202B', '202C', '202D', '202E', '202F',
33 #   '205F',
34 #   '2060', '2061', '2062', '2063', '2064', '2065', '2066', '2067',
35 #   '2068', '2069', '206A', '206B', '206C', '206D', '206E', '206F',
36 # ]
37 replace_non_ascii_spaces() {
38     sed -e 's/\xc2\x{a0}/ /g' \
39           -e 's/\xe2\x{80}\x{80}/ /g' \
40           -e 's/\xe2\x{80}\x{81}/ /g' \
41           -e 's/\xe2\x{80}\x{82}/ /g' \
42           -e 's/\xe2\x{80}\x{83}/ /g' \
43           -e 's/\xe2\x{80}\x{84}/ /g' \
44           -e 's/\xe2\x{80}\x{85}/ /g' \
45           -e 's/\xe2\x{80}\x{86}/ /g' \
46           -e 's/\xe2\x{80}\x{87}/ /g' \
47           -e 's/\xe2\x{80}\x{88}/ /g' \
48           -e 's/\xe2\x{80}\x{89}/ /g'

```

```

49      -e 's/>\xe2\x80\x8a/ /g' \
50      -e 's/>\xe2\x80\x8b/ /g' \
51      -e 's/>\xe2\x80\x8c/ /g' \
52      -e 's/>\xe2\x80\x8d/ /g' \
53      -e 's/>\xe2\x80\x8e/ /g' \
54      -e 's/>\xe2\x80\x8f/ /g' \
55      -e 's/>\xe2\x80\xaa/ /g' \
56      -e 's/>\xe2\x80\xaa9/ /g' \
57      -e 's/>\xe2\x80\xaa/ /g' \
58      -e 's/>\xe2\x80\xab/ /g' \
59      -e 's/>\xe2\x80\xac/ /g' \
60      -e 's/>\xe2\x80\xad/ /g' \
61      -e 's/>\xe2\x80\xae/ /g' \
62      -e 's/>\xe2\x80\xaf/ /g' \
63      -e 's/>\xe2\x81\x9f/ /g' \
64      -e 's/>\xe2\x81\xaa0/ /g' \
65      -e 's/>\xe2\x81\xaa1/ /g' \
66      -e 's/>\xe2\x81\xaa2/ /g' \
67      -e 's/>\xe2\x81\xaa3/ /g' \
68      -e 's/>\xe2\x81\xaa4/ /g' \
69      -e 's/>\xe2\x81\xaa5/ /g' \
70      -e 's/>\xe2\x81\xaa6/ /g' \
71      -e 's/>\xe2\x81\xaa7/ /g' \
72      -e 's/>\xe2\x81\xaa8/ /g' \
73      -e 's/>\xe2\x81\xaa9/ /g' \
74      -e 's/>\xe2\x81\xaa/ /g' \
75      -e 's/>\xe2\x81\xab/ /g' \
76      -e 's/>\xe2\x81\xac/ /g' \
77      -e 's/>\xe2\x81\xad/ /g' \
78      -e 's/>\xe2\x81\xae/ /g' \
79      -e 's/>\xe2\x81\xaf/ /g'
80  }

```

2.20 build_and_checks_dependencies/overwrite_if_not_equal.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope

```

```

12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "overwrite_if_not_equal.sh"
25 # to "overwrite_if_not_equal.libr.sh".
26
27 overwrite_if_not_equal() {
28     # $1=$target_file_name
29     # $2=$temp_file_name
30     # $3=$keep_temp
31     # $4=$tree_mode : when the two files are the output of tree command,
32     # we ignore if all the differences are only on directories.
33     # Return 0 target was the same
34     #      1 target was different
35     #      2 target did not already exist
36     if [[ ! -f "$1" ]]; then
37         if [[ -n "$3" ]]; then
38             cp -p "$2" "$1"
39         else
40             mv "$2" "$1"
41         fi
42         return 2
43     fi
44     declare -i LFBFL_is_equal
45     if [[ -n "$4" ]]; then
46         # Not setting pipefail since the result would still be incorrect.
47         # shellcheck disable=SC2312
48         diff "$1" "$2" \
49             | grep -E "^(>|<)" \
50             | grep -v "/$"
51         LFBFL_is_equal=${PIPESTATUS[2]}
52     else
53         diff -q "$1" "$2"
54         LFBFL_is_equal=1-$?
55     fi
56     if [[ LFBFL_is_equal -eq 1 ]]; then
57         if [[ -z "$3" ]]; then

```

```

58         rm "$2"
59     fi
60     return 0
61 fi
62 if [[ -n "$3" ]]; then
63     cp -p "$2" "$1"
64 else
65     mv "$2" "$1"
66 fi
67 return 1
68 }

```

2.21 build_and_checks_dependencies/python_black_complement.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "python_black_complement.sh"
25 # to "python_black_complement.libr.sh".
26 #
27 # It was rejected to enhance black with this currently.
28 # https://github.com/psf/black/issues/2370
29 # And moreover, at some point, I saw the reverse rule
30 # in black or pylint to add an empty line...
31 # This regexp may give false positives,
32 # but that's not the end of the world.

```

```

33  check_no_empty_line_after_python_function_docstring() {
34      echo "Checking empty lines after Python function docstrings"
35      pcregrep -M '$def [^"]*""([^\"]|"(?!"))*""\n\n(?!`def)`\
36          -- **/*.py
37  }
38
39  python_black_complement() {
40      check_no_empty_line_after_python_function_docstring
41  }

```

2.22 build_and_checks_dependencies/python_isort_complement.libr.sh

```

1  #!/usr/bin/env bash
2  # This file is part of DevOrSysAdminScripts library.
3  #
4  # DevOrSysAdminScripts is free software:
5  # you can redistribute it and/or modify it under the terms
6  # of the GNU General Public License
7  # as published by the Free Software Foundation,
8  # either version 3 of the License,
9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "python_isort_complement.sh"
25 # to "python_isort_complement.libr.sh".
26
27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=lines_filters.libr.sh
29 source "./${LFBFL_subdir}/lines_filters.libr.sh"
30 # shellcheck source=overwrite_if_not_equal.libr.sh
31 source "./${LFBFL_subdir}/overwrite_if_not_equal.libr.sh"
32
33 check_collections_abc_place(){
34     echo "Checking import of _collections_abc is at the right place"

```

```

35     if [[ ! -o pipefail ]]; then
36         set -o pipefail
37         trap 'set +o pipefail' RETURN
38     fi
39     declare -r LFBFL_temp=".check_collections_abc_place.temp"
40     local LFBFL_file_name
41     find . -name ".*.py" \
42         | relevant_find \
43         | while read -r LFBFL_file_name;
44     do
45         [[ -f "${LFBFL_file_name}" ]] || continue
46         sed -Ez 's/\n(\nfrom _collections_abc[^\\n]*)/\1\n/Mg' \
47             "${LFBFL_file_name}" \
48             > "${LFBFL_file_name}${LFBFL_temp}"
49         overwrite_if_not_equal "${LFBFL_file_name}" \
50             "${LFBFL_file_name}${LFBFL_temp}"
51     done
52 }
53
54 python_isort_complement() {
55     check_collections_abc_place
56 }
```

2.23 build_and_checks_dependencies/split_score.exec.php

```

1 #!/usr/bin/php
2 <?php
3 /*
4 This file is part of DevOrSysAdminScripts library.
5
6 DevOrSysAdminScripts is free software:
7 you can redistribute it and/or modify it under the terms
8 of the GNU Lesser General Public License
9 as published by the Free Software Foundation,
10 either version 3 of the License,
11 or (at your option) any later version.
12
13 DevOrSysAdminScripts is distributed in the hope
14 that it will be useful,
15 but WITHOUT ANY WARRANTY;
16 without even the implied warranty of
17 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
18 See the GNU Lesser General Public License for more details.
19
20 You should have received a copy of
21 the GNU Lesser General Public License
```

```

22 along with DevOrSysAdminScripts.
23 If not, see <https://www.gnu.org/licenses/>.
24
25 ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
26 2026-01-04T13:38+01:00: This file was renamed from
27 "split_line_at_most.exec.php"
28 to
29 "split_score.exec.php".
30 This file was renamed from
31 "call_split_score_after_before_or_before_after.exec.php" to
32 "split_line_at_most.exec.php".
33 This file was renamed from
34 "call_split_score_after_before_or_before_after.php" to
35 "call_split_score_after_before_or_before_after.exec.php".
36 */
37
38 require_once('./build_and_checks_dependencies/split_score.libr.php');
39
40 /*
41 php -r\
42 '$require_once("./build_and_checks_dependencies/' \
43 '$split_score.libr.php');' \
44 '$echo(generate_split_score' \
45 '$(true, 70, ["/\'])(/\'\', 60, true));'
46 */
47
48 // The first cast "(int) generate_split_score" is to avoid false
49 // positive errors on possible XSS if output is unescaped.
50 echo(
51     (int) generate_split_score(
52         (bool) $argv[1], (int) $argv[2], explode(',', $argv[3]),
53         ($argv[4], (int) $argv[5], (bool) $argv[6])
54     );

```

2.24 build_and_checks_dependencies/split_score.libr.php

```

1 <?php
2 /*
3 This file is part of DevOrSysAdminScripts library.
4
5 DevOrSysAdminScripts is free software:
6 you can redistribute it and/or modify it under the terms
7 of the GNU Lesser General Public License
8 as published by the Free Software Foundation,
9 either version 3 of the License,
10 or (at your option) any later version.

```

```

11
12 DevOrSysAdminScripts is distributed in the hope
13 that it will be useful,
14 but WITHOUT ANY WARRANTY;
15 without even the implied warranty of
16 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
17 See the GNU Lesser General Public License for more details.
18
19 You should have received a copy of
20 the GNU Lesser General Public License
21 along with DevOrSysAdminScripts.
22 If not, see <https://www.gnu.org/licenses/>.
23
24 ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
25 2026-01-04T13:38+01:00: This file was renamed from
26 "split_line_at_most.libr.php"
27 to
28 "split_score.libr.php".
29 This file was renamed from "split_line_at_most.php" to
30 "split_line_at_most.libr.php".
31 */
32
33 function generate_split_score(
34     $b_after_before,
35     $i_max_length,
36     $a_delimiter_strings_domain,
37 ) {
38     return function(
39         $s_delimiter_string,
40         $i_cut_position,
41         $b_is_cut_after,
42     ) use ($b_after_before, $i_max_length, $a_delimiter_strings_domain) {
43         if (
44             !in_array(
45                 $s_delimiter_string, $a_delimiter_strings_domain, true
46             )
47         ) {
48             // I do not handle regexps for the moment.
49             // I'm too brainfucked. I just want to finish this feature fast
50             // for HTML and Tex/PDF listings generation with HTML and Tex on
51             // at most 70 characters.
52             return 0;
53         }
54         if ($b_is_cut_after === $b_after_before) {
55             // When $b_after_before,
56             // always way better to cut after '/' than before it.

```

```

57     // When !$b_after_before (aka before_after),
58     // always way better to cut before '/' than after it.
59     return 1 + $i_max_length + $i_cut_position;
60   }
61   return 1 + $i_cut_position;
62 }
63 }
```

2.25 build_and_checks_dependencies/strings_functions.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # I think I created a file strings_functions.sh or renamed to
25 # strings_functions.sh at some point, before committing.
26 # Hence, someone must have changed or reversed the name of this file
27 # to string_functions.sh at some point.
28 # This file was renamed from "string_functions.sh" to
29 # "strings_functions.libr.sh".
30 #
31 LFBFL_subdir="build_and_checks_dependencies"
32 # shellcheck source=get_common_text_glob_patterns.libr.sh
33 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
34 # shellcheck source=comparisons.libr.sh
35 source "./${LFBFL_subdir}/comparisons.libr.sh"
```

```

37  split_line_at() {
38      # $1=$line
39      # $2=$position
40      declare -g split_line_at_result_beginning="${1:0:$2}"
41      declare -g split_line_at_result_end="${1:$2}"
42      # echo "$split_line_at_result_beginning"
43      # echo "$split_line_at_result_end"
44  }
45
46  split_score_simple() {
47      # $1=$after_before
48      # $2=$delimiters_strings_domain concatenated characters/delimiters
49      # $3=$delimiter_string a single character
50      # $4=$is_cut_after
51      declare -gi split_score_result
52      if [[ "$2" != *$3* ]]; then
53          split_score_result=0
54          return
55      fi
56      if [[ "$4" == "$1" ]]; then
57          split_score_result=2
58          return
59      fi
60      split_score_result=1
61  }
62
63  get_split_score_simple() {
64      # $1=$after_before
65      # $2=$max_length
66      # $3=$delimiters_strings_domain
67      declare -g get_split_score_result="split_score_simple $1 '$3'"
68      declare -gi get_split_score_result2
69      get_split_score_result2=$((7+$1*4))
70      # if [[ $1 -eq 1 ]]; then
71      #     get_split_score_result2=7
72      # else
73      #     get_split_score_result2=11
74      # fi
75  }
76
77  get_split_score_exec() {
78      declare -g SPLIT_SCORE_EXEC="./build_and_checks_dependencies"
79      SPLIT_SCORE_EXEC+="/split_score.exec.php"
80  }
81
82  split_score() {

```

```

83     # $1=$after_before
84     # $2=$max_length
85     # $3=$delimiters_strings_domain concatenated characters/delimiters
86     # $4=$delimiter_string a single character
87     # $5=$cut_position
88     # $6=$is_cut_after
89     get_split_score_exec
90     declare -r LFBFL_command="${SPLIT_SCORE_EXEC} $1 $2 '$3' '$4' $5 $6"
91     declare -gi split_score_result
92     split_score_result=$(eval "${LFBFL_command}")
93 }
94
95 # This function is currently not called,
96 # hence we need that shellcheck disable.
97 # shellcheck disable=SC2034
98 get_split_score() {
99     # $1=$after_before
100    # $1=$max_length
101    # $2=$delimiters_strings_domain
102    declare -g get_split_score_result="split_score $1 $2 '$3'"
103    declare -gi get_split_score_result2
104    get_split_score_result2=$((5+$1*4))
105 }
106
107 split_line_at_most() {
108     # $1=$line
109     # $2=$max_length of beginning result string
110     # $3=$split_score_command
111     # negative score means do not consider as a possible split
112     # $4=$split_score_command_properties :
113     # These properties are flags.
114     # If someday some property is not a flag,
115     # then create split_score_command_properties2.
116     # - null or 0 no property;
117     # - 1 split score doesn't depend on which delimiter matches;
118     # - 2 split score doesn't depend on the current position;
119     # - 4 any split score after is always larger
120     # than any split score before;
121     # - 8 any split score before is always larger
122     # than any split score after;
123     # $5=$forbidden_previous_character
124     # $6=$split_score_command_delimiter_strings_domain
125     # optimisation if small domain
126     # - this argument can be "null", then we will use $3 only instead,
127     # in that case, only mono-character strings are considered,
128     # - this argument can be an array of characters,

```

```

129      # optimize accordingly,
130      # - this argument can be an array of strings,
131      # optimize accordingly,
132      # - this argument can be an array of regexps,
133      # optimize accordingly.
134      # Not sure an array of regexps makes more sense than a single
135      # one. Maybe they could be both cases with corresponding
136      # optimizations. Same for a single character or substring.
137      #
138      # Note that we're always looking for the last position to split
139      # among maximum score;
140      # if some position gets as score from 2 delimiters,
141      # its score is always the maximum of the two corresponding scores.
142      declare -g split_line_at_most_result_start
143      declare -g split_line_at_most_result_end
144      declare -A LFBFL_positions
145      # echo "$1 $2 $3 $4"
146      # For my use case in bash scripts, I will need only an array of
147      # characters. See get_split_score_after_before().
148      if [[ -n "$6" ]]; then
149          echo "split_line_at_most() \$4 NOT IMPLEMENTED YET"
150      fi
151      if [[ $2 -ge ${#1} ]]; then
152          split_line_at_most_result_start=$1
153          split_line_at_most_result_end=""
154          return
155      fi
156      declare -r LFBFL_sort_command="sort --numeric-sort"
157      declare -ir LFBFL_length_minus_1=$(( ${#1} - 1 ))
158      declare -ir LFBFL_i_max=$(( ${#1} - 1 ))
159      min "${LFBFL_sort_command}" "${LFBFL_length_minus_1}" "$2"
160      )
161      declare -i LFBFL_i
162      declare -i LFBFL_j
163      local LFBFL_current_char
164
165      # At the beginning, each position has score 0.
166      for ((LFBFL_i=0; LFBFL_i<=LFBFL_i_max; ++LFBFL_i)) do
167          LFBFL_positions["${LFBFL_i}"]="0"
168      done
169      LFBFL_j=$((LFBFL_i+1))
170      LFBFL_positions["${LFBFL_j}"]="0"
171
172      # If we have a forbidden previous character,
173      # we mark the forbidden positions with -1.
174      if [[ -n "$5" ]]; then

```

```

175  for ((LFBFL_i=0; LFBFL_i<LFBFL_i_max; ++LFBFL_i)) do
176      LFBFL_j=$((LFBFL_i+1))
177      LFBFL_current_char="${1:$LFBFL_i:1}"
178      if [[ ${LFBFL_current_char} == "$5" ]]; then
179          LFBFL_positions["${LFBFL_j}"]="-1"
180      fi
181      done
182  fi
183
184  if [[ "$4" == "7" ]]; then
185      for ((LFBFL_j=LFBFL_i_max; LFBFL_j>0; --LFBFL_j)) do
186          LFBFL_i=$((LFBFL_j-1))
187          LFBFL_current_char="${1:$LFBFL_i:1}"
188          eval "$3 '$LFBFL_current_char' 0"
189          # echo "${split_score_result}|${LFBFL_i}"
190          if [[ split_score_result -ge 1 ]]; then
191              # echo "${split_score_result}|${LFBFL_i}"
192              if [[ ${LFBFL_positions["${LFBFL_i}"]} != "-1" ]]; then
193                  LFBFL_positions["${LFBFL_i}"]=$(((
194                      max ${LFBFL_sort_command} \\
195                      ${LFBFL_positions["${LFBFL_i}"]} ) \\
196                      ${split_score_result}))
197              )
198          fi
199      fi
200      eval "$3 '$LFBFL_current_char' 1"
201      # echo "${split_score_result}|${LFBFL_j}"
202      if [[ split_score_result -ge 1 ]]; then
203          # echo "${split_score_result}|${LFBFL_j}"
204          if [[ ${LFBFL_positions["${LFBFL_j}"]} != "-1" ]]; then
205              LFBFL_positions["${LFBFL_j}"]=$(((
206                  max ${LFBFL_sort_command} \\
207                  ${LFBFL_positions["${LFBFL_j}"]} ) \\
208                  ${split_score_result}))
209              )
210          break
211      fi
212  fi
213  done
214 else
215     for ((LFBFL_i=0; LFBFL_i<LFBFL_i_max; ++LFBFL_i)) do
216         LFBFL_j=$((LFBFL_i+1))
217         LFBFL_current_char="${1:$LFBFL_i:1}"
218         eval "$3 '$LFBFL_current_char' ${LFBFL_i} 0"
219         # echo "${split_score_result}|${LFBFL_i}"
220         if [[ split_score_result -ge 1 ]]; then

```

```

221     # echo "${split_score_result}|${LFBFL_i}"
222     if [[ ${LFBFL_positions["${LFBFL_i}"]} != "-1" ]]; then
223         LFBFL_positions["${LFBFL_i}"]=$(((
224             max "${LFBFL_sort_command}" \
225                 "${LFBFL_positions["${LFBFL_i}"]}") \
226             "${split_score_result}")
227         )
228     fi
229 fi
230 eval "$3'${LFBFL_current_char}' ${LFBFL_j} 1"
231 # echo "${split_score_result}|${LFBFL_j}"
232 if [[ split_score_result -ge 1 ]]; then
233     # echo "${split_score_result}|${LFBFL_j}"
234     if [[ ${LFBFL_positions["${LFBFL_j}"]} != "-1" ]]; then
235         LFBFL_positions["${LFBFL_j}"]=$(((
236             max "${LFBFL_sort_command}" \
237                 "${LFBFL_positions["${LFBFL_j}"]}") \
238             "${split_score_result}")
239         )
240     fi
241 fi
242 done
243 fi
244 declare -ir LFBFL_max_score=$(((
245     max 'sort --numeric-sort' "${LFBFL_positions[@]}"))
246 )
247 local LFBFL_value
248 declare -i LFBFL_best_position
249 for ((LFBFL_i=0; LFBFL_i<=LFBFL_i_max; ++LFBFL_i)) do
250     local LFBFL_value=${LFBFL_positions[$LFBFL_i]}
251     # echo "${LFBFL_i} ${LFBFL_value}"
252     if [[ "${LFBFL_value}" == "${LFBFL_max_score}" ]]; then
253         LFBFL_best_position=$((LFBFL_i))
254     fi
255 done
256 # shellcheck disable=SC2248
257 split_line_at "$1" ${LFBFL_best_position}
258 split_line_at_most_result_start="${split_line_at_result_beginning}"
259 split_line_at_most_result_end="${split_line_at_result_end}"
260 # echo "$split_line_at_most_result_start"
261 # echo "$split_line_at_most_result_end"
262 }
263
264 split_last_line() {
265     # $1=$new_lines
266     # $2=$prefix

```

```

267  # $3=$max_length
268  # $4=$suffix
269  # $5=$split_score_command
270  # $6=$split_score_command_properties
271  # $7=$forbidden_previous_character ('\\' usually)
272  # In general, a prefix/suffix can be a required line prefix/suffix
273  # for all final lines like a comment prefix,
274  # or it can be a continuation line prefix/suffix applied only
275  # when splitting a line in 2 lines,
276  # or it can be a prefix/suffix for the string made of all the lines.
277  # Clearly, there is no gain to couple handling a prefix/suffix for
278  # the string made of all the lines in this function.
279  # Since we add a prefix/suffix when splitting,
280  # it is easy to see that we do not need to distinguish between
281  # required line prefix/suffix and continuation line prefix/suffix.
282  # The $2 and $4 arguments are given as concatenation of the
283  # required and continuation line prefixes/suffixes as needed by the
284  # user.
285  # Thus lines arguments should be given already prefixed
286  # and suffixed.
287  # This is logic, since it would be cumbersome to handle prefixing
288  # or not already splitted lines in this function
289  # (see repeated_split_last_line).
290  # The given max_length is the goal.
291  # This function adapts the effective max position for the split by
292  # taking into account the length of the suffix.
293 if [[ ! -o pipefail ]]; then
294     set -o pipefail
295     trap 'set +o pipefail' RETURN
296 fi
297
298 declare -g split_last_line_result="$1"
299 declare -ir LFBFL_max_length_plus=$((3 + 1))
300 declare -ir LFBFL_length2=$((3 - ${#4}))
301 declare -r LFBFL-regexp='.\{${LFBFL_max_length_plus}\}'\}$
302 if
303     echo "$1" \
304         | sed -e 's/\n/g' \
305         | grep -q "${LFBFL_regex}";
306 then
307     declare -r LFBFL_start=$((
308         echo "$1" \
309             | sed -e 's/\n/g' \
310             | head --lines=-1 \
311             | sed -z 's/\n/g'
312     ))

```

```

313 # echo "start: $LFBFL_start"
314 declare -r LFBFL_last_line=$(
315     echo "$1" \
316     | sed -e 's/\n/\n/g' \
317     | tail --lines=1
318 )
319 # echo "last_line: ${LFBFL_last_line}"
320 split_last_line_result=""
321 if [[ -n "${LFBFL_start}" ]]; then
322     split_last_line_result="${LFBFL_start}\n"
323 fi
324 if [[ -n "$5" ]]; then
325     split_line_at_most "${LFBFL_last_line}" "${LFBFL_length2}" \
326         "$5" "$6" "$7"
327     split_last_line_result+="${split_line_at_most_result_start}$4"
328     split_last_line_result+="\n"
329     split_last_line_result+="$2${split_line_at_most_result_end}"
330 else
331     split_last_line_result+="${LFBFL_last_line}:0:${LFBFL_length2}}"
332     split_last_line_result+="\$4\n"
333     split_last_line_result+="$2${LFBFL_last_line}:${LFBFL_length2}}"
334 fi
335 fi
336 }
337
338 repeated_split_last_line() {
339     declare -g repeated_split_last_line_result="$1"
340     local LFBFL_result="$1x"
341     until
342         [[ "${LFBFL_result}" == "${repeated_split_last_line_result}" ]];
343     do
344         split_last_line "${repeated_split_last_line_result}" "$2" "$3" \
345             "$4" "$5" "$6" "$7"
346         LFBFL_result="${repeated_split_last_line_result}"
347         repeated_split_last_line_result="${split_last_line_result}"
348         # echo "${repeated_split_last_line_result}"
349     done
350 }

```

2.26 build_and_checks_dependencies/too_long_code_lines.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms

```

```

6  # of the GNU General Public License
7  # as published by the Free Software Foundation,
8  # either version 3 of the License,
9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "too_long_code_lines.sh"
25 # to "too_long_code_lines.libr.sh".
26
27 LFBFL_subdir="build_and_checks_dependencies"
28 # shellcheck source=get_common_text_glob_patterns.libr.sh
29 source "./${LFBFL_subdir}/get_common_text_glob_patterns.libr.sh"
30 # shellcheck source=lines_filters.libr.sh
31 source "./${LFBFL_subdir}/lines_filters.libr.sh"
32
33 too_long_code_lines() {
34     # Options:
35     #   --verbose
36     #   --max-line-length
37     local LFBFL_arg
38
39     declare -i LFBFL_verbose=0
40     if [[ "$*" == *--verbose* ]]; then
41         echo "$0 $*"
42         LFBFL_verbose=1
43     fi
44     readonly LFBFL_verbose
45
46     declare -i LFBFL_max_line_length=70
47     for LFBFL_arg in "$@"; do
48         if [[ "${LFBFL_arg}" == --max-line-length=* ]]; then
49             LFBFL_arg=${LFBFL_arg#--max-line-length=}
50             LFBFL_max_line_length=$(( ${LFBFL_arg} ))
51             break

```

```

52         fi
53     done
54
55     if [[ ! -o pipefail ]]; then
56         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
57         set -o pipefail
58         trap 'set +o pipefail' RETURN
59     fi
60
61     get_COMMON_TEXT_FILES_GLOB_PATTERNS
62     local LFBFL_pattern
63     local LFBFL_long_line
64     local LFBFL_file_name
65     local LFBFL_line
66     local LFBFL_extension
67     local LFBFL_base_name
68     declare -ir LFBFL_overlength=$((LFBFL_max_line_length+1))
69     for LFBFL_pattern in "${COMMON_TEXT_FILES_GLOB_PATTERNS[@]}"; do
70         [[ LFBFL_verbose -eq 0 ]]\
71             || echo "Iterating on pattern: ${LFBFL_pattern}"
72         # shellcheck disable=SC2248
73         find . -type f -name "${LFBFL_pattern}" -printf '%P\n'\
74             | relevant_find\
75             | not_license_find\
76             | xargs grep -H '\${LFBFL_overlength}'\
77             | while read -r LFBFL_long_line;
78     do
79         LFBFL_file_name=${LFBFL_long_line%*:} # Drop long ':' suffix
80         LFBFL_line=${LFBFL_long_line#*:} # Drop short ':' prefix
81         LFBFL_line="${LFBFL_line%\/*}" # Drop/Replace '/' suffix by ''
82         LFBFL_extension=${LFBFL_file_name##*.} # Drop long '.' prefix
83         LFBFL_base_name=${LFBFL_file_name%.*} # Drop short '..' suffix
84         if [[ "${LFBFL_extension}" == "html" ]]; then
85             if [[ -f "${LFBFL_base_name}.md" ]]; then
86                 continue
87             fi
88             if grep --quiet --fixed-strings -- "${LFBFL_line}"\
89                 "./build_and_checks_variables/temp/files_listing.html.tpl"
90             then
91                 continue
92             fi
93             echo "${LFBFL_long_line}"
94         elif [[ "${LFBFL_extension}" == "md" ]]; then
95             if ! [[ -f "${LFBFL_base_name}.md.tpl" ]]; then
96                 echo "${LFBFL_long_line}"
97             fi

```

```

98         else
99             echo "${LFBFL_long_line}"
100        fi
101    done
102  done
103 }

```

2.27 build_and_checks_dependencies/update_common_build_and_checks.exec.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "update_common_build_and_checks.sh"
25 # to "update_common_build_and_checks.exec.sh".
26
27 source "overwrite_if_not_equal.libr.sh"
28
29 update_common_build_and_checks() {
30     declare -i LFBFL_verbose=0
31     if [[ "$*" == *--verbose* ]]; then
32         echo "$0 $*"
33         LFBFL_verbose=1
34     fi
35     readonly LFBFL_verbose
36
37     if [[ ! -o pipefail ]]; then

```

```

38     [ [ LFBFL_verbose -eq 1 ] ] && echo "pipefail option activated"
39     set -o pipefail
40     trap 'set +o pipefail' RETURN
41 fi
42
43 declare -r LFBFL_common_file_name="common_build_and_checks.exec.sh"
44 cp "./${LFBFL_common_file_name}.tpl"\ \
45   "./${LFBFL_common_file_name}.temp"
46
47 declare -ar LFBFL_file_names=(

48   "build_md_from_printable_md.exec.sh"
49   "check_shell_scripts_beginnings.libr.sh"
50   "check_shell_scripts_indentation.libr.sh"
51   "check_URLs.libr.sh"
52   "comparisons.libr.sh"
53   "create_PDF.exec.sh"
54   "generate_from_template.libr.sh"
55   "get_common_text_glob_patterns.libr.sh"
56   "grammar_and_spell_check.libr.sh"
57   "licenses_templates/build_licenses_templates.exec.sh"
58   "licenses_templates/license_file_header_AGPLv3+.tpl"
59   "licenses_templates/license_file_header_GPLv3+.tpl"
60   "licenses_templates/license_file_header_LGPLv3+.tpl"
61   "lines_counts.libr.sh"
62   "lines_filters.libr.sh"
63   "lines_maps.libr.sh"
64   "update_or_check_files_names_listing.exec.sh"
65   "overwrite_if_not_equal.libr.sh"
66   "python_black_complement.libr.sh"
67   "python_isort_complement.libr.sh"
68   "split_score.exec.php"
69   "split_score.libr.php"
70   "strings_functions.libr.sh"
71   "too_long_code_lines.libr.sh"
72 )
73
74 local LFBFL_file_name
75 local LFBFL_file_sha512
76 local LFBFL_file_sha512_all
77 local LFBFL_base_file_name
78 for LFBFL_file_name in "${LFBFL_file_names[@]}"; do
79   LFBFL_file_sha512=$((
80     sha512sum "./${LFBFL_file_name}"\
81     | cut -f1 -d' '
82   ))
83   LFBFL_file_sha512_all="local LFBFL_correct_sha512\n"

```

```

94     LFBFL_file_sha512_all+="  LFBFL_correct_sha512="
95     LFBFL_file_sha512_all+="${LFBFL_file_sha512:0:45}'\n"
96     LFBFL_file_sha512_all+="  LFBFL_correct_sha512\+="
97     LFBFL_file_sha512_all+="${LFBFL_file_sha512:45:44}'\n"
98     LFBFL_file_sha512_all+="  LFBFL_correct_sha512\+="
99     LFBFL_file_sha512_all+="${LFBFL_file_sha512:89}'"
100    LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
101    sed -i \
102      "s|@sha512_${LFBFL_base_file_name}@|${LFBFL_file_sha512_all}|g" \
103      "./${LFBFL_common_file_name}.temp"
104 done
105
106 overwrite_if_not_equal "./${LFBFL_common_file_name}" \
107   "./${LFBFL_common_file_name}.temp"
108
109 declare -r LFBFL_main_file_name="build_and_checks.exec.sh"
110 cp "./${LFBFL_main_file_name}.tpl" "../${LFBFL_main_file_name}.temp"
111
112 LFBFL_file_sha512=$(
113   sha512sum "./${LFBFL_common_file_name}" \
114   | cut -f1 -d' '
115 )
116 LFBFL_file_sha512_all="local LFBFL_correct_sha512\n"
117 LFBFL_file_sha512_all+="  LFBFL_correct_sha512="
118 LFBFL_file_sha512_all+="${LFBFL_file_sha512:0:45}'\n"
119 LFBFL_file_sha512_all+="  LFBFL_correct_sha512\+="
120 LFBFL_file_sha512_all+="${LFBFL_file_sha512:45:44}'\n"
121 LFBFL_file_sha512_all+="  LFBFL_correct_sha512\+="
122 LFBFL_file_sha512_all+="${LFBFL_file_sha512:89}'"
123 sed -i \
124   "s|@sha512_${LFBFL_common_file_name}@|${LFBFL_file_sha512_all}|g" \
125   "../${LFBFL_main_file_name}.temp"
126
127 overwrite_if_not_equal "../${LFBFL_main_file_name}" \
128   "../${LFBFL_main_file_name}.temp"
129 }
130
131 update_common_build_and_checks "$@"

```

2.28 build_and_checks_dependencies/update_or_check_files_names_listing.exec.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms

```

```

6  # of the GNU General Public License
7  # as published by the Free Software Foundation,
8  # either version 3 of the License,
9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24
25 # 2025-12-30T21:33+01:00: This file was moved from
26 # "./build_and_checks_dependencies/listings"
27 # to
28 # "./build_and_checks_dependencies".
29 # This file was renamed from "update_or_check_files_names_listing.sh"
30 # to "update_or_check_files_names_listing.exec.sh".
31 # This file was renamed from "update_or_check_file_names_listing.sh"
32 # to "update_or_check_files_names_listing.sh"?
33
34 LFBFL_subdir="build_and_checks_dependencies"
35 # shellcheck source=lines_filters.libr.sh
36 source "./${LFBFL_subdir}/lines_filters.libr.sh"
37 # shellcheck source=strings_functions.libr.sh
38 source "./${LFBFL_subdir}/strings_functions.libr.sh"
39
40 update_or_check_files_names_listing() {
41     declare -i LFBFL_verbose=0
42     if [[ "$*" == *--verbose* ]]; then
43         echo "$0 $*"
44         LFBFL_verbose=1
45     fi
46     readonly LFBFL_verbose
47
48     declare -i LFBFL_write=0
49     if [[ "$*" == *--write* ]]; then
50         LFBFL_write=1
51     fi

```

```

52     readonly LFBFL_write
53
54     declare -i LFBFL_append=0
55     if [[ "$*" == *--append* ]]; then
56         LFBFL_append=1
57     fi
58     readonly LFBFL_append
59
60     if [[ ! -o pipefail ]]; then
61         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
62         set -o pipefail
63         trap 'set +o pipefail' RETURN
64     fi
65
66     LFBFL_subdir2="build_and_checks_variables"
67     local LFBFL_listing="./${LFBFL_subdir2}/files_names_listing.txt"
68     readonly LFBFL_listing
69     local LFBFL_data_file_name="./${LFBFL_subdir2}/repository_data.txt"
70     readonly LFBFL_data_file_name
71
72     declare -i LFBFL_max_line_length
73     grep_variable "${LFBFL_data_file_name}" max_line_length\
74         --result-variable-prefix="LFBFL_"
75
76     if [[ LFBFL_write -eq 1 ]]; then
77         : > "${LFBFL_listing}"
78     fi
79     # Remove line returns here to keep lines short.
80     sed -Ez 's/\n//Mg' "${LFBFL_listing}"\
81         > "${LFBFL_listing}.temp"
82     shopt -s dotglob
83     # shellcheck disable=SC2248
84     get_split_score_simple 1 ${LFBFL_max_line_length} /
85     declare -r LFBFL_split_score_command="${get_split_score_result}"
86     local LFBFL_split_score_command_properties
87     LFBFL_split_score_command_properties="${get_split_score_result2}"
88     readonly LFBFL_split_score_command_properties
89     local LFBFL_suffix=\\"\\\" # instead of '\\"' to avoid shellcheck SC1003
90     readonly LFBFL_suffix
91     local LFBFL_file_name
92     local LFBFL_base_file_name
93     local LFBFL_split_file_name
94     find . -type f -printf '%P\n'\
95         | relevant_find\
96         | sort\
97         | while read -r LFBFL_file_name;

```

```

98 do
99     git check-ignore -q "${LFBFL_file_name}" && continue
100    LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
101    [[ "${LFBFL_base_file_name}" != files_names_listing.txt* ]] \
102        || continue
103    if [[ "${LFBFL_base_file_name}" == *.md ]]; then
104        if [[ -f "${LFBFL_file_name}.tpl" ]]; then
105            continue
106        fi
107    fi
108    # shellcheck disable=SC2248
109    repeated_split_last_line "${LFBFL_file_name}" \
110        """ \
111        ${LFBFL_max_line_length} \
112        "${LFBFL_suffix}" \
113        "${LFBFL_split_score_command}" \
114        "${LFBFL_split_score_command_properties}"
115    LFBFL_split_file_name="${repeated_split_last_line_result}"
116    if [[ LFBFL_write -eq 1 ]]; then
117        # shellcheck disable=SC2001
118        echo "${LFBFL_file_name}" \
119            | sed -e "s|${LFBFL_file_name}|${LFBFL_split_file_name}|g" \
120            >> "${LFBFL_listing}"
121        continue
122    fi
123    if grep -q "${LFBFL_file_name}\$" "${LFBFL_listing}.temp"; then
124        :
125    else
126        echo \
127            "The file ${LFBFL_file_name} is not listed in ${LFBFL_listing}."
128        if [[ LFBFL_append -eq 1 ]]; then
129            # shellcheck disable=SC2001
130            echo "${LFBFL_file_name}" \
131                | sed -e "s|${LFBFL_file_name}|${LFBFL_split_file_name}|g" \
132                >> "${LFBFL_listing}"
133        fi
134    fi
135 done
136
137 while read -r LFBFL_file_name;
138 do
139    [[ "${LFBFL_file_name}" != /** ]] || continue
140    LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
141    if ! [[ -f "${LFBFL_file_name}" ]]; then
142        echo\
143            "The non-file ${LFBFL_file_name} is listed in ${LFBFL_listing}.""

```

```

144         fi
145     done < "${LFBFL_listing}.temp"
146
147     rm "${LFBFL_listing}.temp"
148     shopt -u dotglob
149     shopt -s globstar
150
151     if [[ LFBFL_verbose -eq 1 ]]; then
152         cat "${LFBFL_listing}"
153     fi
154 }
155
156 update_or_check_files_names_listing "$@"

```

2.29 build_and_checks_variables/.gitignore

```

1 temp/
2 upgrade_venvs_ts
3

```

2.30 build_and_checks_variables/bandit.ini

```

1 # FILE: .bandit
2 [bandit]

```

2.31 build_and_checks_variables/bandit_baseline.json

```

1 {"results": []}

```

2.32 build_and_checks_variables/black.toml

```

1 [tool.black]
2 line-length = 70

```

2.33 build_and_checks_variables/DevOrSysAdminScripts.html.tpl

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <!--
4 This file is part of DevOrSysAdminScripts library.
5
6 DevOrSysAdminScripts is free software:
7 you can redistribute it and/or modify it under the terms
8 of the GNU Lesser General Public License
9 as published by the Free Software Foundation,
10 either version 3 of the License,

```

```

11 or (at your option) any later version.
12
13 DevOrSysAdminScripts is distributed in the hope
14 that it will be useful,
15 but WITHOUT ANY WARRANTY;
16 without even the implied warranty of
17 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
18 See the GNU Lesser General Public License for more details.
19
20 You should have received a copy of
21 the GNU Lesser General Public License
22 along with DevOrSysAdminScripts.
23 If not, see <https://www.gnu.org/licenses/>.
24
25 ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
26 -->
27 <head>
28 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
29 <meta name="generator" content="DevOrSysAdminScripts">
30 <style>
31 .title{
32     margin: 2ex auto;
33     text-align: center;
34 }
35 .titlemain{
36     margin: 1ex 2ex 2ex 1ex;
37 }
38 .titlerest{
39     font-size: 1.17em;
40     margin: 0ex 2ex;
41 }
42 table{
43     border-collapse: collapse;
44 }
45 td{
46     padding: 0;
47 }
48 pre{
49     text-align: left;
50     margin-left: 0ex;
51     margin-right: auto;
52 }
53 pre.numbered_lines:before{
54     background: snow;
55     border-right: 1px black solid;
56     color: #010101;

```

```

57     content: attr(line-numbers);
58     float: left;
59     padding: 0 1ch;
60     text-align: right;
61   }
62   blockquote{
63     margin-left: 4ex;
64     margin-right: 4ex;
65     text-align: left;
66   }
67   ol.level1, ol.level2{
68     list-style-type: none;
69     /*
70     list-style-type: decimal;
71     Previous rule is shorter and more efficient if you need only level1.
72     But you don't have accessible counter if you use decimal.
73     */
74   }
75   ol.level1{
76     counter-reset: level1;
77   }
78   ol.level2{
79     counter-reset: level2;
80   }
81   ol.level1 li:before{
82     content: counter(level1) ". ";
83     counter-increment: level1;
84   }
85   ol.level2 li:before{
86     content: counter(level1) "." counter(level2) " ";
87     counter-increment: level2;
88   }
89 </style>
90 <title>@repository_name@</title>
91 </head>
92 <body >
93 <table class="title">
94   <tr>
95     <td style="padding:1ex">
96       <h1 class="titlemain">@repository_name@</h1>
97       <h2 class="titlerest">
98         @author_full_name@
99         <br>
100        <a href="@author_website@">
101          <span style="font-family:monospace">
102            @author_website@

```

```

103         </span>
104     </a>
105     <br>
106     <a href="mailto:@author_email@">
107         <span style="font-family:monospace">
108             @author_email@
109         </span>
110     </a>
111     </h2>
112     </td>
113   </tr>
114 </table>
115
116 <blockquote class="abstract">
117     <span style="font-weight:bold">Abstract:</span>
118     @abstract@
119 </blockquote>
120 <p>Current version: @current_date@</p>
121 <p>Current number of commits: @number_of_commits@</p>
122 <p>Current git SHA1: @current_git_SHA1@</p>
123 <p>
124 Code lines: @number_of_lines@
125 </p>
126
127
128 <h2>Contents</h2>
129
130 <ol class="level1">
131     <li><a href="#section1">Files Tree</a></li>
132     <li><a href="#section2">Listing of files</a><br>
133         <ol class="level2">
134             @files_lis@
135         </ol>
136     </li>
137 </ol>
138
139
140 <h2 id="section1">1 Files tree</h2>
141 <pre class="verbatim">
142 @current_tree_light@
143 </pre>
144 <pre class="verbatim">
145 @current_tree@
146 </pre>
147
148

```

```

149 <h2 id="section2">2 Listing of files</h2>
150
151 <p>
152 This repository is dual licensed.
153 All the Bash scripts source code is covered by GPLv3+.
154 All the PHP scripts source code is covered by LGPLv3+.
155 </p>
156
157 <p>
158 The text of the licenses is available at:
159 <a href="https://www.gnu.org/licenses/"><span
160   style="font-family:monospace"
161 >https://www.gnu.org/licenses/</span></a>.
162 The git repository of this source code is also available at:
163 <a href="@repository_web_url@"><span
164   style="font-family:monospace"
165 >@repository_web_url@</span></a>.
166 Clone it with this URL:
167 <span style="font-family:monospace"
168 >@repository_git_url@</span>.
169 </p>
170
171 <div>
172 <p>
173 Here is a listing of the dependencies of the code in this repository:
174 </p>
175 <ul>
176 <li> The following dependencies are GPLv3+ licensed:<br>
177 basename, bash (and its builtins), cat, chmod, cp, cut, date, diff,
178 dirname, find, grep, head, mkdir, mv, realpath, rm, sed, sha512sum,
179 shellcheck, sort, tail, wc, wget, xargs
180 <li> The following dependencies are GPLv2+ licensed:<br>
181 more, pandoc, pdftex, pdflatex, tree
182 <li> The following dependencies are GPLv2 licensed:<br>
183 git, pylint
184 <li> The following dependencies are LGPLv2.1+ licensed:<br>
185 notify-send
186 <li> The following dependencies are MIT licensed:<br>
187 black, isort, mypy, ruff
188 <li> The following dependencies are Apache-2.0 licensed:<br>
189 bandit
190 <li> The following dependencies are BSD-3-Clause licensed:<br>
191 phpmd
192 <li> php is licensed under PHP License v3.01
193 <li> tex is licensed under Knuth license
194 <li> pcregrep has a custom license from Cambridge University

```

```

195  </ul>
196  </div>
197
198 <p>
199 Remark: Some commands listed under GPLv3+ are POSIX
200 and can have different implementations under another license;
201 for example, uutils coreutils is MIT licensed,
202 whilst GNU coreutils is GPLv3+ licensed.
203 </p>
204
205 @files_listing_HTMLPreInput@
206
207 <p>
208 @acknowledgments@
209 </p>
210
211 <script>
212 "use strict";
213 const pres = Array.from(
214   document.getElementsByClassName('numbered_lines')
215 );
216 pres.forEach(
217   (pre) => {
218     const nb = Array.from(pre.innerText.matchAll("\n")).length + 1;
219     let numbers = "";
220     for(let i = 1; i <= nb; ++i){
221       numbers += i.toString();
222       if(i < nb){ numbers += "\n ";}
223     }
224     pre.setAttribute('line-numbers', numbers);
225   }
226 );
227 </script>
228 </html>

```

2.34 build_and_checks_variables/DevOrSysAdminScripts.tex.tpl

```

1 % This file is part of DevOrSysAdminScripts library.
2 %
3 % DevOrSysAdminScripts is free software:
4 % you can redistribute it and/or modify it under the terms
5 % of the GNU Lesser General Public License
6 % as published by the Free Software Foundation,
7 % either version 3 of the License,
8 % or (at your option) any later version.
9 %

```

```

10  % DevOrSysAdminScripts is distributed in the hope
11  % that it will be useful,
12  % but WITHOUT ANY WARRANTY;
13  % without even the implied warranty of
14  % MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
15  % See the GNU Lesser General Public License for more details.
16  %
17  % You should have received a copy of
18  % the GNU Lesser General Public License
19  % along with DevOrSysAdminScripts.
20  % If not, see <https://www.gnu.org/licenses/>.
21  %
22  % ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
23
24  % 2025-12-29T17:08+01:00: This file was moved from
25  % "./latex"
26  % to
27  % "./build_and_checks_variables".
28
29  \documentclass{article}
30
31  \usepackage[utf8]{inputenc}
32  \usepackage[T1]{fontenc}
33  \usepackage{subfigure}
34  \usepackage{amsmath}
35  \usepackage{amssymb}
36  \usepackage{amsthm}
37  \usepackage[pdftex]{hyperref}
38  \usepackage{tikz}
39  \usepackage{caption}
40  \usepackage[round]{natbib}
41  \usepackage{fancyhdr}
42  \usepackage{amsfonts}
43  \usepackage{times}
44  \usepackage{ifpdf}
45  \usepackage{latextsym}
46  \usepackage{graphicx}
47  \usepackage{enumerate}
48  \usepackage{pmbboxdraw}
49  \usepackage{fancyvrb}
50  \usepackage[french,english]{babel}
51
52  % *** les environnements ***
53  %\theoremstyle{break}
54  \newtheorem{definition}{Definition}[section]
55  \newtheorem{proposition}[definition]{Proposition}

```

```

56 \newtheorem{theorem}{definition}{Theorem}
57 \newtheorem{lemma}{definition}{Lemma}
58 \newtheorem{corollary}{definition}{Corollary}
59 \newtheorem{remark}{definition}{Remark}
60 \newtheorem{openproblem}{definition}{Open problem}
61
62 \begin{document}
63
64 \author{
65   @author_full_name@ \\
66   \url{@author_website@} \\
67   \href{mailto:@author_email@}{@author_email@}
68 }
69 \title{@repository_name@}
70
71 \maketitle
72 \begin{abstract}
73 @abstract@
74 \end{abstract}
75
76 Current version: @current_date@
77
78 Current number of commits: @number_of_commits@
79
80 Current git SHA1: @current_git_SHA1@
81
82 Code lines: @number_of_lines@
83
84 \tableofcontents
85 \newpage
86
87 \section{Files tree}
88 \label{section:tree}
89
90 \begin{verbatim}
91 @current_tree_light@
92 \end{verbatim}
93
94 \begin{verbatim}
95 @current_tree@
96 \end{verbatim}
97
98 \section{Listing of files}
99 \label{section:listing}
100
101 This repository is dual licensed.

```

```

102 All the Bash scripts source code is covered by GPLv3+.
103 All the PHP scripts source code is covered by LGPLv3+.
104
105 The text of the licenses is available at:
106 \url{https://www.gnu.org/licenses/}.
107 The git repository of this source code is also available at:
108 \url{@repository_web_url@}.
109 Clone it with this URL:
110 \url{@repository_git_url@}.

111
112 Here is a listing of the dependencies of the code in this repository:
113 \begin{itemize}
114     \item The following dependencies are GPLv3+ licensed:\newline
115         basename, bash (and its builtins), cat, chmod, cp, cut, date, diff,
116         dirname, find, grep, head, mkdir, mv, realpath, rm, sed, sha512sum,
117         shellcheck, sort, tail, wc, wget, xargs
118     \item The following dependencies are GPLv2+ licensed:\newline
119         more, pandoc, pdftex, pdflatex, tree
120     \item The following dependencies are GPLv2 licensed:\newline
121         git, pylint
122     \item The following dependencies are LGPLv2.1+ licensed:\newline
123         notify-send
124     \item The following dependencies are MIT licensed:\newline
125         black, isort, mypy, ruff
126     \item The following dependencies are Apache-2.0 licensed:\newline
127         bandit
128     \item The following dependencies are BSD-3-Clause licensed:\newline
129         phpmd
130     \item php is licensed under PHP License v3.01
131     \item tex is licensed under Knuth license
132     \item pcregrep has a custom license from Cambridge University
133 \end{itemize}

134
135 Remark: Some commands listed under GPLv3+ are POSIX
136 and can have different implementations under another license;
137 for example, uutils coreutils is MIT licensed,
138 whilst GNU coreutils is GPLv3+ licensed.

139
140 @files_listing_VerbatimInput@

141
142 @acknowledgments@

143
144 \end{document}

```

2.35 build_and_checks_variables/files_listing.tex.tpl

```
1 \subsection{  
2     build\_\_and\_\_checks\_\_dependencies/build\_\_and\_\_checks.exec.sh.tpl  
3 }  
4 \label{  
5     build_and_checks_dependencies:build_and_checksexecshtpl  
6 }  
7  
8 \VerbatimInput [numbers=left,xleftmargin=-5mm] {  
9 build_and_checks_dependencies/build_and_checks.exec.sh.tpl  
10 }  
11  
12  
13 \subsection{  
14     build\_\_and\_\_checks\_\_dependencies/%  
15 build\_\_md\_\_from\_\_printable\_\_md.exec.sh  
16 }  
17 \label{  
18     build_and_checks_dependencies:build_md_from_printable_mdexecsh  
19 }  
20  
21 \VerbatimInput [numbers=left,xleftmargin=-5mm] {  
22 build_and_checks_dependencies/build_md_from_printable_md.exec.sh  
23 }  
24  
25  
26 \subsection{  
27     build\_\_and\_\_checks\_\_dependencies/%  
28 check\_\_shell\_\_scripts\_\_beginnings.libr.sh  
29 }  
30 \label{  
31     build_and_checks_dependencies:check_shell_scripts_beginningslibrsh  
32 }  
33  
34 \VerbatimInput [numbers=left,xleftmargin=-5mm] {  
35 build_and_checks_dependencies/check_shell_scripts_beginnings.libr.sh  
36 }  
37  
38  
39 \subsection{  
40     build\_\_and\_\_checks\_\_dependencies/%  
41 check\_\_shell\_\_scripts\_\_indentation.libr.sh  
42 }  
43 \label{  
44     build_and_checks_dependencies:check_shell_scripts_indentationlibrsh
```

```

45 }
46
47 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
48 build_and_checks_dependencies/check_shell_scripts_indentation.libr.sh
49 }
50
51
52 \subsection{
53   build\_\_and\_\_checks\_\_dependencies/check\_URLs.libr.sh
54 }
55 \label{
56   build_and_checks_dependencies:check_URLslibrsh
57 }
58
59 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
60 build_and_checks_dependencies/check_URLs.libr.sh
61 }
62
63
64 \subsection{
65   build\_\_and\_\_checks\_\_dependencies/common\_build\_\_and\_\_checks.exec.sh
66 }
67 \label{
68   build_and_checks_dependencies:common_build_and_checksexecsh
69 }
70
71 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
72 build_and_checks_dependencies/common_build_and_checks.exec.sh
73 }
74
75
76 \subsection{
77   build\_\_and\_\_checks\_\_dependencies/%
78 common\_build\_\_and\_\_checks.exec.sh.tpl
79 }
80 \label{
81   build_and_checks_dependencies:common_build_and_checksexecstpl
82 }
83
84 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
85 build_and_checks_dependencies/common_build_and_checks.exec.sh.tpl
86 }
87
88
89 \subsection{
90   build\_\_and\_\_checks\_\_dependencies/comparisons.libr.sh

```

```

91    }
92 \label{
93     build_and_checks_dependencies:comparisonslibrsh
94 }
95
96 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
97 build_and_checks_dependencies/comparisons.libr.sh
98 }
99
100
101 \subsection{
102     build\_and\_checks\_dependencies/create\_PDF.exec.sh
103 }
104 \label{
105     build_and_checks_dependencies:create_PDFexecsh
106 }
107
108 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
109 build_and_checks_dependencies/create_PDF.exec.sh
110 }
111
112
113 \subsection{
114     build\_and\_checks\_dependencies/generate\_from\_template.libr.sh
115 }
116 \label{
117     build_and_checks_dependencies:generate_from_templatelibrsh
118 }
119
120 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
121 build_and_checks_dependencies/generate_from_template.libr.sh
122 }
123
124
125 \subsection{
126     build\_and\_checks\_dependencies/%
127 get_common_text_glob_patterns.libr.sh
128 }
129 \label{
130     build_and_checks_dependencies:get_common_text_glob_patternslibrsh
131 }
132
133 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
134 build_and_checks_dependencies/get_common_text_glob_patterns.libr.sh
135 }
136

```

```

137
138 \subsection{
139     build\_and\_checks\_dependencies/grammar\_and\_spell\_check.libr.sh
140 }
141 \label{
142     build_and_checks_dependencies:grammar_and_spell_checklibrsh
143 }
144
145 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
146 build_and_checks_dependencies/grammar_and_spell_check.libr.sh
147 }
148
149
150 \subsection{
151     build\_and\_checks\_dependencies/licenses\_templates/%
152 build\_licenses\_templates.exec.sh
153 }
154 \label{
155     build_and_checks_dependencies:licenses_templates:%
156 build_licenses_templatesexecsh
157 }
158
159 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
160 build_and_checks_dependencies/licenses_templates/%
161 build_licenses_templates.exec.sh
162 }
163
164
165 \subsection{
166     build\_and\_checks\_dependencies/licenses\_templates/%
167 license\_file\_header\_AGPLv3+.tpl
168 }
169 \label{
170     build_and_checks_dependencies:licenses_templates:%
171 license_file_header_AGPLv3+tpl
172 }
173
174 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
175 build_and_checks_dependencies/licenses_templates/%
176 license_file_header_AGPLv3+.tpl
177 }
178
179
180 \subsection{
181     build\_and\_checks\_dependencies/licenses\_templates/%
182 license\_file\_header\_GPLv3+.tpl

```

```

183 }
184 \label{
185   build_and_checks_dependencies:licenses_templates:%
186   license_file_header_GPLv3+tpl
187 }
188
189 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
190 build_and_checks_dependencies/licenses_templates/%
191 license_file_header_GPLv3+.tpl
192 }
193
194
195 \subsection{
196   build\_\_and\_\_checks\_\_dependencies/licenses\_templates/%
197   license\_file\_header\_LGPLv3+.tpl
198 }
199 \label{
200   build_and_checks_dependencies:licenses_templates:%
201   license_file_header_LGPLv3+tpl
202 }
203
204 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
205 build_and_checks_dependencies/licenses_templates/%
206 license_file_header_LGPLv3+.tpl
207 }
208
209
210 \subsection{
211   build\_\_and\_\_checks\_\_dependencies/lines\_counts.libr.sh
212 }
213 \label{
214   build_and_checks_dependencies:lines_countslibrsh
215 }
216
217 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
218 build_and_checks_dependencies/lines_counts.libr.sh
219 }
220
221
222 \subsection{
223   build\_\_and\_\_checks\_\_dependencies/lines\_filters.libr.sh
224 }
225 \label{
226   build_and_checks_dependencies:lines_filterslibrsh
227 }
228

```

```

229 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
230 build_and_checks_dependencies/lines_filters.libr.sh
231 }
232
233
234 \subsection{
235     build\_and\_checks\_dependencies/lines\_maps.libr.sh
236 }
237 \label{
238     build_and_checks_dependencies:lines_mapslibrsh
239 }
240
241 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
242 build_and_checks_dependencies/lines_maps.libr.sh
243 }
244
245
246 \subsection{
247     build\_and\_checks\_dependencies/overwrite\_if\_not\_equal.libr.sh
248 }
249 \label{
250     build_and_checks_dependencies:overwrite_if_not_equallibrsh
251 }
252
253 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
254 build_and_checks_dependencies/overwrite_if_not_equal.libr.sh
255 }
256
257
258 \subsection{
259     build\_and\_checks\_dependencies/python\_black\_complement.libr.sh
260 }
261 \label{
262     build_and_checks_dependencies:python_black_complementlibrsh
263 }
264
265 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
266 build_and_checks_dependencies/python_black_complement.libr.sh
267 }
268
269
270 \subsection{
271     build\_and\_checks\_dependencies/python\_isort\_complement.libr.sh
272 }
273 \label{
274     build_and_checks_dependencies:python_isort_complementlibrsh

```

```

275 }
276 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
277 build_and_checks_dependencies/python_isort_complement.libr.sh
278 }
280
281
282 \subsection{
283     build_and_checks_dependencies/split\_score.exec.php
284 }
285 \label{
286     build_and_checks_dependencies:split_scoreexecphp
287 }
288
289 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
290 build_and_checks_dependencies/split_score.exec.php
291 }
292
293
294 \subsection{
295     build_and_checks_dependencies/split\_score.libr.php
296 }
297 \label{
298     build_and_checks_dependencies:split_scorelibrphp
299 }
300
301 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
302 build_and_checks_dependencies/split_score.libr.php
303 }
304
305
306 \subsection{
307     build_and_checks_dependencies/strings\_functions.libr.sh
308 }
309 \label{
310     build_and_checks_dependencies:strings_functionslibrsh
311 }
312
313 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
314 build_and_checks_dependencies/strings_functions.libr.sh
315 }
316
317
318 \subsection{
319     build_and_checks_dependencies/too\_long\_code\_lines.libr.sh
320 }

```

```

321 \label{
322     build_and_checks_dependencies:too_long_code_lineslibrsh
323 }
324
325 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
326 build_and_checks_dependencies/too_long_code_lines.libr.sh
327 }
328
329
330 \subsection{
331     build\_\_and\_\_checks\_\_dependencies/%
332 update\_\_common\_\_build\_\_and\_\_checks.exec.sh
333 }
334 \label{
335     build_and_checks_dependencies:update_common_build_and_checksexecsh
336 }
337
338 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
339 build_and_checks_dependencies/update_common_build_and_checks.exec.sh
340 }
341
342
343 \subsection{
344     build\_\_and\_\_checks\_\_dependencies/%
345 update\_\_or\_\_check\_\_files\_\_names\_\_listing.exec.sh
346 }
347 \label{
348     build_and_checks_dependencies:%
349 update_or_check_files_names_listingexecsh
350 }
351
352 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
353 build_and_checks_dependencies/%
354 update_or_check_files_names_listing.exec.sh
355 }
356
357
358 \subsection{
359     build\_\_and\_\_checks\_\_variables/.gitignore
360 }
361 \label{
362     build_and_checks_variables:gitignore
363 }
364
365 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
366 build_and_checks_variables/.gitignore

```

```

367     }
368
369
370 \subsection{
371     build\_and\_checks\_variables/bandit.ini
372 }
373 \label{
374     build_and_checks_variables:banditini
375 }
376
377 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
378 build_and_checks_variables/bandit.ini
379 }
380
381
382 \subsection{
383     build\_and\_checks\_variables/bandit\_baseline.json
384 }
385 \label{
386     build_and_checks_variables:bandit_baselinejson
387 }
388
389 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
390 build_and_checks_variables/bandit_baseline.json
391 }
392
393
394 \subsection{
395     build\_and\_checks\_variables/black.toml
396 }
397 \label{
398     build_and_checks_variables:blacktoml
399 }
400
401 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
402 build_and_checks_variables/black.toml
403 }
404
405
406 \subsection{
407     build\_and\_checks\_variables/DevOrSysAdminScripts.html.tpl
408 }
409 \label{
410     build_and_checks_variables:DevOrSysAdminScriptshmtltp1
411 }
412

```

```

413 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
414 build_and_checks_variables/DevOrSysAdminScripts.html.tpl
415 }
416
417
418 \subsection{
419     build_and_checks_variables/DevOrSysAdminScripts.tex.tpl
420 }
421 \label{
422     build_and_checks_variables:DevOrSysAdminScriptstexttpl
423 }
424
425 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
426 build_and_checks_variables/DevOrSysAdminScripts.tex.tpl
427 }
428
429
430 \subsection{
431     build_and_checks_variables/files_listing.tex.tpl
432 }
433 \label{
434     build_and_checks_variables:files_listingtexttpl
435 }
436
437 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
438 build_and_checks_variables/files_listing.tex.tpl
439 }
440
441
442 \subsection{
443     build_and_checks_variables/files_names_listing.txt
444 }
445 \label{
446     build_and_checks_variables:files_names_listingtxt
447 }
448
449 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
450 build_and_checks_variables/files_names_listing.txt
451 }
452
453
454 \subsection{
455     build_and_checks_variables/licenses_notes.txt
456 }
457 \label{
458     build_and_checks_variables:licenses_notestxt

```

```

459 }
460
461 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
462 build_and_checks_variables/licenses_notes.txt
463 }
464
465
466 \subsection{
467     build\_\_and\_\_checks\_\_variables/phpmd\_\_baseline.xml
468 }
469 \label{
470     build_and_checks_variables:phpmd_baselinexml
471 }
472
473 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
474 build_and_checks_variables/phpmd_baseline.xml
475 }
476
477
478 \subsection{
479     build\_\_and\_\_checks\_\_variables/pylintrc.toml
480 }
481 \label{
482     build_and_checks_variables:pylintrctoml
483 }
484
485 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
486 build_and_checks_variables/pylintrc.toml
487 }
488
489
490 \subsection{
491     build\_\_and\_\_checks\_\_variables/repository\_data.txt
492 }
493 \label{
494     build_and_checks_variables:repository_datatxt
495 }
496
497 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
498 build_and_checks_variables/repository_data.txt
499 }
500
501
502 \subsection{
503     build\_\_and\_\_checks\_\_variables/ruff.toml
504 }

```

```

505 \label{
506     build_and_checks_variables:rufftoml
507 }
508
509 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
510 build_and_checks_variables/ruff.toml
511 }
512
513
514 \subsection{
515     build_and_checks_variables/shellcheck.ini
516 }
517 \label{
518     build_and_checks_variables:shellcheckini
519 }
520
521 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
522 build_and_checks_variables/shellcheck.ini
523 }
524
525
526 \subsection{
527     build_and_checks_variables/shellcheck_notes.txt
528 }
529 \label{
530     build_and_checks_variables:shellcheck_notestxt
531 }
532
533 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
534 build_and_checks_variables/shellcheck_notes.txt
535 }
536
537
538 \subsection{
539     build_and_checks_exec.sh
540 }
541 \label{
542     build_and_checks_execsh
543 }
544
545 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
546 build_and_checks_exec.sh
547 }
548
549
550 \subsection{

```

```

551     COPYING
552 }
553 \label{
554     COPYING
555 }
556
557 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
558 COPYING
559 }
560
561
562 \subsection{
563     COPYING.LESSER
564 }
565 \label{
566     COPYINGLESSER
567 }
568
569 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
570 COPYING.LESSER
571 }
572
573
574 \subsection{
575     git\_helpers/git\_checks.libr.sh
576 }
577 \label{
578     git_helpers:git_checkslibrsh
579 }
580
581 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
582 git_helpers/git_checks.libr.sh
583 }
584
585
586 \subsection{
587     git\_helpers/post-commit
588 }
589 \label{
590     git_helpers:post-commit
591 }
592
593 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
594 git_helpers/post-commit
595 }
596

```

```

597 \subsection{
598     git\_helpers/post\_commit\_renaming\_comment.libr.sh
599 }
600 \label{
601     git_helpers:post_commit_renaming_commentlibrsh
602 }
603 }
604
605 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
606 git_helpers/post_commit_renaming_comment.libr.sh
607 }
608
609 \subsection{
610     git\_helpers/pre-commit
611 }
612 \label{
613     git_helpers:pre-commit
614 }
615 }
616
617 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
618 git_helpers/pre-commit
619 }
620
621 \subsection{
622     IMAP\_transfert.py
623 }
624 \label{
625     IMAP_transfertpy
626 }
627 }
628
629 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
630 IMAP_transfert.py
631 }
632
633 \subsection{
634     LiensUtiles.txt
635 }
636 \label{
637     LiensUtilestxt
638 }
639 }
640
641 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
642 LiensUtiles.txt

```

```

643 }
644
645
646 \subsection{
647     README.md.tpl
648 }
649 \label{
650     READMEmdtpl
651 }
652
653 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
654 README.md.tpl
655 }
656
657
658 \subsection{
659     unicode.libr.php
660 }
661 \label{
662     unicodelibrphp
663 }
664
665 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
666 unicode.libr.php
667 }
668
669
670 \subsection{
671     upgrade\_build\_and\_checks.exec.sh
672 }
673 \label{
674     upgrade_build_and_checksexecsh
675 }
676
677 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
678 upgrade_build_and_checks.exec.sh
679 }
680
681
682 \subsection{
683     wget\_sha512.libr.sh
684 }
685 \label{
686     wget_sha512librsh
687 }
688

```

```

689 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
690 wget_sha512.libr.sh
691 }
692
693

```

2.36 build_and_checks_variables/files_names_listing.txt

```

1 // This file is part of DevOrSysAdminScripts library.
2 //
3 // DevOrSysAdminScripts is free software:
4 // you can redistribute it and/or modify it under the terms
5 // of the GNU Lesser General Public License
6 // as published by the Free Software Foundation,
7 // either version 3 of the License,
8 // or (at your option) any later version.
9 //
10 // DevOrSysAdminScripts is distributed in the hope
11 // that it will be useful,
12 // but WITHOUT ANY WARRANTY;
13 // without even the implied warranty of
14 // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
15 // See the GNU Lesser General Public License for more details.
16 //
17 // You should have received a copy of
18 // the GNU Lesser General Public License
19 // along with DevOrSysAdminScripts.
20 // If not, see <https://www.gnu.org/licenses/>.
21 //
22 // ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
23 // 2024-10-31T21:37+01:00: This file was moved from
24 // "./build_and_checks_dependencies/listings"
25 // to
26 // "./build_and_checks_variables".
27 // Blasé20240517Résultat.txt
28 // Blasé20240517.sh
29 // Blasé20240518_2.txt
30 // Blasé20240518_3.txt
31 build_and_checks_dependencies/build_and_checks.exec.sh.tpl
32 build_and_checks_dependencies/build_md_from_printable_md.exec.sh
33 build_and_checks_dependencies/check_shell_scripts_beginnings.libr.sh
34 build_and_checks_dependencies/check_shell_scripts_indentation.libr.sh
35 build_and_checks_dependencies/check_URLs.libr.sh
36 build_and_checks_dependencies/common_build_and_checks.exec.sh
37 build_and_checks_dependencies/common_build_and_checks.exec.sh.tpl
38 build_and_checks_dependencies/comparisons.libr.sh

```

```

39 build_and_checks_dependencies/create_PDF.exec.sh
40 build_and_checks_dependencies/generate_from_template.libr.sh
41 build_and_checks_dependencies/get_common_text_glob_patterns.libr.sh
42 build_and_checks_dependencies/grammar_and_spell_check.libr.sh
43 build_and_checks_dependencies/licenses_templates/\
44 build_licenses_templates.exec.sh
45 build_and_checks_dependencies/licenses_templates/\
46 license_file_header GPLv3+.tpl
47 build_and_checks_dependencies/licenses_templates/\
48 license_file_header GPLv3+.tpl
49 build_and_checks_dependencies/licenses_templates/\
50 license_file_header GPLv3+.tpl
51 build_and_checks_dependencies/lines_counts.libr.sh
52 build_and_checks_dependencies/lines_filters.libr.sh
53 build_and_checks_dependencies/lines_maps.libr.sh
54 // build_and_checks_dependencies/listings/files_listing.tex.tpl
55 build_and_checks_dependencies/overwrite_if_not_equal.libr.sh
56 build_and_checks_dependencies/python_black_complement.libr.sh
57 build_and_checks_dependencies/python_isort_complement.libr.sh
58 build_and_checks_dependencies/split_score.exec.php
59 build_and_checks_dependencies/split_score.libr.php
60 build_and_checks_dependencies/strings_functions.libr.sh
61 build_and_checks_dependencies/too_long_code_lines.libr.sh
62 build_and_checks_dependencies/update_common_build_and_checks.exec.sh
63 build_and_checks_dependencies/\
64 update_or_check_files_names_listing.exec.sh
65 build_and_checks_variables/.gitignore
66 build_and_checks_variables/bandit.ini
67 build_and_checks_variables/bandit_baseline.json
68 build_and_checks_variables/black.toml
69 build_and_checks_variables/DevOrSysAdminScripts.html.tpl
70 // build_and_checks_variables/DevOrSysAdminScripts.tex
71 build_and_checks_variables/DevOrSysAdminScripts.tex.tpl
72 build_and_checks_variables/files_listing.tex.tpl
73 build_and_checks_variables/files_names_listing.txt
74 build_and_checks_variables/licenses_notes.txt
75 build_and_checks_variables/phpmd_baseline.xml
76 build_and_checks_variables/pylintrc.toml
77 build_and_checks_variables/repository_data.txt
78 build_and_checks_variables/ruff.toml
79 build_and_checks_variables/shellcheck.ini
80 build_and_checks_variables/shellcheck_notes.txt
81 // build_and_checks_variables/upgrade_venvs_ts
82 build_and_checks.exec.sh
83 COPYING
84 COPYING.LESSER

```

```

85 // DevOrSysAdminScripts.html
86 // DevOrSysAdminScripts.pdf
87 // EncoreBlasé20240517-20240518.txt
88 git_helpers/git_checks.libr.sh
89 git_helpers/post-commit
90 git_helpers/post_commit_renaming_comment.libr.sh
91 git_helpers/pre-commit
92 IMAP_transfert.py
93 LiensUtiles.txt
94 // README.html
95 // README.md
96 README.md.tpl
97 unicode.libr.php
98 upgrade_build_and_checks.exec.sh
99 wget_sha512.libr.sh

```

2.37 build_and_checks_variables/licenses_notes.txt

```

1 This repository is dual licensed.
2 All the Bash scripts source code is covered by GPLv3+.
3 All the PHP scripts source code is covered by LGPLv3+.
4
5 Here is a listing of the dependencies of the code in this repository:
6 - The following dependencies are GPLv3+ licensed:
7 basename, bash (and its builtins), cat, chmod, cp, cut, date, diff,
8 dirname, find, grep, head, mkdir, mv, realpath, rm, sed, sha512sum,
9 shellcheck, sort, tail, wc, wget, xargs
10 - The following dependencies are GPLv2+ licensed:
11 more, pandoc, pdftex, pdflatex, tree
12 - The following dependencies are GPLv2 licensed:
13 git, pylint
14 - The following dependencies are GPLv2.1+ licensed:
15 notify-send
16 - The following dependencies are MIT licensed:
17 black, isort, mypy, ruff
18 - The following dependencies are Apache-2.0 licensed:
19 bandit
20 - The following dependencies are BSD-3-Clause licensed:
21 phpmd
22 - php is licensed under PHP License v3.01
23 - tex is licensed under Knuth license
24 - pcregrep has a custom license from Cambridge University
25
26 Remark: Some commands listed under GPLv3+ are POSIX
27 and can have different implementations under another license;
28 for example, uutils coreutils is MIT licensed,

```

29 whilst GNU coreutils is GPLv3+ licensed.

2.38 build_and_checks_variables/phpmd_baseline.xml

```
1 <?xml version="1.0"?>
2 <phpmd-baseline>
3   <violation rule="PHPMD\Rule\Controversial\CamelCaseParameterName" file="build_and_
4     <violation rule="PHPMD\Rule\Naming\LongVariable" file="build_and_checks_dependencies.xml"/>
5     <violation rule="PHPMD\Rule\Controversial\CamelCaseVariableName" file="build_and_checks_dependencies.xml"/>
6     <violation rule="PHPMD\Rule\Design\LongMethod" file="unicode.libr.php"/>
7     <violation rule="PHPMD\Rule\Controversial\CamelCaseParameterName" file="unicode.libr.php"/>
8     <violation rule="PHPMD\Rule\Naming\LongVariable" file="unicode.libr.php"/>
9     <violation rule="PHPMD\Rule\Controversial\CamelCaseVariableName" file="unicode.libr.php"/>
10    <violation rule="PHPMD\Rule\CleanCode\MissingImport" file="unicode.libr.php"/>
11    <violation rule="PHPMD\Rule\CyclomaticComplexity" file="unicode.libr.php"/>
12    <violation rule="PHPMD\Rule\Design\NpathComplexity" file="unicode.libr.php"/>
13 </phpmd-baseline>
```

2.39 build_and_checks_variables/pylintrc.toml

```
1 [tool pylint.main]
2 exit-zero = true
3 ignore = ["node_modules"]
4 jobs = 0
5 [tool pylint.basic]
6 # We use uppercase acronyms like HTML, IMAP, etc.
7 module-rgx = "^[_a-zA-Z0-9]+$"
8 function-rgx = "^[_a-zA-Z0-9]+$"
9 [tool pylint."messages control"]
10 # Code like R0912 is better to order disables by category.
11 disable = [
12   "C1804", # use-implicit-booleanness-not-comparison-to-string
13   "I0011", # locally-disabled
14   "I0020", # suppressed-message
15   "R0912", # too-many-branches
16   "R0913", # too-many-arguments
17   "R0914", # too-many-locals
18   "R0915", # too-many-statements
19   "R0917", # too-many-positional-arguments
20 ]
21 enable = "all"
22 [tool pylint.reports]
23 output-format = "colorized"
```

2.40 build_and_checks_variables/repository_data.txt

```
1 Look at grep_variable to understand the structure of this file.
2 Basically, it has almost no structure apart from some lines
3 starting with a "variable_name=".
4 Anything else is comments.
5
6 repository_name=DevOrSysAdminScripts
7 repository_web_url=https://github.com/LLyaudet/DevOrSysAdminScripts/
8 repository_git_url=git@github.com:LLyaudet/DevOrSysAdminScripts.git
9
10 abstract=Some "small" useful scripts
11 abstract=for enhancing development or system-administrators tasks
12
13 license=GPLv3+
14 license2=LGPLv3+
15
16 author_email=Laurent.Lyaudet@gmail.com
17 author_full_name=Laurent Frédéric Bernard François Lyaudet
18 author_last_name=Lyaudet
19 author_name=Laurent Lyaudet
20 author_website=https://lyaudet.eu/laurent/
21
22 acknowledgments=Merci Dieu !
23 acknowledgments=Merci Père ! Merci Seigneur ! Merci Saint Esprit !
24
25 max_line_length=70
26
27 grammar_or_spell_checker_command=more
28
29
30 Why one venv for each Python dependency ?
31 Because no language of my knowledge did it right to handle all
32 possible dependency conflicts.
33 So in case of such a conflict, one specialized venv can get you out
34 of any problem.
35 The best solution would have been to allow install of different
36 versions of the same package, and have a mechanism:
37 - first intent, use the latest available version of a package among
38 those that satisfy the declared dependency constraints of the package
39 that has this dependency;
40 - second intent, check a dependency overload configuration file
41 with tree structure: if calling python package is A version 1.1.1
42 with dependency B itself with dependency C, use B 2.2.2 and C 3.3.3;
43 you could have along that another rule saying that when called
44 directly B 2.2.2 should use C 4.4.4;
```

```

45 and even more complicated:
46 if calling python package is A version 1.1.1 with dependency B and D
47 both themselves with dependency C, use B 2.2.2, D 0.1.0,
48 and C 3.3.3 for B, and C 4.4.4 for D;
49 once you've seen that it can never be more complicated than that once
50 the same version number guarantees that the code is the same,
51 you create a config language "à la CSS" for cascading dependencies ;
52 XD fit into JSON the dumbest possible way and you stop bothering
53 about dependency conflicts for the rest of history of informatics.
54 {
55     "A": {
56         "1.1.1": {
57             "B": {
58                 "2.2.2": {
59                     "C": {
60                         // Dumb, you did not found it by yourself? Failed exam.
61                         "3.3.3" : {}
62                     }
63                 }
64             },
65             "D": {
66                 "0.1.0": {
67                     "C": {
68                         // Dumb, you did not found it by yourself? Failed exam.
69                         "4.4.4" : {}
70                     }
71                 }
72             }
73         },
74         "B": {
75             "2.2.2": {
76                 "C": {
77                     // Dumb, you did not found it by yourself? Failed exam.
78                     "4.4.4" : {}
79                 }
80             }
81         }
82     }
83 Wonderful someone rediscovered nested structures XD
84 Seriously so many engineers or computer scientists are dumb to cry.
85
86 bandit_venv=../venv_bandit
87 black_venv=../venv_black
88 isort_venv=../venv_isort
89 mypy_venv=../venv_mypy
90 pylint_venv=../venv_pylint

```

```

91 ruff_venv=../venv_ruff
92
93
94 upgrade venvs can be auto or ask, any other value is interpreted
95 as ask.
96
97 upgrade_venvs=ask
98
99
100 time interval can be 0 to always upgrade or ask to upgrade,
101 or a number at most bash max int
102 https://superuser.com/questions/1030122/
103 what-is-the-maximum-value-of-a-numeric-bash-shell-variable
104 it should be enough to simulate never "upgrade or ask for upgrade"
105 unless you claim to live many billions years ;)
106 If it is wat, this is a random int XD (see RANDOM).
107 Note that the probability that it upgrades your dependencies is not
108 1/2, it increases over time, and is 1 after slightly
109 less than one day.
110 If it is watyouwant?, see SRANDOM XD.
111
112 upgrade_venvs_time_interval_in_seconds=wat
113
114
115 For the Nu HTML, CSS, and SVG checker.
116
117 vnu_jar_path=../jar_venv/vnu.jar
118
119 Not fully implemented yet, it will only warn that vnu jar was not
120 updated since some time.
121 "wat" works, but "watyouwant?" joke was not repeated ;).
122
123 upgrade_vnu_jar_time_interval_in_seconds=604800

```

2.41 build_and_checks_variables/ruff.toml

```

1 [lint]
2 select = [
3     "ALL",
4     "D211",  # blank-line-before-class no...before-class-docstring!
5 ]
6 # ignore = ["D203", "D212", "D404"]
7 ignore = [
8     "C901",  # complex-structure AKA give-me-toy-code
9     "D203",  # (want-)?incorrect-blank-line-before-class
10    "D205",  # missing-blank-line-after-summary So so cute|true ;) XD.

```

```

11 "D212",   # multi-line-summary-first-line
12 "D404",   # docstring-starts-with-this Shame! XD
13
14 # raw-string-in-exception may be useful for a real project
15 # with "dictionnaries" of errors made of constants.
16 "EM101",
17
18 # commented-out-code Of course git log is always faster...
19 # Note that if you're tired or someone else in your team is,
20 # it may catch some commit that lacked some review,
21 # or automated tests: "Last manual test succeeded, ah but when I
22 # debugged at some point I commented a part that was needed in
23 # another case."...
24 "ERA001",
25
26 "N802",   # invalid-function-name IMAP, HTML, etc.
27 "PLR0912", # too-many-branches AKA ...
28 "PLR0913", # too-many-arguments AKA ...
29 "PLR0915", # too-many-statements AKA ...
30 # Not enough reading when young makes brain slow to read and
31 # understand lengthy functions.
32 # If the project is finished, you may play with AKA rules.
33 ]
34 [lint.per-file-ignores]
35 "build_and_checks_dependencies/licenses_templates/*.py" =
36     "INP001" # implicit-namespace-package
37 ]
38 "IMAP_transfert.py" =
39     "T201", # print Use log instead in full applications.
40     "TRY003", # raise-vanilla-args
41 ]

```

2.42 build_and_checks_variables/shellcheck.ini

```

1 enable=all
2 check-sourced=true
3 external-sources=true
4 source-path=".
5 source-path="SCRIPTDIR"
6 source-path="./build_and_checks_dependencies"
7 source-path="./git_helpers"
8

```

2.43 build_and_checks_variables/shellcheck_notes.txt

```
1 A listing of the current suppressed warnings/errors from shellcheck:
2 - SC1090 : Can't follow non-constant source.
3     Use a directive to specify location
4     # shellcheck source=bla/bli.sh
5     Only remaining cases are in common_build_and_checks
6     when sourcing some Python venv
7     source venv/bin/active
8 - SC1091 : Not following (file not found most of the time).
9     Only remaining cases are in common_build_and_checks
10    when sourcing some Python venv
11    source venv/bin/active
12 - SC1111 : Use of slanted double quotes (valid inside single quotes)
13     False positive, but may catch some problem.
14 - SC2001 : Replace some sed with bash builtin ${a//b/c}
15     False positive when c contains \n.
16 - SC2034 : Unused variable
17     False positive for globals that are declared in functions
18     that are not currently called.
19     Somehow it can be used to detect dead code.
20 - SC2125 : Brace expansions and globs are literal.
21     False positive when assigning a variable declared with -i.
22     https://github.com/koalaman/shellcheck/issues/914
23 - SC2145 : Argument mixes string and array.
24     False positive when using "$@" line return and pipes.
25     https://github.com/koalaman/shellcheck/issues/3192
26 Example:
27     # shellcheck disable=SC2145
28     all_self_code_lines "$@\\"
29         | empty_lines_after_file_name
30 - SC2162 : read without -r will mangle backslashes.
31     False positive when read is used without variable:
32     https://github.com/koalaman/shellcheck/issues/623
33 - SC2248 : Prefer double quoting
34     False positive on returning integer values.
35     https://github.com/koalaman/shellcheck/issues/3405
36 - SC2312 : Some command is not invoked separately and thus its return
37     value is masked. Frequent with pipes.
38
39 The command to check if there is anything else:
40 grep -r "shellcheck disable" . | grep -v -e '.git' -e "=SC1090$"\n
41 -e "=SC1091$" -e "=SC1090,SC1091$" -e "=SC1111$" -e "=SC2001$"\n
42 -e "=SC2034$" -e "=SC2125$" -e "=SC2145$" -e "=SC2162$"\n
43 -e "=SC2248$" -e "=SC2312$" -e "=SC2162,SC2312$"
```

2.44 build_and_checks.exec.sh

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # The file "build_and_checks.exec.sh.tpl" was renamed from
25 # "build_and_checks.sh.tpl" to "build_and_checks.exec.sh.tpl".
26 # The file "build_and_checks.exec.sh" generated from the file
27 # "build_and_checks.sh.tpl" or "build_and_checks.exec.sh.tpl"
28 # was renamed from
29 # "build_and_checks.sh" to "build_and_checks.exec.sh".
30
31 build_and_checks(){
32     local LFBFL_working_directory="."
33     if [[ -n "$1" ]]; then
34         LFBFL_working_directory="$1"
35     fi
36     readonly LFBFL_working_directory
37
38     local LFBFL_verbose=""
39     if [[ "$*" == *--verbose* ]]; then
40         echo "$0 $*"
41         LFBFL_verbose="--verbose"
42     fi
43     readonly LFBFL_verbose
44 }
```

```

45     source ./wget_sha512.libr.sh
46
47 LFBFL_subdir="build_and_checks_dependencies"
48 mkdir -p "${LFBFL_subdir}/licenses_templates"
49 mkdir -p "build_and_checks_variables/temp"
50 if ! [[ -f "build_and_checks_variables/.gitignore" ]]; then
51 {
52     echo "temp/"
53     echo ""
54 } > "build_and_checks_variables/.gitignore"
55 fi
56
57 # LFBFL_dependencies_raw_content_download_URL
58 local LFBFL_dependencies_URL
59 LFBFL_dependencies_URL="https://raw.githubusercontent.com/LLyaudet/"
60 LFBFL_dependencies_URL+="DevOrSysAdminScripts/main/${LFBFL_subdir}"
61 readonly LFBFL_dependencies_URL
62
63 declare -r LFBFL_common_file_name="common_build_and_checks.exec.sh"
64 # LFBFL_script_download_URL
65 declare -r \
66     LFBFL_script="${LFBFL_dependencies_URL}/${LFBFL_common_file_name}"
67 declare -r \
68     LFBFL_file_path="./${LFBFL_subdir}/${LFBFL_common_file_name}"
69 local LFBFL_correct_sha512
70 LFBFL_correct_sha512='b838ea704913baaf45efd1b069a85c3dbc3b1a0d558a3'
71 LFBFL_correct_sha512+='c62a4b6f18b0eeaa8a553c2cd6af1990c7223ae1f8d9'
72 LFBFL_correct_sha512+='f285d40c8ada967239d5124f800f1fd19f278d1'
73 wget_sha512 "${LFBFL_file_path}" "${LFBFL_script}" \
74     "${LFBFL_correct_sha512}" "${LFBFL_verbose}"
75 chmod +x "./${LFBFL_file_path}"
76
77 "${LFBFL_file_path}" "${LFBFL_working_directory}" \
78     "${LFBFL_dependencies_URL}" "${LFBFL_verbose}"
79 }
80
81 if [[ "$*" == *--fixed_point_build* ]]; then
82     echo "--fixed_point_build"
83     source "./build_and_checks_dependencies/lines_filters.libr.sh"
84     LFBFL_working_directory=."
85     if [[ -n "$1" ]]; then
86         LFBFL_working_directory="$1"
87     fi
88     LFBFL_variables_directory="${LFBFL_working_directory}/"
89     LFBFL_variables_directory+="build_and_checks_variables"
90     LFBFL_data_file_name="${LFBFL_variables_directory}/"

```

```

91 LFBFL_data_file_name+="repository_data.txt"
92 LFBFL_repository_name=""
93 grep_variable "${LFBFL_data_file_name}" repository_name\
94   --result-variable-prefix="LFBFL_"\ \
95   --replace-line-returns-by=""
96 # Touching the 3 following files first let us iterate
97 # upgrade_build_and_checks 2 times instead of 3 to reach
98 # a fixed point.
99 touch "${LFBFL_variables_directory}/${LFBFL_repository_name}.tex"
100 touch "${LFBFL_working_directory}/${LFBFL_repository_name}.pdf"
101 touch "${LFBFL_working_directory}/${LFBFL_repository_name}.html"
102 # The two iterations of build_and_checks needs to complete
103 # during the same minute to have a fixed point.
104 build_and_checks "$@"
105 # build_and_checks "$@" second iteration below
106 # Then calling this script without --fixed_point_build should not
107 # yield any new superficial modifications.
108 fi
109
110 build_and_checks "$@"

```

2.45 COPYING

```

1      GNU GENERAL PUBLIC LICENSE
2          Version 3, 29 June 2007
3
4  Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
5  Everyone is permitted to copy and distribute verbatim copies
6  of this license document, but changing it is not allowed.
7
8          Preamble
9
10     The GNU General Public License is a free, copyleft license for
11 software and other kinds of works.
12
13     The licenses for most software and other practical works are designed
14 to take away your freedom to share and change the works. By contrast,
15 the GNU General Public License is intended to guarantee your freedom to
16 share and change all versions of a program--to make sure it remains free
17 software for all its users. We, the Free Software Foundation, use the
18 GNU General Public License for most of our software; it applies also to
19 any other work released this way by its authors. You can apply it to
20 your programs, too.
21
22     When we speak of free software, we are referring to freedom, not
23 price. Our General Public Licenses are designed to make sure that you

```

24 have the freedom to distribute copies of free software (and charge for
25 them if you wish), that you receive source code or can get it if you
26 want it, that you can change the software or use pieces of it in new
27 free programs, and that you know you can do these things.

28
29 To protect your rights, we need to prevent others from denying you
30 these rights or asking you to surrender the rights. Therefore, you have
31 certain responsibilities if you distribute copies of the software, or if
32 you modify it: responsibilities to respect the freedom of others.

33
34 For example, if you distribute copies of such a program, whether
35 gratis or for a fee, you must pass on to the recipients the same
36 freedoms that you received. You must make sure that they, too, receive
37 or can get the source code. And you must show them these terms so they
38 know their rights.

39
40 Developers that use the GNU GPL protect your rights with two steps:
41 (1) assert copyright on the software, and (2) offer you this License
42 giving you legal permission to copy, distribute and/or modify it.

43
44 For the developers' and authors' protection, the GPL clearly explains
45 that there is no warranty for this free software. For both users' and
46 authors' sake, the GPL requires that modified versions be marked as
47 changed, so that their problems will not be attributed erroneously to
48 authors of previous versions.

49
50 Some devices are designed to deny users access to install or run
51 modified versions of the software inside them, although the manufacturer
52 can do so. This is fundamentally incompatible with the aim of
53 protecting users' freedom to change the software. The systematic
54 pattern of such abuse occurs in the area of products for individuals to
55 use, which is precisely where it is most unacceptable. Therefore, we
56 have designed this version of the GPL to prohibit the practice for those
57 products. If such problems arise substantially in other domains, we
58 stand ready to extend this provision to those domains in future versions
59 of the GPL, as needed to protect the freedom of users.

60
61 Finally, every program is threatened constantly by software patents.
62 States should not allow patents to restrict development and use of
63 software on general-purpose computers, but in those that do, we wish to
64 avoid the special danger that patents applied to a free program could
65 make it effectively proprietary. To prevent this, the GPL assures that
66 patents cannot be used to render the program non-free.

67
68 The precise terms and conditions for copying, distribution and
69 modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source

116 form of a work.

117

118 A "Standard Interface" means an interface that either is an official
119 standard defined by a recognized standards body, or, in the case of
120 interfaces specified for a particular programming language, one that
121 is widely used among developers working in that language.

122

123 The "System Libraries" of an executable work include anything, other
124 than the work as a whole, that (a) is included in the normal form of
125 packaging a Major Component, but which is not part of that Major
126 Component, and (b) serves only to enable use of the work with that
127 Major Component, or to implement a Standard Interface for which an
128 implementation is available to the public in source code form. A
129 "Major Component", in this context, means a major essential component
130 (kernel, window system, and so on) of the specific operating system
131 (if any) on which the executable work runs, or a compiler used to
132 produce the work, or an object code interpreter used to run it.

133

134 The "Corresponding Source" for a work in object code form means all
135 the source code needed to generate, install, and (for an executable
136 work) run the object code and to modify the work, including scripts to
137 control those activities. However, it does not include the work's
138 System Libraries, or general-purpose tools or generally available free
139 programs which are used unmodified in performing those activities but
140 which are not part of the work. For example, Corresponding Source
141 includes interface definition files associated with source files for
142 the work, and the source code for shared libraries and dynamically
143 linked subprograms that the work is specifically designed to require,
144 such as by intimate data communication or control flow between those
145 subprograms and other parts of the work.

146

147 The Corresponding Source need not include anything that users
148 can regenerate automatically from other parts of the Corresponding
149 Source.

150

151 The Corresponding Source for a work in source code form is that
152 same work.

153

154 2. Basic Permissions.

155

156 All rights granted under this License are granted for the term of
157 copyright on the Program, and are irrevocable provided the stated
158 conditions are met. This License explicitly affirms your unlimited
159 permission to run the unmodified Program. The output from running a
160 covered work is covered by this License only if the output, given its
161 content, constitutes a covered work. This License acknowledges your

162 rights of fair use or other equivalent, as provided by copyright law.

163

164 You may make, run and propagate covered works that you do not
165 convey, without conditions so long as your license otherwise remains
166 in force. You may convey covered works to others for the sole purpose
167 of having them make modifications exclusively for you, or provide you
168 with facilities for running those works, provided that you comply with
169 the terms of this License in conveying all material for which you do
170 not control copyright. Those thus making or running the covered works
171 for you must do so exclusively on your behalf, under your direction
172 and control, on terms that prohibit them from making any copies of
173 your copyrighted material outside their relationship with you.

174

175 Conveying under any other circumstances is permitted solely under
176 the conditions stated below. Sublicensing is not allowed; section 10
177 makes it unnecessary.

178

179 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

180

181 No covered work shall be deemed part of an effective technological
182 measure under any applicable law fulfilling obligations under article
183 11 of the WIPO copyright treaty adopted on 20 December 1996, or
184 similar laws prohibiting or restricting circumvention of such
185 measures.

186

187 When you convey a covered work, you waive any legal power to forbid
188 circumvention of technological measures to the extent such circumvention
189 is effected by exercising rights under this License with respect to
190 the covered work, and you disclaim any intention to limit operation or
191 modification of the work as a means of enforcing, against the work's
192 users, your or third parties' legal rights to forbid circumvention of
193 technological measures.

194

195 4. Conveying Verbatim Copies.

196

197 You may convey verbatim copies of the Program's source code as you
198 receive it, in any medium, provided that you conspicuously and
199 appropriately publish on each copy an appropriate copyright notice;
200 keep intact all notices stating that this License and any
201 non-permissive terms added in accord with section 7 apply to the code;
202 keep intact all notices of the absence of any warranty; and give all
203 recipients a copy of this License along with the Program.

204

205 You may charge any price or no price for each copy that you convey,
206 and you may offer support or warranty protection for a fee.

207

208 5. Conveying Modified Source Versions.

209
210 You may convey a work based on the Program, or the modifications to
211 produce it from the Program, in the form of source code under the
212 terms of section 4, provided that you also meet all of these conditions:

213
214 a) The work must carry prominent notices stating that you modified
215 it, and giving a relevant date.

216
217 b) The work must carry prominent notices stating that it is
218 released under this License and any conditions added under section
219 7. This requirement modifies the requirement in section 4 to
220 "keep intact all notices".

221
222 c) You must license the entire work, as a whole, under this
223 License to anyone who comes into possession of a copy. This
224 License will therefore apply, along with any applicable section 7
225 additional terms, to the whole of the work, and all its parts,
226 regardless of how they are packaged. This License gives no
227 permission to license the work in any other way, but it does not
228 invalidate such permission if you have separately received it.

229
230 d) If the work has interactive user interfaces, each must display
231 Appropriate Legal Notices; however, if the Program has interactive
232 interfaces that do not display Appropriate Legal Notices, your
233 work need not make them do so.

234
235 A compilation of a covered work with other separate and independent
236 works, which are not by their nature extensions of the covered work,
237 and which are not combined with it such as to form a larger program,
238 in or on a volume of a storage or distribution medium, is called an
239 "aggregate" if the compilation and its resulting copyright are not
240 used to limit the access or legal rights of the compilation's users
241 beyond what the individual works permit. Inclusion of a covered work
242 in an aggregate does not cause this License to apply to the other
243 parts of the aggregate.

244
245 6. Conveying Non-Source Forms.

246
247 You may convey a covered work in object code form under the terms
248 of sections 4 and 5, provided that you also convey the
249 machine-readable Corresponding Source under the terms of this License,
250 in one of these ways:

251
252 a) Convey the object code in, or embodied in, a physical product
253 (including a physical distribution medium), accompanied by the

254 Corresponding Source fixed on a durable physical medium
255 customarily used for software interchange.

256

257 b) Convey the object code in, or embodied in, a physical product
258 (including a physical distribution medium), accompanied by a
259 written offer, valid for at least three years and valid for as
260 long as you offer spare parts or customer support for that product
261 model, to give anyone who possesses the object code either (1) a
262 copy of the Corresponding Source for all the software in the
263 product that is covered by this License, on a durable physical
264 medium customarily used for software interchange, for a price no
265 more than your reasonable cost of physically performing this
266 conveying of source, or (2) access to copy the
267 Corresponding Source from a network server at no charge.

268

269 c) Convey individual copies of the object code with a copy of the
270 written offer to provide the Corresponding Source. This
271 alternative is allowed only occasionally and noncommercially, and
272 only if you received the object code with such an offer, in accord
273 with subsection 6b.

274

275 d) Convey the object code by offering access from a designated
276 place (gratis or for a charge), and offer equivalent access to the
277 Corresponding Source in the same way through the same place at no
278 further charge. You need not require recipients to copy the
279 Corresponding Source along with the object code. If the place to
280 copy the object code is a network server, the Corresponding Source
281 may be on a different server (operated by you or a third party)
282 that supports equivalent copying facilities, provided you maintain
283 clear directions next to the object code saying where to find the
284 Corresponding Source. Regardless of what server hosts the
285 Corresponding Source, you remain obligated to ensure that it is
286 available for as long as needed to satisfy these requirements.

287

288 e) Convey the object code using peer-to-peer transmission, provided
289 you inform other peers where the object code and Corresponding
290 Source of the work are being offered to the general public at no
291 charge under subsection 6d.

292

293 A separable portion of the object code, whose source code is excluded
294 from the Corresponding Source as a System Library, need not be
295 included in conveying the object code work.

296

297 A "User Product" is either (1) a "consumer product", which means any
298 tangible personal property which is normally used for personal, family,
299 or household purposes, or (2) anything designed or sold for incorporation

300 into a dwelling. In determining whether a product is a consumer product,
301 doubtful cases shall be resolved in favor of coverage. For a particular
302 product received by a particular user, "normally used" refers to a
303 typical or common use of that class of product, regardless of the status
304 of the particular user or of the way in which the particular user
305 actually uses, or expects or is expected to use, the product. A product
306 is a consumer product regardless of whether the product has substantial
307 commercial, industrial or non-consumer uses, unless such uses represent
308 the only significant mode of use of the product.

309
310 "Installation Information" for a User Product means any methods,
311 procedures, authorization keys, or other information required to install
312 and execute modified versions of a covered work in that User Product from
313 a modified version of its Corresponding Source. The information must
314 suffice to ensure that the continued functioning of the modified object
315 code is in no case prevented or interfered with solely because
316 modification has been made.

317
318 If you convey an object code work under this section in, or with, or
319 specifically for use in, a User Product, and the conveying occurs as
320 part of a transaction in which the right of possession and use of the
321 User Product is transferred to the recipient in perpetuity or for a
322 fixed term (regardless of how the transaction is characterized), the
323 Corresponding Source conveyed under this section must be accompanied
324 by the Installation Information. But this requirement does not apply
325 if neither you nor any third party retains the ability to install
326 modified object code on the User Product (for example, the work has
327 been installed in ROM).

328
329 The requirement to provide Installation Information does not include a
330 requirement to continue to provide support service, warranty, or updates
331 for a work that has been modified or installed by the recipient, or for
332 the User Product in which it has been modified or installed. Access to a
333 network may be denied when the modification itself materially and
334 adversely affects the operation of the network or violates the rules and
335 protocols for communication across the network.

336
337 Corresponding Source conveyed, and Installation Information provided,
338 in accord with this section must be in a format that is publicly
339 documented (and with an implementation available to the public in
340 source code form), and must require no special password or key for
341 unpacking, reading or copying.

342
343 7. Additional Terms.

344
345 "Additional permissions" are terms that supplement the terms of this

346 License by making exceptions from one or more of its conditions.
347 Additional permissions that are applicable to the entire Program shall
348 be treated as though they were included in this License, to the extent
349 that they are valid under applicable law. If additional permissions
350 apply only to part of the Program, that part may be used separately
351 under those permissions, but the entire Program remains governed by
352 this License without regard to the additional permissions.

353
354 When you convey a copy of a covered work, you may at your option
355 remove any additional permissions from that copy, or from any part of
356 it. (Additional permissions may be written to require their own
357 removal in certain cases when you modify the work.) You may place
358 additional permissions on material, added by you to a covered work,
359 for which you have or can give appropriate copyright permission.

360
361 Notwithstanding any other provision of this License, for material you
362 add to a covered work, you may (if authorized by the copyright holders of
363 that material) supplement the terms of this License with terms:

- 364
- 365 a) Disclaiming warranty or limiting liability differently from the
366 terms of sections 15 and 16 of this License; or
 - 367
 - 368 b) Requiring preservation of specified reasonable legal notices or
369 author attributions in that material or in the Appropriate Legal
370 Notices displayed by works containing it; or
 - 371
 - 372 c) Prohibiting misrepresentation of the origin of that material, or
373 requiring that modified versions of such material be marked in
374 reasonable ways as different from the original version; or
 - 375
 - 376 d) Limiting the use for publicity purposes of names of licensors or
377 authors of the material; or
 - 378
 - 379 e) Declining to grant rights under trademark law for use of some
380 trade names, trademarks, or service marks; or
 - 381
 - 382 f) Requiring indemnification of licensors and authors of that
383 material by anyone who conveys the material (or modified versions of
384 it) with contractual assumptions of liability to the recipient, for
385 any liability that these contractual assumptions directly impose on
386 those licensors and authors.

387
388 All other non-permissive additional terms are considered "further
389 restrictions" within the meaning of section 10. If the Program as you
390 received it, or any part of it, contains a notice stating that it is
391 governed by this License along with a term that is a further

392 restriction, you may remove that term. If a license document contains
393 a further restriction but permits relicensing or conveying under this
394 License, you may add to a covered work material governed by the terms
395 of that license document, provided that the further restriction does
396 not survive such relicensing or conveying.

397
398 If you add terms to a covered work in accord with this section, you
399 must place, in the relevant source files, a statement of the
400 additional terms that apply to those files, or a notice indicating
401 where to find the applicable terms.

402
403 Additional terms, permissive or non-permissive, may be stated in the
404 form of a separately written license, or stated as exceptions;
405 the above requirements apply either way.

406
407 8. Termination.

408
409 You may not propagate or modify a covered work except as expressly
410 provided under this License. Any attempt otherwise to propagate or
411 modify it is void, and will automatically terminate your rights under
412 this License (including any patent licenses granted under the third
413 paragraph of section 11).

414
415 However, if you cease all violation of this License, then your
416 license from a particular copyright holder is reinstated (a)
417 provisionally, unless and until the copyright holder explicitly and
418 finally terminates your license, and (b) permanently, if the copyright
419 holder fails to notify you of the violation by some reasonable means
420 prior to 60 days after the cessation.

421
422 Moreover, your license from a particular copyright holder is
423 reinstated permanently if the copyright holder notifies you of the
424 violation by some reasonable means, this is the first time you have
425 received notice of violation of this License (for any work) from that
426 copyright holder, and you cure the violation prior to 30 days after
427 your receipt of the notice.

428
429 Termination of your rights under this section does not terminate the
430 licenses of parties who have received copies or rights from you under
431 this License. If your rights have been terminated and not permanently
432 reinstated, you do not qualify to receive new licenses for the same
433 material under section 10.

434
435 9. Acceptance Not Required for Having Copies.

436
437 You are not required to accept this License in order to receive or

438 run a copy of the Program. Ancillary propagation of a covered work
439 occurring solely as a consequence of using peer-to-peer transmission
440 to receive a copy likewise does not require acceptance. However,
441 nothing other than this License grants you permission to propagate or
442 modify any covered work. These actions infringe copyright if you do
443 not accept this License. Therefore, by modifying or propagating a
444 covered work, you indicate your acceptance of this License to do so.

445

446 10. Automatic Licensing of Downstream Recipients.

447

448 Each time you convey a covered work, the recipient automatically
449 receives a license from the original licensors, to run, modify and
450 propagate that work, subject to this License. You are not responsible
451 for enforcing compliance by third parties with this License.

452

453 An "entity transaction" is a transaction transferring control of an
454 organization, or substantially all assets of one, or subdividing an
455 organization, or merging organizations. If propagation of a covered
456 work results from an entity transaction, each party to that
457 transaction who receives a copy of the work also receives whatever
458 licenses to the work the party's predecessor in interest had or could
459 give under the previous paragraph, plus a right to possession of the
460 Corresponding Source of the work from the predecessor in interest, if
461 the predecessor has it or can get it with reasonable efforts.

462

463 You may not impose any further restrictions on the exercise of the
464 rights granted or affirmed under this License. For example, you may
465 not impose a license fee, royalty, or other charge for exercise of
466 rights granted under this License, and you may not initiate litigation
467 (including a cross-claim or counterclaim in a lawsuit) alleging that
468 any patent claim is infringed by making, using, selling, offering for
469 sale, or importing the Program or any portion of it.

470

471 11. Patents.

472

473 A "contributor" is a copyright holder who authorizes use under this
474 License of the Program or a work on which the Program is based. The
475 work thus licensed is called the contributor's "contributor version".

476

477 A contributor's "essential patent claims" are all patent claims
478 owned or controlled by the contributor, whether already acquired or
479 hereafter acquired, that would be infringed by some manner, permitted
480 by this License, of making, using, or selling its contributor version,
481 but do not include claims that would be infringed only as a
482 consequence of further modification of the contributor version. For
483 purposes of this definition, "control" includes the right to grant

484 patent sublicenses in a manner consistent with the requirements of
485 this License.

486
487 Each contributor grants you a non-exclusive, worldwide, royalty-free
488 patent license under the contributor's essential patent claims, to
489 make, use, sell, offer for sale, import and otherwise run, modify and
490 propagate the contents of its contributor version.

491
492 In the following three paragraphs, a "patent license" is any express
493 agreement or commitment, however denominated, not to enforce a patent
494 (such as an express permission to practice a patent or covenant not to
495 sue for patent infringement). To "grant" such a patent license to a
496 party means to make such an agreement or commitment not to enforce a
497 patent against the party.

498
499 If you convey a covered work, knowingly relying on a patent license,
500 and the Corresponding Source of the work is not available for anyone
501 to copy, free of charge and under the terms of this License, through a
502 publicly available network server or other readily accessible means,
503 then you must either (1) cause the Corresponding Source to be so
504 available, or (2) arrange to deprive yourself of the benefit of the
505 patent license for this particular work, or (3) arrange, in a manner
506 consistent with the requirements of this License, to extend the patent
507 license to downstream recipients. "Knowingly relying" means you have
508 actual knowledge that, but for the patent license, your conveying the
509 covered work in a country, or your recipient's use of the covered work
510 in a country, would infringe one or more identifiable patents in that
511 country that you have reason to believe are valid.

512
513 If, pursuant to or in connection with a single transaction or
514 arrangement, you convey, or propagate by procuring conveyance of, a
515 covered work, and grant a patent license to some of the parties
516 receiving the covered work authorizing them to use, propagate, modify
517 or convey a specific copy of the covered work, then the patent license
518 you grant is automatically extended to all recipients of the covered
519 work and works based on it.

520
521 A patent license is "discriminatory" if it does not include within
522 the scope of its coverage, prohibits the exercise of, or is
523 conditioned on the non-exercise of one or more of the rights that are
524 specifically granted under this License. You may not convey a covered
525 work if you are a party to an arrangement with a third party that is
526 in the business of distributing software, under which you make payment
527 to the third party based on the extent of your activity of conveying
528 the work, and under which the third party grants, to any of the
529 parties who would receive the covered work from you, a discriminatory

530 patent license (a) in connection with copies of the covered work
531 conveyed by you (or copies made from those copies), or (b) primarily
532 for and in connection with specific products or compilations that
533 contain the covered work, unless you entered into that arrangement,
534 or that patent license was granted, prior to 28 March 2007.

535
536 Nothing in this License shall be construed as excluding or limiting
537 any implied license or other defenses to infringement that may
538 otherwise be available to you under applicable patent law.

539
540 12. No Surrender of Others' Freedom.

541
542 If conditions are imposed on you (whether by court order, agreement or
543 otherwise) that contradict the conditions of this License, they do not
544 excuse you from the conditions of this License. If you cannot convey a
545 covered work so as to satisfy simultaneously your obligations under this
546 License and any other pertinent obligations, then as a consequence you may
547 not convey it at all. For example, if you agree to terms that obligate you
548 to collect a royalty for further conveying from those to whom you convey
549 the Program, the only way you could satisfy both those terms and this
550 License would be to refrain entirely from conveying the Program.

551
552 13. Use with the GNU Affero General Public License.

553
554 Notwithstanding any other provision of this License, you have
555 permission to link or combine any covered work with a work licensed
556 under version 3 of the GNU Affero General Public License into a single
557 combined work, and to convey the resulting work. The terms of this
558 License will continue to apply to the part which is the covered work,
559 but the special requirements of the GNU Affero General Public License,
560 section 13, concerning interaction through a network will apply to the
561 combination as such.

562
563 14. Revised Versions of this License.

564
565 The Free Software Foundation may publish revised and/or new versions of
566 the GNU General Public License from time to time. Such new versions will
567 be similar in spirit to the present version, but may differ in detail to
568 address new problems or concerns.

569
570 Each version is given a distinguishing version number. If the
571 Program specifies that a certain numbered version of the GNU General
572 Public License "or any later version" applies to it, you have the
573 option of following the terms and conditions either of that numbered
574 version or of any later version published by the Free Software
575 Foundation. If the Program does not specify a version number of the

576 GNU General Public License, you may choose any version ever published
577 by the Free Software Foundation.

578
579 If the Program specifies that a proxy can decide which future
580 versions of the GNU General Public License can be used, that proxy's
581 public statement of acceptance of a version permanently authorizes you
582 to choose that version for the Program.

583
584 Later license versions may give you additional or different
585 permissions. However, no additional obligations are imposed on any
586 author or copyright holder as a result of your choosing to follow a
587 later version.

588
589 15. Disclaimer of Warranty.

590
591 THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
592 APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
593 HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
594 OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
595 THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
596 PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
597 IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
598 ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

599
600 16. Limitation of Liability.

601
602 IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
603 WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
604 THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
605 GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
606 USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
607 DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
608 PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
609 EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
610 SUCH DAMAGES.

611
612 17. Interpretation of Sections 15 and 16.

613
614 If the disclaimer of warranty and limitation of liability provided
615 above cannot be given local legal effect according to their terms,
616 reviewing courts shall apply local law that most closely approximates
617 an absolute waiver of all civil liability in connection with the
618 Program, unless a warranty or assumption of liability accompanies a
619 copy of the Program in return for a fee.

620
621 END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<https://www.gnu.org/licenses/>>.

668
669 The GNU General Public License does not permit incorporating your program
670 into proprietary programs. If your program is a subroutine library, you
671 may consider it more useful to permit linking proprietary applications with
672 the library. If this is what you want to do, use the GNU Lesser General
673 Public License instead of this License. But first, please read
674 <<https://www.gnu.org/licenses/why-not-lgpl.html>>.

2.46 COPYING.LESSER

37 The "Corresponding Application Code" for a Combined Work means the
38 object code and/or source code for the Application, including any data
39 and utility programs needed for reproducing the Combined Work from the
40 Application, but excluding the System Libraries of the Combined Work.

41

42 1. Exception to Section 3 of the GNU GPL.

43

44 You may convey a covered work under sections 3 and 4 of this License
45 without being bound by section 3 of the GNU GPL.

46

47 2. Conveying Modified Versions.

48

49 If you modify a copy of the Library, and, in your modifications, a
50 facility refers to a function or data to be supplied by an Application
51 that uses the facility (other than as an argument passed when the
52 facility is invoked), then you may convey a copy of the modified
53 version:

54

55 a) under this License, provided that you make a good faith effort to
56 ensure that, in the event an Application does not supply the
57 function or data, the facility still operates, and performs
58 whatever part of its purpose remains meaningful, or

59

60 b) under the GNU GPL, with none of the additional permissions of
61 this License applicable to that copy.

62

63 3. Object Code Incorporating Material from Library Header Files.

64

65 The object code form of an Application may incorporate material from
66 a header file that is part of the Library. You may convey such object
67 code under terms of your choice, provided that, if the incorporated
68 material is not limited to numerical parameters, data structure
69 layouts and accessors, or small macros, inline functions and templates
70 (ten or fewer lines in length), you do both of the following:

71

72 a) Give prominent notice with each copy of the object code that the
73 Library is used in it and that the Library and its use are
74 covered by this License.

75

76 b) Accompany the object code with a copy of the GNU GPL and this license
77 document.

78

79 4. Combined Works.

80

81 You may convey a Combined Work under terms of your choice that,
82 taken together, effectively do not restrict modification of the

83 portions of the Library contained in the Combined Work and reverse
84 engineering for debugging such modifications, if you also do each of
85 the following:

86

87 a) Give prominent notice with each copy of the Combined Work that
88 the Library is used in it and that the Library and its use are
89 covered by this License.

90

91 b) Accompany the Combined Work with a copy of the GNU GPL and this license
92 document.

93

94 c) For a Combined Work that displays copyright notices during
95 execution, include the copyright notice for the Library among
96 these notices, as well as a reference directing the user to the
97 copies of the GNU GPL and this license document.

98

99 d) Do one of the following:

100

101 0) Convey the Minimal Corresponding Source under the terms of this
102 License, and the Corresponding Application Code in a form
103 suitable for, and under terms that permit, the user to
104 recombine or relink the Application with a modified version of
105 the Linked Version to produce a modified Combined Work, in the
106 manner specified by section 6 of the GNU GPL for conveying
107 Corresponding Source.

108

109 1) Use a suitable shared library mechanism for linking with the
110 Library. A suitable mechanism is one that (a) uses at run time
111 a copy of the Library already present on the user's computer
112 system, and (b) will operate properly with a modified version
113 of the Library that is interface-compatible with the Linked
114 Version.

115

116 e) Provide Installation Information, but only if you would otherwise
117 be required to provide such information under section 6 of the
118 GNU GPL, and only to the extent that such information is
119 necessary to install and execute a modified version of the
120 Combined Work produced by recombining or relinking the
121 Application with a modified version of the Linked Version. (If
122 you use option 4d0, the Installation Information must accompany
123 the Minimal Corresponding Source and Corresponding Application
124 Code. If you use option 4d1, you must provide the Installation
125 Information in the manner specified by section 6 of the GNU GPL
126 for conveying Corresponding Source.)

127

128 5. Combined Libraries.

129
 130 You may place library facilities that are a work based on the
 131 Library side by side in a single library together with other library
 132 facilities that are not Applications and are not covered by this
 133 License, and convey such a combined library under terms of your
 134 choice, if you do both of the following:
 135
 136 a) Accompany the combined library with a copy of the same work based
 137 on the Library, uncombined with any other library facilities,
 138 conveyed under the terms of this License.
 139
 140 b) Give prominent notice with the combined library that part of it
 141 is a work based on the Library, and explaining where to find the
 142 accompanying uncombined form of the same work.
 143
 144 6. Revised Versions of the GNU Lesser General Public License.
 145
 146 The Free Software Foundation may publish revised and/or new versions
 147 of the GNU Lesser General Public License from time to time. Such new
 148 versions will be similar in spirit to the present version, but may
 149 differ in detail to address new problems or concerns.
 150
 151 Each version is given a distinguishing version number. If the
 152 Library as you received it specifies that a certain numbered version
 153 of the GNU Lesser General Public License "or any later version"
 154 applies to it, you have the option of following the terms and
 155 conditions either of that published version or of any later version
 156 published by the Free Software Foundation. If the Library as you
 157 received it does not specify a version number of the GNU Lesser
 158 General Public License, you may choose any version of the GNU Lesser
 159 General Public License ever published by the Free Software Foundation.
 160
 161 If the Library as you received it specifies that a proxy can decide
 162 whether future versions of the GNU Lesser General Public License shall
 163 apply, that proxy's public statement of acceptance of any version is
 164 permanent authorization for you to choose that version for the
 165 Library.

2.47 git_helpers/git_checks.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License

```

```

7  # as published by the Free Software Foundation,
8  # either version 3 of the License,
9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24 # 2024-05-31T01:57+02:00: This file was renamed from
25 # "git_checks.libr.sh"
26 # to
27 # "git_checks.libr.sh".
28 # 2024-05-31T01:57+02:00: This file was moved from
29 # "."
30 # to
31 # "./git_helpers".
32 # 2024-05-31T01:49+02:00: This file was renamed from
33 # "git_checks.libr.sh"
34 # to
35 # "git_checks.libr.sh".
36 # 2024-05-31T01:49+02:00: This file was moved from
37 # "."
38 # to
39 # "./git_helpers".
40 # This file was renamed from "git_checks.sh" to "git_checks.libr.sh".
41
42 check_files(){
43     # Checks used on Django projects.
44     declare -r LFBFL_send_summary_1="ATTENTION : models modifiés"
45     declare -r LFBFL_send_body_1="Vérifiez si besoin de migrations"
46     local LFBFL_send_body_2="Pensez aussi aux contraintes d'unicités"
47     LFBFL_send_body_2+=" dès la création du modèle"
48     readonly LFBFL_send_body_2
49     declare -r LFBFL_send_summary_2="ATTENTION : serializers modifiés"
50     declare -r LFBFL_send_body_3=\
51 "Les nouveaux preloadings sont interdits -> #Prefetch()"
52     git diff --cached --name-only\
```

```

53      > /tmp/DOSAS_django_git_check1.temp
54 if grep models /tmp/DOSAS_django_git_check1.temp;
55 then
56     notify-send "${LFBFL_send_summary_1}" "${LFBFL_send_body_1}"
57     notify-send "${LFBFL_send_summary_1}" "${LFBFL_send_body_2}"
58 fi
59 if grep serializer /tmp/DOSAS_django_git_check1.temp;
60 then
61     notify-send "${LFBFL_send_summary_2}" "${LFBFL_send_body_3}"
62 fi
63 rm /tmp/DOSAS_django_git_check1.temp
64 return 0
65 }
66
67
68 check_no_abusive_trailing_comma() {
69     # Useful for Python code, although it regularly yields
70     # false positives in Python code
71     # (but it is easy to distinguish by hand).
72     # Use `git commit --no-verify` after check.
73     declare -r LFBFL_send_summary_1="ATTENTION"
74     local LFBFL_send_body_1="Il semblerait que vous affectiez un tuple"
75     LFBFL_send_body_1+=" au lieu d'une autre valeur dans une variable."
76     readonly LFBFL_send_body_1
77     git diff --cached -r \
78         > /tmp/DOSAS_django_git_check2.temp
79     if grep ' = .*,\s*$' /tmp/DOSAS_django_git_check2.temp;
80     then
81         notify-send "${LFBFL_send_summary_1}" "${LFBFL_send_body_1}"
82         return 1
83     fi
84     rm /tmp/DOSAS_django_git_check2.temp
85     return 0
86 }
87
88
89 check_black_code_formatting() {
90     # attention ça ne marche que sur les fichiers "stagés" avec git add
91     declare -r LFBFL_files_string=$(
92         git diff --cached --name-only \
93         | grep '\.py'
94     )
95     echo "${LFBFL_files_string}"
96     declare -a LFBFL_some_files
97     mapfile -t LFBFL_some_files <<< "${LFBFL_files_string}"
98     readonly LFBFL_some_files

```

```

99 declare -i LFBFL_error=0
100 # Not very useful since LFBFL_file should be local to the loop by
101 # default.
102 local LFBFL_file
103 for LFBFL_file in "${LFBFL_some_files[@]}"; do
104     echo "Black will check formatting on file ${LFBFL_file}"
105     if ! black --check --diff "${LFBFL_file}";
106     then
107         LFBFL_error=1
108     fi
109 done
110 # shellcheck disable=SC2248
111 return ${LFBFL_error}
112 }
```

2.48 git_helpers/post-commit

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2024 Laurent Frédéric Bernard François Lyaudet
24 # 2024-05-31T01:49+02:00: This file was renamed from
25 # "post-commit"
26 # to
27 # "post-commit".
28 # 2024-05-31T01:49+02:00: This file was moved from
29 # "."
```

```

30  # to
31  # "./git_helpers".
32
33 LFBFL_library="../DevOrSysAdminScripts/git_helpers/"
34 LFBFL_library+="post_commit_renaming_comment.libr.sh"
35 # shellcheck source=post_commit_renaming_comment.libr.sh
36 source "${LFBFL_library}"
37 LFBFL_library="../DevOrSysAdminScripts/build_and_checks_dependencies/"
38 LFBFL_library+="lines_filters.libr.sh"
39 # shellcheck source=lines_filters.libr.sh
40 source "${LFBFL_library}"
41
42 LFBFL_data_file_name="build_and_checks_variables/repository_data.txt"
43 LFBFL_max_comment_line_length=""
44 if [[ -f "${LFBFL_data_file_name}" ]]; then
45     LFBFL_max_line_length=""
46     grep_variable "${LFBFL_data_file_name}" "max_line_length" \
47         --result-variable-prefix="LFBFL_"
48     LFBFL_max_comment_line_length="--max-comment-line-length="
49     LFBFL_max_comment_line_length+="${LFBFL_max_line_length}"
50 fi
51
52 commit_a_file_renamed_comment --verbose --log-directory-change\
53     "${LFBFL_max_comment_line_length}"
54 exit 0

```

2.49 git_helpers/post_commit_renaming_comment.libr.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of

```

```

19  # the GNU General Public License
20  # along with DevOrSysAdminScripts.
21  # If not, see <https://www.gnu.org/licenses/>.
22  #
23  # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24  # 2024-05-31T01:49+02:00: This file was renamed from
25  # "post_commit_renaming_comment.libr.sh"
26  # to
27  # "post_commit_renaming_comment.libr.sh".
28  # 2024-05-31T01:49+02:00: This file was moved from
29  # "."
30  # to
31  # "./git_helpers".
32  # This file was renamed from "post_commit_renaming_comment"
33  # to "post_commit_renaming_comment.libr.sh".
34  # (Before first commit it seems. But I had the old name in listing.)
35
36 LFBFL_this_file_path=$(realpath "${BASH_SOURCE[0]}")
37 LFBFL_this_file_directory=$(dirname "${LFBFL_this_file_path}")
38 LFBFL_some_directory="${LFBFL_this_file_directory}/.."
39 declare -i LFBFL_cd_result2
40 pushd .
41 cd "${LFBFL_some_directory}" || {
42     LFBFL_cd_result2=$?
43     echo "post_commit_renaming_comment.libr.sh no such directory"
44     # shellcheck disable=SC2248
45     exit ${LFBFL_cd_result2}
46 }
47 # shellcheck source=strings_functions.libr.sh
48 source "build_and_checks_dependencies/strings_functions.libr.sh"
49 declare -i LFBFL_popd_result2
50 popd || {
51     LFBFL_popd_result2=$?
52     echo "post_commit_renaming_comment.libr.sh no popd"
53     # shellcheck disable=SC2248
54     exit ${LFBFL_popd_result2}
55 }
56
57 # Never forget that the worse security is the illusion of security
58 # (Kevin Mitnick). So, do not expect to have an accurate trackability
59 # of code history just because you use this post-commit hook or
60 # something else more evolved. This is useful but not a guarantee.
61 # For this reason, this function will only handle when all files
62 # impacted in the preceding commit are only renamed.
63 # It assumes that 'diff' is the current default difference utility
64 # parametered with 'git'. This point could be enhanced.

```

```

65  commit_a_file_renamed_comment () {
66      local LFBFL_arg
67      declare -i LFBFL_verbose=0
68      if [[ "$*" == *--verbose* ]]; then
69          echo "$0 commit_a_file_renamed_comment $*"
70          LFBFL_verbose=1
71      fi
72      readonly LFBFL_verbose
73
74      if [[ ! -o pipefail ]]; then
75          [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
76          set -o pipefail
77          trap 'set +o pipefail' RETURN
78      fi
79
80      declare -i LFBFL_log_directory_change=0
81      if [[ "$*" == *--log-directory-change* ]]; then
82          LFBFL_log_directory_change=1
83      fi
84      readonly LFBFL_log_directory_change
85
86      declare -i LFBFL_max_comment_line_length=70
87      for LFBFL_arg in "$@"; do
88          if [[ "${LFBFL_arg}" == --max-comment-line-length=* ]]; then
89              LFBFL_arg=${LFBFL_arg#--max-comment-line-length=}
90              LFBFL_max_comment_line_length=$((LFBFL_arg))
91              break
92          fi
93      done
94      readonly LFBFL_max_comment_line_length
95
96      if [[ LFBFL_verbose -eq 1 ]]; then
97          echo "LFBFL_log_directory_change: ${LFBFL_log_directory_change}"
98          echo "LFBFL_max_comment_line_length:" \
99              "${LFBFL_max_comment_line_length}"
100     fi
101 #-----
102
103     declare -r LFBFL_renaming_lines=$((
104         git log -p -1 \
105         | grep '^diff --git' -A 1
106     ))
107     if [[ LFBFL_verbose -eq 1 ]]; then
108         echo "LFBFL_renaming_lines: ${LFBFL_renaming_lines}"
109     fi
110

```

```

111 declare -a LFBFL_renaming_lines_array
112 mapfile -t LFBFL_renaming_lines_array <<< "${LFBFL_renaming_lines}"
113 readonly LFBFL_renaming_lines_array
114
115 declare -r LFBFL_timestamp=$(date --iso-8601="minute")
116 if [[ LFBFL_verbose -eq 1 ]]; then
117     echo "LFBFL_timestamp: ${LFBFL_timestamp}"
118 fi
119 get_split_score_simple 1 "${LFBFL_max_comment_line_length}" \
120 LFBFL_split_score_command1="${get_split_score_result}"
121 LFBFL_split_score_command_properties1="${get_split_score_result2}"
122 get_split_score_simple 1 "${LFBFL_max_comment_line_length}" /
123 LFBFL_split_score_command2="${get_split_score_result}"
124 LFBFL_split_score_command_properties2="${get_split_score_result2}"
125 declare -i LFBFL_renaming_happened=0
126 declare -i LFBFL_i
127 declare -i LFBFL_j
128 # declare -r LFBFL_special_file_name="files_names_listing.txt"
129 declare -r LFBFL_sfn="files_names_listing.txt"
130 local LFBFL_diff_line
131 local LFBFL_similarity_line
132 local LFBFL_old_file_path
133 local LFBFL_old_file_name
134 local LFBFL_old_file_directory
135 local LFBFL_new_file_path
136 local LFBFL_new_file_name
137 local LFBFL_new_file_directory
138 local LFBFL_useful_file_name
139 local LFBFL_extension
140 local LFBFL_comment_prefix
141 local LFBFL_comment_prefix2
142 local LFBFL_old_file_name2
143 local LFBFL_old_file_directory2
144 local LFBFL_new_file_name2
145 local LFBFL_new_file_directory2
146 local LFBFL_new_comment
147 declare -i LFBFL_copyright_line_number
148 declare -r LFBFL_send_summary_2="ATTENTION: No line with copyright"
149 declare -i LFBFL_line_count
150 declare -i LFBFL_lines_after
151 local LFBFL_temp_file_path
152 for ((LFBFL_i=0; LFBFL_i<#${LFBFL_renaming_lines_array[@]};)); do
153     LFBFL_diff_line="${LFBFL_renaming_lines_array[$LFBFL_i]}"
154     LFBFL_i=$((LFBFL_i + 1))
155     LFBFL_similarity_line="${LFBFL_renaming_lines_array[$LFBFL_i]}"
156     LFBFL_i=$((LFBFL_i + 1))

```

```

157 # We need to increment 3 times since grep adds "--" lines between.
158 LFBFL_i=$((LFBFL_i + 1))
159 if [[ LFBFL_verbose -eq 1 ]]; then
160     echo "LFBFL_diff_line: ${LFBFL_diff_line}"
161     echo "LFBFL_similarity_line: ${LFBFL_similarity_line}"
162 fi
163 if [[ "${LFBFL_similarity_line:17:4}" != "100%" ]]; then
164     continue
165 fi
166
167 # Extract old file name from diff line. -----
168 LFBFL_old_file_path=$((
169     echo "${LFBFL_diff_line}" \
170     | cut -d ' ' -f 3
171 ))
172 # It starts by "a/".
173 LFBFL_old_file_path=".${LFBFL_old_file_path:1}"
174 LFBFL_old_file_name=$(basename "${LFBFL_old_file_path}")
175 LFBFL_old_file_directory=$(dirname "${LFBFL_old_file_path}")
176 # Extract new file name from diff line. -----
177 LFBFL_new_file_path=$((
178     echo "${LFBFL_diff_line}" \
179     | cut -d ' ' -f 4
180 ))
181 # It starts by "b/".
182 LFBFL_new_file_path=".${LFBFL_new_file_path:1}"
183 LFBFL_new_file_name=$(basename "${LFBFL_new_file_path}")
184 LFBFL_new_file_directory=$(dirname "${LFBFL_new_file_path}")
185 if [[ LFBFL_verbose -eq 1 ]]; then
186     echo "LFBFL_old_file_path: ${LFBFL_old_file_path}"
187     echo "LFBFL_old_file_name: ${LFBFL_old_file_name}"
188     echo "LFBFL_old_file_directory: ${LFBFL_old_file_directory}"
189     echo "LFBFL_new_file_path: ${LFBFL_new_file_path}"
190     echo "LFBFL_new_file_name: ${LFBFL_new_file_name}"
191     echo "LFBFL_new_file_directory: ${LFBFL_new_file_directory}"
192 fi
193 # Extract "LFBFL_extension" with smart guessing for .tpl -----
194 # and set prefix for line with ©Copyright. -----
195 LFBFL_useful_file_name="${LFBFL_new_file_name}"
196 LFBFL_extension="${LFBFL_useful_file_name##*.}"
197 if [[ LFBFL_verbose -eq 1 ]]; then
198     echo "LFBFL_useful_file_name: ${LFBFL_useful_file_name}"
199     echo "LFBFL_extension: ${LFBFL_extension}"
200 fi
201 for ((LFBFL_j=0; LFBFL_j<3; ++LFBFL_j)); do
202     if [[ "${LFBFL_extension}" != "tpl" ]]; then

```

```

203         break
204     fi
205     LFBFL_useful_file_name="${LFBFL_useful_file_name:0:-4}"
206     LFBFL_extension="${LFBFL_useful_file_name##*.}"
207     if [[ LFBFL_verbose -eq 1 ]]; then
208         echo "LFBFL_useful_file_name: ${LFBFL_useful_file_name}"
209         echo "LFBFL_extension: ${LFBFL_extension}"
210     fi
211 done
212 LFBFL_comment_prefix=""
213 if [[ "${LFBFL_extension}" == "sh" ]]; then
214     LFBFL_comment_prefix="# "
215 fi
216 if [[ "${LFBFL_extension}" == "sql" ]]; then
217     LFBFL_comment_prefix="-- "
218 fi
219 if [[ "${LFBFL_extension}" == "tex" ]]; then
220     LFBFL_comment_prefix="% "
221 fi
222 if [[ "${LFBFL_useful_file_name}" == "${LFBFL_sfn}" ]]; then
223     LFBFL_comment_prefix="// "
224 fi
225 if [[ LFBFL_verbose -eq 1 ]]; then
226     echo "LFBFL_comment_prefix: ${LFBFL_comment_prefix}"
227 fi
228 #-----
229 # Split if necessary old_file_name, new_file_name, etc. to have
230 # comment lines below max_comment_line_length.
231 LFBFL_comment_prefix2="${LFBFL_comment_prefix}''"
232 # -----
233 LFBFL_old_file_name2="${LFBFL_comment_prefix2}"
234 LFBFL_old_file_name2+="${LFBFL_old_file_name}"
235 repeated_split_last_line "${LFBFL_old_file_name2}" \
236     "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
237     "'"' "${LFBFL_split_score_command1}" \
238     "${LFBFL_split_score_command_properties1}"
239 LFBFL_old_file_name2="${repeated_split_last_line_result}"
240 # -----
241 LFBFL_old_file_directory2="${LFBFL_comment_prefix2}"
242 LFBFL_old_file_directory2+="${LFBFL_old_file_directory}"
243 repeated_split_last_line "${LFBFL_old_file_directory2}" \
244     "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
245     "'"' "${LFBFL_split_score_command2}" \
246     "${LFBFL_split_score_command_properties2}"
247 LFBFL_old_file_directory2="${repeated_split_last_line_result}"
248 # -----

```

```

249 LFBFL_new_file_name2="${LFBFL_comment_prefix2}"
250 LFBFL_new_file_name2+="${LFBFL_new_file_name}"
251 repeated_split_last_line "${LFBFL_new_file_name2}" \
252     "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
253     ' "' "${LFBFL_split_score_command1}" \
254     "${LFBFL_split_score_command_properties1}"
255 LFBFL_new_file_name2="${repeated_split_last_line_result}"
256 # -----
257 LFBFL_new_file_directory2="${LFBFL_comment_prefix2}"
258 LFBFL_new_file_directory2+="${LFBFL_new_file_directory}"
259 repeated_split_last_line "${LFBFL_new_file_directory2}" \
260     "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
261     ' "' "${LFBFL_split_score_command2}" \
262     "${LFBFL_split_score_command_properties2}"
263 LFBFL_new_file_directory2="${repeated_split_last_line_result}"
264 # Create new comments strings. -----
265 LFBFL_new_comment=""
266 if [[ "${LFBFL_new_file_name2}" != "${LFBFL_old_file_name2}" ]];
267 then
268     LFBFL_new_comment="${LFBFL_comment_prefix}"
269     LFBFL_new_comment+=" ${LFBFL_timestamp}"
270     LFBFL_new_comment+=": This file was renamed from\n"
271     LFBFL_new_comment+=" ${LFBFL_old_file_name2}"'\n'
272     LFBFL_new_comment+=" ${LFBFL_comment_prefix}to\n"
273     LFBFL_new_comment+=" ${LFBFL_new_file_name2}"'.'"
274 fi
275 if [[ LFBFL_log_directory_change -eq 1 ]] && [[ \
276     "${LFBFL_new_file_directory}" \
277     != "${LFBFL_old_file_directory}" \
278 ]]; then
279     if [[ "${LFBFL_new_file_name2}" != "${LFBFL_old_file_name2}" ]];
280     then
281         LFBFL_new_comment+="\n"
282     fi
283     LFBFL_new_comment+=" ${LFBFL_comment_prefix}"
284     LFBFL_new_comment+=" ${LFBFL_timestamp}"
285     LFBFL_new_comment+=": This file was moved from\n"
286     LFBFL_new_comment+=" ${LFBFL_old_file_directory2}"'\n'
287     LFBFL_new_comment+=" ${LFBFL_comment_prefix}to\n"
288     LFBFL_new_comment+=" ${LFBFL_new_file_directory2}"'.'"
289 fi
290 if [[ LFBFL_verbose -eq 1 ]]; then
291     echo "LFBFL_new_comment: ${LFBFL_new_comment}"
292 fi
293 # Find line with ©Copyright. -----
294 LFBFL_copyright_line_number=$((
```

```

295     grep -n '@Copyright' "${LFBFL_new_file_path}" \
296     | head --lines=1 \
297     | cut -d ':' -f 1
298 )
299 if [[ LFBFL_verbose -eq 1 ]]; then
300     echo "LFBFL_copyright_line_number:" \
301         "${LFBFL_copyright_line_number}"
302 fi
303 if [[ LFBFL_copyright_line_number -eq 0 ]]; then
304     notify-send "${LFBFL_send_summary_2}" \
305         "${LFBFL_send_summary_2}" for: ${LFBFL_new_comment}
306     if [[ LFBFL_verbose -eq 1 ]]; then
307         echo "No copyright line. Skipping file."
308     fi
309     continue
310 fi
311 # Get total number of lines. -----
312 LFBFL_line_count=$(((
313     wc -l \
314     < "${LFBFL_new_file_path}"
315 ))
316 if [[ LFBFL_verbose -eq 1 ]]; then
317     echo "LFBFL_line_count: ${LFBFL_line_count}"
318 fi
319 # Compute number of lines after. -----
320 LFBFL_lines_after=$(((
321     LFBFL_line_count - LFBFL_copyright_line_number
322 ))
323 if [[ LFBFL_verbose -eq 1 ]]; then
324     echo "LFBFL_lines_after: ${LFBFL_lines_after}"
325 fi
326 # Update file. -----
327 LFBFL_temp_file_path="${LFBFL_new_file_path}.LFBFL.temp"
328 {
329     head --lines="${LFBFL_copyright_line_number}" \
330         "${LFBFL_new_file_path}"
331     echo -e "${LFBFL_new_comment}"
332     tail --lines="${LFBFL_lines_after}" "${LFBFL_new_file_path}"
333 } >> "${LFBFL_temp_file_path}"
334 mv "${LFBFL_temp_file_path}" "${LFBFL_new_file_path}"
335 # Add file to stage. -----
336 git add "${LFBFL_new_file_path}"
337 LFBFL_renaming_happened=1
338 done
339 readonly LFBFL_renaming_happened
340 if [[ LFBFL_renaming_happened -eq 1 ]]; then

```

```

341     # Commit -----
342     local LFBFL_message="commit_a_file_renamed_comment()"
343     LFBFL_message+=" post-commit hook, ${LFBFL_timestamp}."
344     readonly LFBFL_message
345     git commit --message="${LFBFL_message}"
346   fi
347 }

```

2.50 git_helpers/pre-commit

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2024 Laurent Frédéric Bernard François Lyaudet
24 # 2024-05-31T01:57+02:00: This file was renamed from
25 # "pre-commit"
26 # to
27 # "pre-commit".
28 # 2024-05-31T01:57+02:00: This file was moved from
29 # "."
30 # to
31 # "./git_helpers".
32
33 LFBFL_library="..../DevOrSysAdminScripts/git_helpers/"
34 LFBFL_library+="git_checks.libr.sh"
35 # shellcheck source=git_checks.libr.sh
36 source "${LFBFL_library}"

```

```

37
38 declare -ig LFBFL_git_hook_error=0
39 if ! check_files; then
40     LFBFL_git_hook_error=1
41 fi
42 if ! check_no_abusive_trailing_comma; then
43     LFBFL_git_hook_error=1
44 fi
45 if ! check_black_code_formatting; then
46     LFBFL_git_hook_error=1
47 fi
48 readonly LFBFL_git_hook_error
49 if [[ LFBFL_git_hook_error -eq 1 ]]; then
50     exit 1
51 fi
52 exit 0

```

2.51 IMAP_transfert.py

```

1 """
2 This file is part of DevOrSysAdminScripts library.
3
4 DevOrSysAdminScripts is free software:
5 you can redistribute it and/or modify it under the terms
6 of the GNU Lesser General Public License
7 as published by the Free Software Foundation,
8 either version 3 of the License,
9 or (at your option) any later version.
10
11 DevOrSysAdminScripts is distributed in the hope
12 that it will be useful,
13 but WITHOUT ANY WARRANTY;
14 without even the implied warranty of
15 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 See the GNU Lesser General Public License for more details.
17
18 You should have received a copy of
19 the GNU Lesser General Public License
20 along with DevOrSysAdminScripts.
21 If not, see <https://www.gnu.org/licenses/>.
22
23 ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24
25 Pour transférer d'un compte IMAP à un autre
26 Penser à autoriser les apps moins sûres sur Gmail si nécessaire
27 """

```

```

28
29     import argparse
30     import email
31     import imaplib
32
33
34     class IMAPTransfertBadArgumentError(Exception):
35         """
36             A class to avoid a linter warning,
37             that needs an Error suffix to avoid another linter warning,
38             that needs a docstring to avoid another linter warning.
39             If you don't see its intent, try badminton.
40         """
41
42
43     class SurpriseError(Exception):
44         """
45             A class to avoid a linter warning,
46             that needs an Error suffix to avoid another linter warning,
47             that needs a docstring to avoid another linter warning.
48             If you don't see its intent after seeing its use, try volleyball.
49             But "Cheer up!", now you don't need "pass" XD.
50         """
51
52
53     def IMAP_transfert(
54         from_login: str,
55         from_password: str,
56         from_host: str,
57         from_port: int,
58         to_login: str,
59         to_password: str,
60         to_host: str,
61         to_port: int,
62         *,
63         to_new_folder: str = "",
64         verbose: bool = False,
65     ) -> None:
66         """
67             Transfert all emails from some IMAP account"
68             to another IMAP account.
69         """
70         empty = ""
71         if from_login == "":
72             raise IMAPTransfertBadArgumentError("From login is empty.")
73         # if from_password == "":

```

```

74     if from_password == empty:
75         # https://github.com/PyCQA/bandit/issues/714
76         # Dumb
77         raise IMAPTransfertBadArgumentError("From password is empty.")
78     if from_host == "":
79         raise IMAPTransfertBadArgumentError("From host is empty.")
80     if from_port <= 0:
81         raise IMAPTransfertBadArgumentError(
82             "From port is not positive.",
83         )
84     if to_login == "":
85         raise IMAPTransfertBadArgumentError("To login is empty.")
86     if to_password == empty:
87         # And Dumber The infinite collection XD
88         raise IMAPTransfertBadArgumentError("To password is empty.")
89     if to_host == "":
90         raise IMAPTransfertBadArgumentError("To host is empty.")
91     if to_port <= 0:
92         raise IMAPTransfertBadArgumentError(
93             "To port is not positive.",
94         )
95     # pylint: disable=pointless-string-statement
96 """
97 Maybe one day Python will ship something like that in its standard
98 library:
99 https://www.php.net/manual/fr/function.filter-var.php
100 But now urlparse doesn't do any validation.
101 You may want to check that the hosts or other fields are not
102 "anything".
103 """
104
105 # pylint: disable=unreachable
106 raise SurpriseError(
107     "I haven't tested this version of my script... Ooooooh!\n"
108     "But I linted it without this exception ;). Aaaaaaaaah!\n"
109     "XD XD XD\n"
110     "Untested Scream Horror Show (Soon on BrainfuckedFlix)",
111 )
112 """
113 # I haven't tested any version with an empty to_new_folder...
114
115 if verbose:
116     print("Connecting to from account.")
117 from_account = imaplib.IMAP4_SSL(host=from_host, port=from_port)
118 from_account.login(from_login, from_password)
119

```

```

120     if verbose:
121         print('Connecting to "to account".')
122     to_account = imaplib.IMAP4_SSL(host=to_host, port=to_port)
123     to_account.login(to_login, to_password)
124
125     if to_new_folder != "":
126         if verbose:
127             print("Creating new folder in to account.")
128         to_account.create(to_new_folder)
129
130     if verbose:
131         print("Fetching all emails ids in from account.")
132     from_account.select()
133     result, ids = from_account.uid("search", "all")
134     if verbose:
135         print(f'from_account.uid("search") result:{result} ids:{ids}')
136     ids = ids[0].decode().split()
137     if verbose:
138         print("Fetching and transferring each email in from account.")
139     for my_id in ids:
140         result, raw_message = from_account.uid(
141             "fetch",
142             my_id,
143             "(RFC822)",
144         )
145         if verbose:
146             print(
147                 f'from_account.uid("fetch", {my_id})'
148                 f" result:{result}",
149             )
150         message = raw_message[0][1]
151         my_email = email.message_from_string(message.decode("utf-8"))
152         flags = imaplib.ParseFlags(raw_message[0][0])
153         # flags = " ".join(flags) mypy is unhappy but IMAP flags are
154         # ASCII strings, we give mypy what it wants.
155         flags_as_strs = [flag.decode("ascii") for flag in flags]
156         flags_str = " ".join(flags_as_strs)
157         # date = imaplib.Internaldate2tuple(raw_message[0][0])
158         date = '' + my_email["date"] + ''
159         if verbose:
160             print(
161                 f'from_account.uid("fetch", {my_id})'
162                 f" flags:{flags} date:{date}",
163             )
164         to_account.append(to_new_folder, flags_str, date, message)
165

```

```

166
167 def main() -> None:
168     """
169     Parse command-line arguments and
170     transfert all emails from some IMAP account
171     to another IMAP account.
172     """
173     parser = argparse.ArgumentParser(
174         description=(
175             "Transfert all emails from some IMAP account"
176             " to another IMAP account"
177         ),
178     )
179     parser.add_argument(
180         "from_login",
181         type=str,
182         help="Email account login of user one.",
183     )
184     parser.add_argument(
185         "from_password",
186         type=str,
187         help="Email account password of user one.",
188     )
189     parser.add_argument(
190         "to_login",
191         type=str,
192         help="Email account login of user two.",
193     )
194     parser.add_argument(
195         "to_password",
196         type=str,
197         help="Email account password of user two.",
198     )
199     parser.add_argument(
200         "--to_new_folder",
201         type=str,
202         default="",
203         help="Transfert emails into a new folder of this name.",
204     )
205     parser.add_argument(
206         "--from_host",
207         type=str,
208         default="imap.gmail.com",
209         help="Host of first email account",
210     )
211     parser.add_argument(

```

```

212     "--from_port",
213     type=int,
214     default=993,
215     help="Port on host of first email account",
216 )
217 parser.add_argument(
218     "--to_host",
219     type=str,
220     default="imap.gmail.com",
221     help="Host of second email account",
222 )
223 parser.add_argument(
224     "--to_port",
225     type=int,
226     default=993,
227     help="Port on host of second email account",
228 )
229 parser.add_argument(
230     "-v",
231     "--verbose",
232     action="store_true",
233     help="Print details of operations",
234 )
235
236 args = parser.parse_args()
237
238 if args.verbose:
239     print(
240         "-----\n"
241         "IMAP transfert:\n"
242         "-----\n"
243         "From account:\n"
244         f" - Login: {args.from_login}\n"
245         f" - Password: {args.from_password}\n"
246         f" - Host: {args.from_host}\n"
247         f" - Port: {args.from_port}\n"
248         "To account:\n"
249         f" - Login: {args.to_login}\n"
250         f" - Password: {args.to_password}\n"
251         f" - Host: {args.to_host}\n"
252         f" - Port: {args.to_port}\n"
253         "-----\n",
254     )
255     if args.to_new_folder is not None:
256         print(
257             f"Emails will go in folder {args.to_new_folder}"

```

```

258             " after it is created.",
259         )
260     try:
261         IMAP_transfert(
262             args.from_login,
263             args.from_password,
264             args.from_host,
265             args.from_port,
266             args.to_login,
267             args.to_password,
268             args.to_host,
269             args.to_port,
270             to_new_folder=args.to_new_folder,
271             verbose=args.verbose,
272         )
273     except (IMAPTransfertBadArgumentError, SurpriseError) as e:
274         print(e)
275
276
277 if __name__ == "__main__":
278     main()

```

2.52 LiensUtiles.txt

```

1 https://support.google.com/a/answer/9476255?hl=fr#zippy=%2C%C3%A9tape\
2 -configurez-le-service-de-migration-de-donn%C3%A9es
3 -----
4 https://stackoverflow.com/a/78413334/5796086
5 https://stackoverflow.com/questions/21612980/\
6 why-is/usr-bin-env-bash-superior-to-bin-bash/78413334
7 https://web.archive.org/web/20240501105801/https://stackoverflow.com\
8 /questions/21612980/why-is/usr-bin-env-bash-superior-to-bin-bash/\
9 78413334#78413334
10 -----
11 https://news.ycombinator.com/item?id=22214690
12 https://github.com/koalaman/shellcheck/issues/2005
13 https://opensource.stackexchange.com/questions/6271/\
14 calling-gpld-programs-from-a-shell-script
15 -----

```

2.53 README.md.tpl

```

1 # DevOrSysAdminScripts
2
3 [ ! [CodeFactor-badge] ] [CodeFactor-package-page]
4 [ ! [Codacy-badge] ] [Codacy-package-page]

```

```

5 ! [GitHub-top-language-badge]
6 ! [GitHub-code-size-in-bytes-badge]
7 [ ! [Security:bandit:badge]] [Bandit-GitHub.com]
8 [ ! [Code-style:black:badge]] [Black-GitHub.com]
9 [ ! [Imports:isort:badge]] [Isort-GitHub.io]
10 [ ! [Typecheck:mypy:badge]] [Typecheck-mypy-lang.org]
11 [ ! [Linting:pylint:badge]] [Pylint-GitHub.com]
12 [ ! [Linting:ruff:badge]] [Ruff-GitHub.com]
13
14 |           **Some "small" useful scripts**          |
15 |-----:|
16 | **for enhancing development or system-administrators tasks** |
17
18 Currently, the biggest part of this repository is in
19 a suite of shell scripts about build and checks that
20 I use in some of my other repositories.
21
22 I tried to follow some conventions for my bash code:
23
24 - I tried to encapsulate most of the code in functions.
25 - Unless used as return values, variables in functions are defined
26   as local variables ('local' keyword or 'declare')
27   with prefix "LFBFL_".
28 - When used as return values, variables in functions are defined
29   as global variables ('declare -g') with prefix
30   "@function_name@_result_".
31 - Whenever a variable is no more modified after some point,
32   add the keyword 'readonly' or use 'declare -r'.
33 - Whenever a variable will only contain integer values (or boolean
34   values as 0 or 1), use 'declare -i'.
35 - I named bash files with suffixes ".exec.sh" when the script can
36   be runned
37   (executable, abbreviation stops before a vowel).
38 - I named bash files with suffixes ".libr.sh" when the script contains
39   only functions definitions
40   (library, abbreviation stops before a vowel).
41 - When some ".exec.sh" code is encapsulated into functions,
42   either these functions can be reused and go in some ".libr.sh" file,
43   either they are truly specific to this script and they are kept
44   in ".exec.sh" file.
45
46 But the files "pre-commit" and "post-commit" were not renamed,
47 since it is not possible to give them other names to use them as
48 pre-commit/post-commit hooks in 'git'.
49
50 [CodeFactor-badge]: https://www.codefactor.io/repository/github/\

```

```

51 llyaudet/DevOrSysAdminScripts/badge
52
53 [CodeFactor-package-page]: https://www.codefactor.io/repository/>\n
54 github/llyaudet/DevOrSysAdminScripts
55
56 [Codacy-badge]: https://app.codacy.com/project/badge/Grade/\n
57 3dd6ef85e8db4460991d1ebc404b68dd
58
59 [Codacy-package-page]: https://app.codacy.com/gh/LLyaudet/>\n
60 DevOrSysAdminScripts/dashboard?utm_source=gh&utm_medium=referral\
61 &utm_content=&utm_campaign=Badge_grade
62
63 [GitHub-top-language-badge]: https://img.shields.io/github/>\n
64 languages/top/llyaudet/DevOrSysAdminScripts
65
66 [GitHub-code-size-in-bytes-badge]: https://img.shields.io/github/>\n
67 languages/code-size/llyaudet/DevOrSysAdminScripts
68
69 [Security:bandit:badge]: https://img.shields.io/badge/>\n
70 security-bandit-yellow.svg
71
72 [Bandit-GitHub.com]: https://github.com/PyCQA/bandit
73
74 [Code-style:black:badge]: https://img.shields.io/badge/>\n
75 code%20style-black-000000.svg
76
77 [Black-GitHub.com]: https://github.com/psf/black
78
79 [Imports:isort:badge]: https://img.shields.io/badge/>\n
80 %20imports-isort-%231674b1?style=flat&labelColor=ef8336
81
82 [Isort-GitHub.io]: https://pycqa.github.io/isort/
83
84 [Typecheck:mypy:badge]: https://www.mypy-lang.org/static/>\n
85 mypy_badge.svg
86
87 [Typecheck-mypy-lang.org]: https://mypy-lang.org/
88
89 [Linting:pylint:badge]: https://img.shields.io/badge/>\n
90 linting-pylint-yellowgreen
91
92 [Pylint-GitHub.com]: https://github.com/pylint-dev/pylint
93
94 [Linting:ruff:badge]: https://img.shields.io/endpoint?url=\n
95 https://raw.githubusercontent.com/astral-sh/ruff/main/assets/>\n
96 badge/v2.json

```

```
97
98 [Ruff-GitHub.com]: https://github.com/astral-sh/ruff
```

2.54 unicode.libr.php

```
1  <?php
2  /*
3  This file is part of DevOrSysAdminScripts library.
4
5  DevOrSysAdminScripts is free software:
6  you can redistribute it and/or modify it under the terms
7  of the GNU Lesser General Public License
8  as published by the Free Software Foundation,
9  either version 3 of the License,
10 or (at your option) any later version.
11
12 DevOrSysAdminScripts is distributed in the hope
13 that it will be useful,
14 but WITHOUT ANY WARRANTY;
15 without even the implied warranty of
16 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
17 See the GNU Lesser General Public License for more details.
18
19 You should have received a copy of
20 the GNU Lesser General Public License
21 along with DevOrSysAdminScripts.
22 If not, see <https://www.gnu.org/licenses/>.
23
24 ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
25 This file was renamed from "unicode.php" to "unicode.libr.php".
26 */
27
28 function decimal_code_point_to_UTF8(
29     $i_code_point_in_decimal_notation
30 ) {
31     // var_dump($i_code_point_in_decimal_notation);
32     // 0xxxxxxxx ASCII
33     if($i_code_point_in_decimal_notation < 128){
34         return chr($i_code_point_in_decimal_notation);
35     }
36     $i_continuation_base_value = 128;
37     // 110xxxxx 10xxxxxx
38     if($i_code_point_in_decimal_notation < 256 * (2**3)){
39         $i_first_byte_base_value = 192;
40         $i_last_byte_significant_bits = (
41             $i_code_point_in_decimal_notation % (2**6)
```

```

42 );
43 $i_first_byte_significant_bits = intdiv(
44 (
45     $i_code_point_in_decimal_notation
46     - $i_last_byte_significant_bits
47 ),
48 2**6,
49 );
50 $s_result = chr(
51     $i_first_byte_base_value
52     + $i_first_byte_significant_bits
53 );
54 $s_result .= chr(
55     $i_continuation_base_value
56     + $i_last_byte_significant_bits
57 );
58 return $s_result;
59 }
60 // 1110xxxx 10xxxxxx 10xxxxxx
61 if($i_code_point_in_decimal_notation < 256 * 256) {
62     $i_first_byte_base_value = 224;
63     $arr_arr_data_per_byte_reverse = [
64         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
65         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
66         ["i_bits" => 4, "i_base_value" => $i_first_byte_base_value],
67     ];
68     $i_rest = $i_code_point_in_decimal_notation;
69     $s_result = "";
70     foreach($arr_arr_data_per_byte_reverse as $arr_data) {
71         // _Pragma unroll
72         // J'aurais pu "roller" les 4 cas des ifs.
73         // Mais pour les perfs, il faudrait tout unroller vu le nombre
74         // d'itérations. Et j'aimerais bien comparer du code unrollé
75         // aux 2 niveaux à la main avec celui unrollé par le compilateur
76         // sur les 2 niveaux. Je préjuge/présume qu'il unroll
77         // l'intérieur d'abord (pour ne pas refaire le même boulot)
78         // puis l'extérieur et qu'il oublie de réoptimiser l'intérieur
79         // en fonction de l'extérieur unrollé. Est-ce que j'ai raison ?
80         $i_significant_bits = $i_rest % (2**$arr_data["i_bits"]);
81         $i_rest = intdiv(
82             $i_rest - $i_significant_bits,
83             2**$arr_data["i_bits"],
84         );
85         $s_result = ( // .= or =
86             chr($i_significant_bits + $arr_data["i_base_value"])
87             . $s_result

```

```

88     );
89 }
90 return $s_result;
91 }
92 // 11110xxx 10xxxxxx 10xxxxxx 10xxxxxx
93 if($i_code_point_in_decimal_notation < 256 * 256 * (2**5)) {
94     $i_first_byte_base_value = 240;
95     $arr_arr_data_per_byte_reverse = [
96         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
97         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
98         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
99         ["i_bits" => 3, "i_base_value" => $i_first_byte_base_value],
100    ];
101    $i_rest = $i_code_point_in_decimal_notation;
102    $s_result = "";
103    foreach($arr_arr_data_per_byte_reverse as $arr_data) {
104        // _Pragma unroll
105        $i_significant_bits = $i_rest % (2**$arr_data["i_bits"]);
106        $i_rest = intdiv(
107            $i_rest - $i_significant_bits,
108            2**$arr_data["i_bits"],
109        );
110        $s_result = ( // .= or =
111            chr($i_significant_bits + $arr_data["i_base_value"])
112            . $s_result
113        );
114    }
115    return $s_result;
116 }
117 throw new Exception(
118     "UTF-8 avec jusqu'à 6 octets a été abandonné il y a longtemps."
119     ." Autre anecdote \"débile\" : MySQL a de base UTF-8 avec jusqu'à"
120     ." 3 octets au lieu de 4 :face-palm:."
121     ." Du coup, ça peut donner quelques \"surprises\"."
122     ." Sinon, y'a aussi ASCII dans MySQL."
123     ." Et comme ça joue sur la mémoire et les performances,"
124     ." pourquoi ils n'ont pas fait ASCII, UTF-8-2, UTF-8-3, UTF-8-4 ?"
125     ." :)"
126 );
127 }
128
129
130
131 function hexa_code_point_to_UTF8(
132     $s_code_point_in_hexadecimal_notation
133 ) {

```

```

134     $s_code_point_in_hexadecimal_notation = strtoupper(
135         $s_code_point_in_hexadecimal_notation
136     );
137     $i_code_point_in_decimal_notation = 0;
138     $i_ord_0 = ord('0');
139     $i_ord_A = ord('A');
140     $i_digit_offset = -$i_ord_0;
141     $i_letter_offset = 10 - $i_ord_A;
142     for(
143         $i = 0, $i_max = strlen($s_code_point_in_hexadecimal_notation);
144         $i < $i_max;
145         ++$i
146     ) {
147         $s_hexa_digit = $s_code_point_in_hexadecimal_notation[$i];
148         $i_code_point_in_decimal_notation *= 16;
149         if(ctype_digit($s_hexa_digit)){
150             $i_code_point_in_decimal_notation += (
151                 ord($s_hexa_digit) + $i_digit_offset
152             );
153             continue;
154         }
155         if(ctype_xdigit($s_hexa_digit)){
156             $i_code_point_in_decimal_notation += (
157                 ord($s_hexa_digit) + $i_letter_offset
158             );
159             continue;
160         }
161         throw new Exception("Not an hexadecimal digit.");
162     }
163     return decimal_code_point_to_UTF8(
164         $i_code_point_in_decimal_notation
165     );
166 }
167
168
169
170 function check_string_is_valid_ASCII($s_string){
171     for($i = 0, $i_max = strlen($s_string); $i < $i_max; ++$i){
172         if(ord($s_string[$i]) > 127){
173             throw new Exception(
174                 "Non-ASCII character found at octet position "
175                 . $i
176                 . ", value "
177                 . ord($s_string[$i])
178                 . "."
179             );

```

```

180         }
181     }
182     return true;
183 }
184
185
186
187 function check_file_is_valid_ASCII($s_file_name) {
188     $s_string = file_get_contents($s_file_name);
189     if($s_string === false) {
190         throw new Exception("File ".$s_file_name." not found.");
191     }
192     return check_string_is_valid_ASCII($s_string);
193 }
194
195
196
197 function check_string_is_valid_UTF8($s_string) {
198     $i_continuation_octet_needed = 0;
199     $i_character_start_position = 0;
200     for($i = 0, $i_max = strlen($s_string); $i < $i_max; ++$i){
201         $i_current_octet = ord($s_string[$i]);
202         if($i_continuation_octet_needed > 0){
203             if($i_current_octet < 128 || $i_current_octet >= 192) {
204                 throw new Exception(
205                     "Non-UTF8 character found at start position "
206                     . $i_character_start_position
207                     . ", octet at position "
208                     . $i
209                     . " has value "
210                     . $i_current_octet
211                     . " which is not a continuation octet."
212                 );
213             }
214             --$i_continuation_octet_needed;
215             continue;
216         }
217         $i_character_start_position = $i;
218         if($i_current_octet < 128){ // 0xxxxxxx ASCII
219             continue;
220         }
221         if($i_current_octet >= 128 && $i_current_octet < 192) {
222             throw new Exception(
223                 "Non-UTF8 character found at start position "
224                 . $i_character_start_position

```

```

226         .", octet at position "
227         .$i
228         ." has value "
229         .$i_current_octet
230         ." which is a continuation octet."
231     );
232 }
233 if($i_current_octet >= 192 && $i_current_octet < 224) {
234     // 110xxxxx 10xxxxxx
235     $i_continuation_octet_needed = 1;
236     continue;
237 }
238 if($i_current_octet >= 224 && $i_current_octet < 240) {
239     // 1110xxxx 10xxxxxx 10xxxxxx
240     $i_continuation_octet_needed = 2;
241     continue;
242 }
243 if($i_current_octet >= 240 && $i_current_octet < 248) {
244     // 11110xxx 10xxxxxx 10xxxxxx 10xxxxxx
245     $i_continuation_octet_needed = 3;
246     continue;
247 }
248 throw new Exception(
249     "Non-UTF8 character found at start position "
250     .$i_character_start_position
251     .", octet at position "
252     .$i
253     ." has value "
254     .$i_current_octet
255     ." which is invalid."
256 );
257 }
258 return true;
259 }
260
261
262 function check_file_is_valid_UTF8($s_file_name) {
263     $s_string = file_get_contents($s_file_name);
264     if($s_string === false) {
265         throw new Exception("File ".$s_file_name." not found.");
266     }
267     return check_string_is_valid_UTF8($s_string);
268 }
269
270
271

```

```

272
273 /* 
274 ?>
275 <?php
276 require_once("./unicode.libr.php");
277 var_dump(hexa_code_point_to_UTF8("002B"));
278 var_dump(hexa_code_point_to_UTF8("00E6"));
279 var_dump(hexa_code_point_to_UTF8("1400"));
280 var_dump(hexa_code_point_to_UTF8("10111"));
281 // https://en.wikibooks.org/wiki/Unicode/Character_reference/2000-2FFF
282 // Some invisible unicode characters that are a problem:
283 // 2000-200F 2028-202F 205F 2060-206F
284 $arr = [
285     '2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007',
286     '2008', '2009', '200A', '200B', '200C', '200D', '200E', '200F',
287     '2028', '2029', '202A', '202B', '202C', '202D', '202E', '202F',
288     '205F',
289     '2060', '2061', '2062', '2063', '2064', '2065', '2066', '2067',
290     '2068', '2069', '206A', '206B', '206C', '206D', '206E', '206F',
291 ];
292 $arr_s = [];
293 $arr_s2 = [];
294 $arr_s3 = [];
295 foreach($arr as $s_hexa) {
296     $s = hexa_code_point_to_UTF8($s_hexa);
297     $arr_s []= $s;
298     if(strlen($s) === 2) {
299         $s2 = '\x'.bin2hex($s[0]).'\x'.bin2hex($s[1]);
300     }
301     if(strlen($s) === 3) {
302         $s2 = '\x'.bin2hex($s[0]).'\x'.bin2hex($s[1])
303             .' \x'.bin2hex($s[2]);
304     }
305     $arr_s2 []= $s2;
306     $s3 = " | sed -e 's/" . $s2 . "/g' \\";
307     $arr_s3 []= $s3;
308     echo($s3."\n");
309 }
310 //*/
311

```

2.55 upgrade_build_and_checks.exec.sh

```

1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #

```

```

4  # DevOrSysAdminScripts is free software:
5  # you can redistribute it and/or modify it under the terms
6  # of the GNU General Public License
7  # as published by the Free Software Foundation,
8  # either version 3 of the License,
9  # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023-2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "upgrade_build_and_checks.sh"
25 # to "upgrade_build_and_checks.exec.sh".
26
27 upgrade_build_and_checks() {
28     local LFBFL_verbose=""
29     if [[ "$*" == *--verbose* ]]; then
30         echo "$0 $*"
31         LFBFL_verbose="--verbose"
32     fi
33     local LFBFL_fixed_point_build=""
34     if [[ "$*" == *--fixed_point_build* ]]; then
35         LFBFL_fixed_point_build="--fixed_point_build"
36     fi
37     readonly LFBFL_verbose
38     (
39         cd build_and_checks_dependencies/
40         && ./update_common_build_and_checks.exec.sh "${LFBFL_verbose}" \
41     )
42     ./build_and_checks.exec.sh " ." "${LFBFL_verbose}" \
43     "${LFBFL_fixed_point_build}"
44 }
45
46 upgrade_build_and_checks "$@"

```

2.56 wget_sha512.libr.sh

```
1 #!/usr/bin/env bash
2 # This file is part of DevOrSysAdminScripts library.
3 #
4 # DevOrSysAdminScripts is free software:
5 # you can redistribute it and/or modify it under the terms
6 # of the GNU General Public License
7 # as published by the Free Software Foundation,
8 # either version 3 of the License,
9 # or (at your option) any later version.
10 #
11 # DevOrSysAdminScripts is distributed in the hope
12 # that it will be useful,
13 # but WITHOUT ANY WARRANTY;
14 # without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
16 # See the GNU General Public License for more details.
17 #
18 # You should have received a copy of
19 # the GNU General Public License
20 # along with DevOrSysAdminScripts.
21 # If not, see <https://www.gnu.org/licenses/>.
22 #
23 # ©Copyright 2023–2026 Laurent Frédéric Bernard François Lyaudet
24 # This file was renamed from "wget_sha512.sh"
25 # to "wget_sha512.libr.sh".
26
27 wget_sha512() {
28     # $1 file_path
29     # $2 download_URL
30     # $3 correct_sha512
31     # $4 optional --verbose
32     local LFBFL_verbose=""
33     if [[ "$*" == *--verbose* ]]; then
34         echo "$0 wget_sha512 $*"
35         LFBFL_verbose="--verbose"
36     fi
37     readonly LFBFL_verbose
38     if [[ ! -f "$1" ]];
39     then
40         wget "${LFBFL_verbose}" -O "$1" "$2"
41     fi
42     declare -r LFBFL_present_sha512=$((
43         sha512sum "$1" \
44         | cut -f1 -d' '
```

```
45      )
46  if [ [ "${LFBFL_present_sha512}" != "$3" ] ];
47  then
48      echo "$1 does not have correct sha512"
49      echo "wanted $3"
50      echo "found ${LFBFL_present_sha512}"
51      exit
52  fi
53 }
```

Merci Dieu ! Merci Père ! Merci Seigneur ! Merci Saint Esprit !