

DevOrSysAdminScripts

Laurent Frédéric Bernard François Lyaudet

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

Laurent.Lyaudet@gmail.com

February 24, 2026

Abstract

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

Current version: 2026-02-24

Current number of commits: 374

Current git SHA1: 4e569bf6be2791ce1ce6f703c1b66e07f68a8026

Code lines: 7421 total lines, 6729 not empty lines, 692 empty lines.

Contents

1	Files tree	3
2	Listing of files	6
2.1	build_and_checks_dependencies/build_and_checks.exec.sh.tpl . . .	6
2.2	build_and_checks_dependencies/build_md_from_printable_md.exec.sh	9
2.3	build_and_checks_dependencies/check_shell_scripts_beginnings.libr.sh	11
2.4	build_and_checks_dependencies/check_shell_scripts_indentation.libr.sh	13
2.5	build_and_checks_dependencies/check_URLs.libr.sh	15
2.6	build_and_checks_dependencies/common_build_and_checks.exec.sh	17
2.7	build_and_checks_dependencies/common_build_and_checks.exec.sh.tpl	32
2.8	build_and_checks_dependencies/comparisons.libr.sh	46
2.9	build_and_checks_dependencies/create_PDF.exec.sh	50
2.10	build_and_checks_dependencies/generate_from_template.libr.sh	60
2.11	build_and_checks_dependencies/get_common_text_glob_patterns.libr.sh	62
2.12	build_and_checks_dependencies/grammar_and_spell_check.libr.sh	64
2.13	build_and_checks_dependencies/licenses_templates/build_licenses_templates.exec.sh	66

2.14	build_and_checks_dependencies/licenses_templates/license_file_header_AGPLv3+.tpl	74
2.15	build_and_checks_dependencies/licenses_templates/license_file_header_GPLv3+.tpl	74
2.16	build_and_checks_dependencies/licenses_templates/license_file_header_LGPLv3+.tpl	75
2.17	build_and_checks_dependencies/lines_counts.libr.sh	75
2.18	build_and_checks_dependencies/lines_filters.libr.sh	78
2.19	build_and_checks_dependencies/lines_maps.libr.sh	84
2.20	build_and_checks_dependencies/overwrite_if_not_equal.libr.sh	86
2.21	build_and_checks_dependencies/python_black_complement.libr.sh	88
2.22	build_and_checks_dependencies/python_isort_complement.libr.sh	89
2.23	build_and_checks_dependencies/split_score.exec.php	90
2.24	build_and_checks_dependencies/split_score.libr.php	91
2.25	build_and_checks_dependencies/strings_functions.libr.sh	93
2.26	build_and_checks_dependencies/too_long_code_lines.libr.sh	100
2.27	build_and_checks_dependencies/update_common_build_and_checks.exec.sh	102
2.28	build_and_checks_dependencies/update_or_check_files_names_listing.exec.sh	105
2.29	build_and_checks_variables/.gitignore	108
2.30	build_and_checks_variables/bandit.ini	108
2.31	build_and_checks_variables/bandit_baseline.json	108
2.32	build_and_checks_variables/DevOrSysAdminScripts.html.tpl	109
2.33	build_and_checks_variables/DevOrSysAdminScripts.tex.tpl	114
2.34	build_and_checks_variables/files_listing.tex.tpl	117
2.35	build_and_checks_variables/files_names_listing.txt	131
2.36	build_and_checks_variables/licenses_notes.txt	134
2.37	build_and_checks_variables/phpmd_baseline.xml	134
2.38	build_and_checks_variables/pylintrc.toml	135
2.39	build_and_checks_variables/repository_data.txt	135
2.40	build_and_checks_variables/ruff.toml	138
2.41	build_and_checks_variables/shellcheck.ini	138
2.42	build_and_checks_variables/shellcheck_notes.txt	138
2.43	build_and_checks.exec.sh	139
2.44	COPYING	141
2.45	COPYING.LESSER	156
2.46	git_helpers/git_checks.libr.sh	160
2.47	git_helpers/post-commit	162
2.48	git_helpers/post_commit_renaming_comment.libr.sh	163
2.49	git_helpers/pre-commit	171
2.50	LiensUtiles.txt	172
2.51	README.md.tpl	172
2.52	unicode.libr.php	174
2.53	upgrade_build_and_checks.exec.sh	181
2.54	wget_sha512.libr.sh	182

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
        ├── 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
    └── LiensUtiles.txt
    └── README.html
    └── README.md
    └── README.md.tpl
    └── .ruff_cache
    └── unicode.libr.php
    └── upgrade_build_and_checks.exec.sh
    └── wget_sha512.libr.sh

```

6 directories, 59 files

```

[ 12K Feb 19 16:09]  ./
└── [4.0K Feb 19 16:09]  build_and_checks_dependencies/
    ├── [3.7K Feb 19 15:54]  build_and_checks.exec.sh.tpl/*
    ├── [3.0K Feb 19 15:33]  build_md_from_printable_md.exec.sh/*
    ├── [2.1K Feb 15 11:34]  check_shell_scripts_beginnings.libr.sh
    ├── [2.8K Feb 19 14:54]  check_URLs.libr.sh
    ├── [ 23K Feb 19 15:57]  common_build_and_checks.exec.sh/*
    ├── [ 19K Feb 18 16:15]  common_build_and_checks.exec.sh.tpl
    ├── [4.6K Feb 19 15:09]  comparisons.libr.sh
    ├── [ 14K Feb 19 15:45]  create_PDF.exec.sh/*
    ├── [3.2K Feb 19 15:55]  generate_from_template.libr.sh
    ├── [2.0K Feb 19 15:19]  get_common_text_glob_patterns.libr.sh
    ├── [1.9K Feb 19 14:43]  grammar_and_spell_check.libr.sh
    └── [4.0K Feb 19 15:41]  licenses_templates/
        ├── [ 13K Feb 19 15:41]  build_licenses_templates.exec.sh/*
        ├── [ 728 Dec 30 19:32]  license_file_header GPLv3+.tpl
        ├── [ 707 Dec 27 21:48]  license_file_header GPLv3+.tpl
        └── [ 728 Dec 27 21:48]  license_file_header LGPLv3+.tpl
    └── [2.7K Feb 17 18:48]  lines_counts.libr.sh
    └── [6.8K Feb 19 15:49]  lines_filters.libr.sh
    └── [2.7K Feb 19 14:47]  lines_maps.libr.sh
    └── [1.8K Feb 19 15:01]  overwrite_if_not_equal.libr.sh

```

```

    [1.4K Jan 24 18:32] python_black_complement.libr.sh
    [1.7K Feb 19 15:37] 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 19 14:48] strings_functions.libr.sh
    [2.5K Feb 19 14:53] too_long_code_lines.libr.sh
    [4.1K Feb 19 16:09] update_common_build_and_checks.exec.sh*
    [3.8K Feb 18 16:22] *
    | update_or_check_files_names_listing.exec.sh**
[3.8K Feb 19 15:57] build_and_checks.exec.sh*
[4.0K Feb 19 16:09] build_and_checks_variables/
    [ 15 Jan  5 20:00] bandit_baseline.json
    [ 25 Jan  5 18:49] bandit.ini
    [5.2K Feb 14 15:05] DevOrSysAdminScripts.html.tpl
    [ 22K Feb 19 16:02] DevOrSysAdminScripts.tex
    [3.8K Feb 14 15:04] DevOrSysAdminScripts.tex.tpl
    [ 12K Feb 18 16:27] files_listing.tex.tpl
    [3.8K Feb 16 18:43] files_names_listing.txt
    [ 24 Jan 15 22:26] .gitignore
    [1.2K Feb 14 15:04] licenses_notes.txt
    [ 926 Feb 13 20:48] phpmdb_baseline.xml
    [ 173 Jan 30 19:40] pylintrc.toml
    [3.5K Feb 16 15:47] repository_data.txt
    [ 159 Feb 14 15:02] ruff.toml
    [ 2.0K Feb 18 15:52] shellcheck_notes.txt
    [ 34K May  6 2024] COPYING
    [7.5K Dec 27 21:48] COPYING.LESSER
    [274K Feb 19 16:02] DevOrSysAdminScripts.html
    [270K Feb 19 16:02] DevOrSysAdminScripts.pdf
    [4.0K Feb 19 16:02] git_helpers/
        [3.4K Feb 19 15:04] git_checks.libr.sh
        [1.2K Feb 14 17:57] post-commit
        [ 13K Feb 19 16:02] post_commit_renaming_comment.libr.sh
        [1.4K Feb  8 16:41] pre-commit
    [ 830 Dec 29 19:10] LiensUtiles.txt
    [3.6K Feb 16 17:08] README.html
    [3.5K Feb 16 17:08] README.md
    [3.6K Feb 16 16:59] README.md.tpl
    [4.0K Feb 14 13:01] .ruff_cache/
    [9.1K Feb 12 19:57] 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

```

6 directories, 57 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 GPLv3+.

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/>.

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, utils 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="${LFBFL_variables_directory}/"
88     LFBFL_data_file_name+="repository_data.txt"
89     grep_variable "${LFBFL_data_file_name}" repository_name
90     # Touching the 3 following files first let us iterate
91     # upgrade_build_and_checks 2 times instead of 3 to reach
92     # a fixed point.
93     touch "${LFBFL_variables_directory}/${repository_name}.tex"
94     touch "${LFBFL_working_directory}/${repository_name}.pdf"
95     touch "${LFBFL_working_directory}/${repository_name}.html"
96     # The two iterations of build_and_checks needs to complete
97     # during the same minute to have a fixed point.
98     build_and_checks "$@"

```

```

99     # build_and_checks "$@" second iteration below
100    # Then calling this script without --fixed_point_build should not
101    # yield any new superficial modifications.
102 fi
103
104 build_and_checks "$@"

```

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 "${LFBFL_base_name}.html.temp"
92 # Correcting pandoc HTML "by hand".
93 # https://github.com/jgm/pandoc/issues/10957
94 # With the most open-minded answer found on Internet ;) XD.
95 # This second sed expression is what I need,
96 # but it may be unsuitable for you.
97 # Other problems:
98 # https://github.com/jgm/pandoc/issues/11484
99 local LFBFL_prewrap="s~(code \{(\n|[^}])*\})(      \})"
100 LFBFL_prewrap+="~\1      white-space: pre-wrap;\n\n\3~M"
101 readonly LFBFL_prewrap
102 sed -i -E -e 's~( <meta .*) />~\1>~g' \
103         -e 's~<html .*>~<html lang="en">~' \
104         -e '/    code\{white-space: pre-wrap;\}/d' \
105     "${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 ${LFBFL_base_name}.html"
121 }
122 build_md_from_printable_md "$@"

```

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_SHELL_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="workeduntilstosomeasshole"
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 [[ "${LFBFL_previous_line}" == "workeduntilstosomeasshole" ]]; then
62         LFBFL_previous_line="${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=${#LFBFL_some_line}
71     LFBFL_line_end_length=${#LFBFL_line_end}
72     LFBFL_line_start_length=$(((
73         LFBFL_line_length-LFBFL_line_end_length
74     )))
75     LFBFL_offset=$((LFBFL_line_start_length-2))
76     # Checking if increasing number of digits in the line numbers
77     # (99 -> 100).
78     LFBFL_substring1=${LFBFL_previous_line:${LFBFL_offset}:1}
79     # Nice puzzle below :) Look for comments above :)
80     case ${LFBFL_substring1} in
81         -) ((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
40     if [[ ! -o pipefail ]]; then
41         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
42         set -o pipefail
43         trap 'set +o pipefail' RETURN
44     fi
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"\n
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='5e6560b6f4d0792ab065e5ee5746560d68aa8e0609ffe'
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}/${LFBFL_file_name}"
86 local LFBFL_correct_sha512
87 LFBFL_correct_sha512='d473359eaef8164f8e7ab85fc8812b3ce46e2731ef96'
88 LFBFL_correct_sha512+='ff46df693890131abf064c9f2b1beacbabd50aad778f'
89 LFBFL_correct_sha512+='6d3b9b669bc0469f868f1ae69660fd4b89990f1'
90 wget_sha512 "${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='d07a731c4f6cd5acdd8ac2efe3b314b4c2573c5afc452'
118 LFBFL_correct_sha512+='fd3ca3735ffa15e17e5fdc552ad399f21260b4f46c8d'
119 LFBFL_correct_sha512+='ab340495ec115aad7c16892f9716bde665f30b3'
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='b9f07756772b5c8ad6454d1ad907a9ef153dcade8962a'

```

```

149 LFBFL_correct_sha512+=' ac7d2493aef73603753850734a006c731bb71a4c35ff'
150 LFBFL_correct_sha512+=' d74d8d59d971a53e553b8d4c81e1ff8ee5c7b35'
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='d3b80f26b922dbfbdf8ca44134b1350e0746d6703c2a7'
162 LFBFL_correct_sha512+='1cd147ac7f870c52428d6b975a743f905254159658e8'
163 LFBFL_correct_sha512+='6d66a3ea408b473e1f73911a2547b876077223b'
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}/${LFBFL_file_name}"
171 local LFBFL_correct_sha512
172 LFBFL_correct_sha512='9f9aa56d9b90d5003a12c5ec3370ac31aab74fc4a2524'
173 LFBFL_correct_sha512+='2c5fbcb0197b0abbb8242f62dafc76eaf60faa5da173'
174 LFBFL_correct_sha512+='05924b31b64b037083532fcfd9f98445000e2d0'
175 wget_sha512 "${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='d60575393ddf0ed29ad0b7f5e8614812de4c4d61ce094'
214 LFBFL_correct_sha512+='b0e03dbc3298a7cd5696d5e5d4f875cf51624e334e1'
215 LFBFL_correct_sha512+='f8abd24fddfe688dde9bcd2afada03c9d9a468b'
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='25c3e5082951260295b5f0e87ba1f2c978efde1e8926'
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='8df9334f8204bdb9b098f638d15a3db2dd0ce4446203f'
234 LFBFL_correct_sha512+='fe26678cb3c2b3b26cb0b89957490af5fa21c149a525'
235 LFBFL_correct_sha512+='baf5e3d4999f432f9ff384616631cdc111507cf'
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='d4236ad2b43dfebbaa3eead93b953f735821b8b3548d8'
284 LFBFL_correct_sha512+='8e12f8f2e5640f03b329a61e7352bc561bc1aa35a6e5'
285 LFBFL_correct_sha512+='65b93280b0521f714f20bb4f9fcf4c299da04a6'
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='4a293149e4a4c2c09405ecd689007755d172f4f8996a0'
294      LFBFL_correct_sha512+='bbe8668d20a003a001f72d09a3bd46d07187de4788c4'
295      LFBFL_correct_sha512+='54415ea3a428e7563b36963a7f12acb9850297d'
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='11b509de8c58906143b8b7152b636313e86bf3de748d7'
304      LFBFL_correct_sha512+='20ccc70ecae35379ab3c568dec2e746378609fd229c'
305      LFBFL_correct_sha512+='e644f24ac30e8c5958d0af12e529afc818d5422'
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_upgrade_venvs=0
346 declare -r LFBFL_upgrade_venvs_ts_file=\
347 "build_and_checks_variables/upgrade_venvs_ts"
348 declare -i LFBFL_upgrade_venvs_ts
349 declare -i LFBFL_current_ts
350 local LFBFL_upgrade_venvs_answer
351
352 upgrade_venvs_time_interval_in_seconds=""
353 grep_variable "${LFBFL_data_file_name}" \
354     upgrade_venvs_time_interval_in_seconds
355 case ${upgrade_venvs_time_interval_in_seconds} in
356     wat) upgrade_venvs_time_interval_in_seconds=$((RANDOM));;
357     watyouwant?) upgrade_venvs_time_interval_in_seconds=$((SRANDOM));;
358     *) echo "No wat for you?";; #TempsDeCerveauDisponible XD SC2249 ;;
359 esac
360
361 if [[ -f "${LFBFL_upgrade_venvs_ts_file}" ]]; then
362     LFBFL_upgrade_venvs_ts=$((
363         stat -c %Y "${LFBFL_upgrade_venvs_ts_file}"
364     ))
365     LFBFL_current_ts=$((date +%s))
366     ((LFBFL_current_ts--=LFBFL_upgrade_venvs_ts))
367     ((LFBFL_current_ts-=upgrade_venvs_time_interval_in_seconds))
368     if [[ LFBFL_current_ts -gt 0 ]]; then
369         LFBFL_upgrade_venvs=1
370     fi
371 else
372     LFBFL_upgrade_venvs=1
373 fi
374 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
375     upgrade_venvs=""
376     grep_variable "${LFBFL_data_file_name}" upgrade_venvs
377     if [[ "$upgrade_venvs" != "auto" ]]; then
378         read -r -n 1 -t 10\

```

```

379         -p "Upgrade venvs and composer global? [Y/n]" \
380         LFBFL_upgrade_venvs_answer
381     if [[ "${LFBFL_upgrade_venvs_answer}" == "n" ]]; then
382         LFBFL_upgrade_venvs=0
383     fi
384     fi
385 fi
386
387 echo "Building license headers"
388 "./${LFBFL_subdir2}/build_licenses_templates.exec.sh" \
389     "${LFBFL_working_directory}" "${LFBFL_verbose}"
390
391 echo "Building README.md"
392 "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
393     "${LFBFL_working_directory}" "README" "${LFBFL_verbose}"
394
395 echo "Building other MarkDown files"
396 local LFBFL_some_directory
397 declare -r LFBFL_readme="${LFBFL_working_directory}/README.md.tpl"
398 find "${LFBFL_working_directory}" -name "*.md.tpl" \
399     | relevant_find \
400     | while read -r LFBFL_file_path;
401 do
402     [[ "${LFBFL_file_path}" == "${LFBFL_readme}" ]] && continue
403     echo "Found template ${LFBFL_file_path}"
404     LFBFL_some_directory=$(dirname "${LFBFL_file_path}")
405     LFBFL_file_name=$(basename "${LFBFL_file_path}")
406     LFBFL_file_name=${LFBFL_file_name%.md.tpl}
407     "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
408         "${LFBFL_some_directory}" "${LFBFL_file_name}" \
409         "${LFBFL_verbose}"
410 done
411
412 declare -i LFBFL_cd_result
413 pushd .
414 cd "${LFBFL_working_directory}" || {
415     LFBFL_cd_result=$?
416     echo "common_build_and_checks.exec.sh no such directory"
417     # shellcheck disable=SC2248
418     return ${LFBFL_cd_result}
419 }
420
421 echo "Running shellcheck"
422 find . -name "*.sh" \
423     | relevant_find \
424     | while read -r LFBFL_file_path;

```

```

425 do
426     shellcheck --rcfile=build_and_checks_variables/shellcheck.ini \
427         "${LFBFL_file_path}"
428 done
429
430 echo "---Python---"
431 echo "Running isort"
432 isort_venv=""
433 grep_variable "${LFBFL_data_file_name}" isort_venv
434 if [[ -n "${isort_venv}" ]]; then
435     if command -v deactivate
436     then
437         deactivate
438     fi
439     # shellcheck disable=SC1090,SC1091
440     source "${isort_venv}/bin/activate"
441 fi
442 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
443     pip install --upgrade isort
444 fi
445 isort .
446 if [[ -n "${isort_venv}" ]]; then
447     deactivate
448 fi
449 python_isort_complement
450
451 echo "Running black"
452 black_venv=""
453 grep_variable "${LFBFL_data_file_name}" black_venv
454 if [[ -n "${black_venv}" ]]; then
455     if command -v deactivate
456     then
457         deactivate
458     fi
459     # shellcheck disable=SC1090,SC1091
460     source "${black_venv}/bin/activate"
461 fi
462 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
463     pip install --upgrade black
464 fi
465 black .
466 if [[ -n "${black_venv}" ]]; then
467     deactivate
468 fi
469 python_black_complement
470

```

```

471 mypy_venv=""
472 grep_variable "${LFBFL_data_file_name}" mypy_venv
473 if [[ -n "${mypy_venv}" ]]; then
474     if command -v deactivate
475         then
476             deactivate
477         fi
478     # shellcheck disable=SC1090,SC1091
479     source "${mypy_venv}/bin/activate"
480 fi
481 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
482     pip install --upgrade mypy
483 fi
484 shopt -s lastpipe
485 local LFBFL_directory_path
486 declare -i LFBFL_no_toml=1
487 find . -name "pyproject.toml" \
488     | relevant_find\
489     | while read -r LFBFL_file_path;
490 do
491     if grep -q "Typing :: Typed" "${LFBFL_file_path}"; then
492         echo "Running mypy"
493         LFBFL_directory_path=$(dirname "${LFBFL_file_path}")
494         mypy "${LFBFL_directory_path}"
495     fi
496     LFBFL_no_toml=0
497 done
498 if [[ LFBFL_no_toml -eq 1 ]]; then
499     echo "Running mypy"
500     mypy .
501 fi
502 if [[ -n "${mypy_venv}" ]]; then
503     deactivate
504 fi
505 echo "Running bandit"
506 bandit_venv=""
507 grep_variable "${LFBFL_data_file_name}" bandit_venv
508 if [[ -n "${bandit_venv}" ]]; then
509     if command -v deactivate
510         then
511             deactivate
512         fi
513     # shellcheck disable=SC1090,SC1091
514     source "${bandit_venv}/bin/activate"
515 fi

```

```

517 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
518     pip install --upgrade bandit
519 fi
520 bandit --ini build_and_checks_variables/bandit.ini \
521     -b build_and_checks_variables/bandit_baseline.json \
522     -r .
523 # Saving new baseline in temp if necessary.
524 bandit --ini build_and_checks_variables/bandit.ini \
525     -f json -o build_and_checks_variables/temp/bandit_baseline.json \
526     -r .
527 if [[ -n "${bandit_venv}" ]]; then
528     deactivate
529 fi
530
531 echo "Running pylint"
532 pylint_venv=""
533 grep_variable "${LFBFL_data_file_name}" pylint_venv
534 if [[ -n "${pylint_venv}" ]]; then
535     if command -v deactivate
536         then
537             deactivate
538         fi
539     # shellcheck disable=SC1090,SC1091
540     source "${pylint_venv}/bin/activate"
541 fi
542 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
543     pip install --upgrade pylint
544 fi
545 pylint --rcfile build_and_checks_variables/pylintrc.toml \
546     --recursive=y .
547 if [[ -n "${pylint_venv}" ]]; then
548     deactivate
549 fi
550
551 echo "Running ruff"
552 ruff_venv=""
553 grep_variable "${LFBFL_data_file_name}" ruff_venv
554 if [[ -n "${ruff_venv}" ]]; then
555     if command -v deactivate
556         then
557             deactivate
558         fi
559     # shellcheck disable=SC1090,SC1091
560     source "${ruff_venv}/bin/activate"
561 fi
562 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then

```

```

563     pip install --upgrade ruff
564 fi
565 ruff check --config build_and_checks_variables/ruff.toml
566 if [[ -n "${ruff_venv}" ]]; then
567     deactivate
568 fi
569 echo "---Python end---"
570
571 echo "---PHP---"
572 echo "Running PHPMD"
573 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
574     composer global require phpmd/phpmd
575 fi
576
577 local LFBFL_phpmd_baseline="build_and_checks_variables/"
578 LFBFL_phpmd_baseline+="phpmd_baseline.xml"
579 readonly LFBFL_phpmd_baseline
580
581 local LFBFL_temp_phpmd_baseline="build_and_checks_variables/temp/"
582 LFBFL_temp_phpmd_baseline+="phpmd_baseline.xml"
583 readonly LFBFL_temp_phpmd_baseline
584
585 local LFBFL_phpmd_rulesets="cleancode,codesize,controversial,"
586 LFBFL_phpmd_rulesets+="design,naming,unusedcode"
587 readonly LFBFL_phpmd_rulesets
588
589 phpmd --color --baseline-file "${LFBFL_phpmd_baseline}" \
590     . text "${LFBFL_phpmd_rulesets}"
591 # Saving new baseline in temp if necessary.
592 rm "${LFBFL_temp_phpmd_baseline}"
593 phpmd --color --generate-baseline \
594     --baseline-file "${LFBFL_temp_phpmd_baseline}" \
595     . text "${LFBFL_phpmd_rulesets}"
596 echo "---PHP end---"
597
598 echo "---JS---"
599 npm_lint_directories=""
600 grep_variable "${LFBFL_data_file_name}" npm_lint_directories
601 if [[ -n "${npm_lint_directories}" ]]; then
602     echo "Running ESLint"
603     local LFBFL_JS_directory
604     echo "${npm_lint_directories}" \
605         | sed -e 's/\n/\n/g' \
606         | while read -r LFBFL_JS_directory;
607     do
608         (cd "${LFBFL_JS_directory}" && npm run lint)

```

```

609     done
610 fi
611 echo "---JS end---"
612
613 echo "Analyzing too long lines"
614 too_long_code_lines "$@"
615
616 echo "Analyzing shell scripts beginnings"
617 check_shell_scripts_beginnings\
618     | relevant_grep
619
620 echo "Analyzing shell scripts indentation"
621 check_shell_scripts_indentation\
622     | relevant_grep
623
624 echo "Analyzing URLs"
625 check_URLs "$@" \
626     | relevant_grep
627
628 echo "Analyzing strange characters: hover over in doubt"
629 # shellcheck disable=SC1111
630 LFBFL_usual_characters="\x00-\x7Fàâéèëïîôûç©“”└─|«»"
631 grep --exclude-dir .git --color=always\
632     -nPr "[^${LFBFL_usual_characters}]" .
633
634 echo "Checking listed files"
635 "./${LFBFL_subdir}/update_or_check_files_names_listing.exec.sh"
636
637 echo "Creating the PDF file of the listing of the source code"
638 "./${LFBFL_subdir}/create_PDF.exec.sh" "${LFBFL_verbose}"
639
640 if [[ -f "build_and_checks_variables/post_build.sh" ]]; then
641     ./build_and_checks_variables/post_build.sh
642 fi
643
644 # -----
645 # It is at the end that all the HTML files are generated and that
646 # we can check there are no mistakes.
647 echo "Running Nu W3C HTML CSS and SVG validator"
648 upgrade_vnu_jar_time_interval_in_seconds=""
649 grep_variable "${LFBFL_data_file_name}" \
650     upgrade_vnu_jar_time_interval_in_seconds
651 if [[ "${upgrade_vnu_jar_time_interval_in_seconds}" == "wat" ]]; then
652     upgrade_vnu_jar_time_interval_in_seconds=${RANDOM}
653 fi

```

```

655
656 declare -i LFBFL_vnu_jar_ts
657 vnu_jar_path=""
658 grep_variable "${LFBFL_data_file_name}" vnu_jar_path
659 if [[ -f "${vnu_jar_path}" ]]; then
660     LFBFL_vnu_jar_ts=$(stat -c %Y "${vnu_jar_path}")
661     LFBFL_current_ts=$(date +%s)
662     ((LFBFL_current_ts--=LFBFL_vnu_jar_ts))
663     ((LFBFL_current_ts-=upgrade_vnu_jar_time_interval_in_seconds))
664     if [[ LFBFL_current_ts -gt 0 ]]; then
665         echo $'!/\\ATTENTION: Nu W3C validator is too old./!\\\''
666         echo "Go get the latest version here:"
667         echo "https://github.com/validator/validator/releases"
668     fi
669     declare -r LFBFL_files_for_Nu=${(
670         find . -iregex ".*\.(css|html|svg|xhtml)" \
671         | relevant_find
672     )}
673     declare -a LFBFL_files_for_Nu2
674     mapfile -t LFBFL_files_for_Nu2 <<< "${LFBFL_files_for_Nu}"
675     readonly LFBFL_files_for_Nu2
676     java -Xss128M -jar "${vnu_jar_path}" --exit-zero-always --verbose\
677         "${LFBFL_files_for_Nu2[@]}"
678     fi
679 # -----
680
681     if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
682         touch "${LFBFL_upgrade_venvs_ts_file}"
683     fi
684
685     declare -i LFBFL_popd_result
686     popd || {
687         LFBFL_popd_result=$?
688         echo "common_build_and_checks.exec.sh no popd"
689         # shellcheck disable=SC2248
690         return ${LFBFL_popd_result}
691     }
692 }
693 common_build_and_checks "$@"

```

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_AGPLv3+.tpl"
139 LFBFL_URL="${LFBFL_start_URL2}/${LFBFL_file_name}"
140 LFBFL_file_path="./${LFBFL_subdir2}/${LFBFL_file_name}"
141 @sha512_license_file_header_AGPLv3+.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_upgrade_venvs=0
274 declare -r LFBFL_upgrade_venvs_ts_file=\
275 "build_and_checks_variables/upgrade_venvs_ts"
276 declare -i LFBFL_upgrade_venvs_ts
277 declare -i LFBFL_current_ts
278 local LFBFL_upgrade_venvs_answer
279

```

```

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

```

```

326   find "${LFBFL_working_directory}" -name ".*.md.tpl" \
327     | relevant_find \
328     | while read -r LFBFL_file_path;
329 do
330   [[ "${LFBFL_file_path}" == "${LFBFL_readme}" ]] && continue
331   echo "Found template ${LFBFL_file_path}"
332   LFBFL_some_directory=${(dirname "${LFBFL_file_path}")}
333   LFBFL_file_name=${(basename "${LFBFL_file_path}")}
334   LFBFL_file_name=${LFBFL_file_name%.md.tpl}
335   "./${LFBFL_subdir}/build_md_from_printable_md.exec.sh" \
336   "${LFBFL_some_directory}" "${LFBFL_file_name}" \
337   "${LFBFL_verbose}"
338 done
339
340 declare -i LFBFL_cd_result
341 pushd .
342 cd "${LFBFL_working_directory}" || {
343   LFBFL_cd_result=$?
344   echo "common_build_and_checks.exec.sh no such directory"
345   # shellcheck disable=SC2248
346   return ${LFBFL_cd_result}
347 }
348
349 echo "Running shellcheck"
350 find . -name "*.*sh" \
351   | relevant_find \
352   | while read -r LFBFL_file_path;
353 do
354   shellcheck --rcfile=build_and_checks_variables/shellcheck.ini \
355   "${LFBFL_file_path}"
356 done
357
358 echo "---Python---"
359 echo "Running isort"
360 isort_venv=""
361 grep_variable "${LFBFL_data_file_name}" isort_venv
362 if [[ -n "${isort_venv}" ]]; then
363   if command -v deactivate
364   then
365     deactivate
366   fi
367   # shellcheck disable=SC1090,SC1091
368   source "${isort_venv}/bin/activate"
369 fi
370 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
371   pip install --upgrade isort

```

```

372     fi
373     isort .
374     if [[ -n "${isort_venv}" ]]; then
375         deactivate
376     fi
377     python_isort_complement
378
379     echo "Running black"
380     black_venv=""
381     grep_variable "${LFBFL_data_file_name}" black_venv
382     if [[ -n "${black_venv}" ]]; then
383         if command -v deactivate
384             then
385                 deactivate
386             fi
387             # shellcheck disable=SC1090,SC1091
388             source "${black_venv}/bin/activate"
389     fi
390     if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
391         pip install --upgrade black
392     fi
393     black .
394     if [[ -n "${black_venv}" ]]; then
395         deactivate
396     fi
397     python_black_complement
398
399     mypy_venv=""
400     grep_variable "${LFBFL_data_file_name}" mypy_venv
401     if [[ -n "${mypy_venv}" ]]; then
402         if command -v deactivate
403             then
404                 deactivate
405             fi
406             # shellcheck disable=SC1090,SC1091
407             source "${mypy_venv}/bin/activate"
408     fi
409     if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
410         pip install --upgrade mypy
411     fi
412     shopt -s lastpipe
413     local LFBFL_directory_path
414     declare -i LFBFL_no_toml=1
415     find . -name "pyproject.toml" \
416         | relevant_find\
417         | while read -r LFBFL_file_path;

```

```

418 do
419     if grep -q "Typing :: Typed" "${LFBFL_file_path}"; then
420         echo "Running mypy"
421         LFBFL_directory_path=$(dirname "${LFBFL_file_path}")
422         mypy "${LFBFL_directory_path}"
423     fi
424     LFBFL_no_toml=0
425 done
426 if [[ LFBFL_no_toml -eq 1 ]]; then
427     echo "Running mypy"
428     mypy .
429 fi
430 if [[ -n "${mypy_venv}" ]]; then
431     deactivate
432 fi
433
434 echo "Running bandit"
435 bandit_venv=""
436 grep_variable "${LFBFL_data_file_name}" bandit_venv
437 if [[ -n "${bandit_venv}" ]]; then
438     if command -v deactivate
439     then
440         deactivate
441     fi
442     # shellcheck disable=SC1090,SC1091
443     source "${bandit_venv}/bin/activate"
444 fi
445 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
446     pip install --upgrade bandit
447 fi
448 bandit --ini build_and_checks_variables/bandit.ini \
449     -b build_and_checks_variables/bandit_baseline.json \
450     -r .
451 # Saving new baseline in temp if necessary.
452 bandit --ini build_and_checks_variables/bandit.ini \
453     -f json -o build_and_checks_variables/temp/bandit_baseline.json \
454     -r .
455 if [[ -n "${bandit_venv}" ]]; then
456     deactivate
457 fi
458
459 echo "Running pylint"
460 pylint_venv=""
461 grep_variable "${LFBFL_data_file_name}" pylint_venv
462 if [[ -n "${pylint_venv}" ]]; then
463     if command -v deactivate

```

```

464     then
465         deactivate
466     fi
467     # shellcheck disable=SC1090,SC1091
468     source "${pylint_venv}/bin/activate"
469   fi
470   if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
471     pip install --upgrade pylint
472   fi
473   pylint --rcfile build_and_checks_variables/pylintrc.toml \
474     --recursive=y .
475   if [[ -n "${pylint_venv}" ]]; then
476     deactivate
477   fi
478
479   echo "Running ruff"
480   ruff_venv=""
481   grep_variable "${LFBFL_data_file_name}" ruff_venv
482   if [[ -n "${ruff_venv}" ]]; then
483     if command -v deactivate
484       then
485         deactivate
486       fi
487     # shellcheck disable=SC1090,SC1091
488     source "${ruff_venv}/bin/activate"
489   fi
490   if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
491     pip install --upgrade ruff
492   fi
493   ruff check --config build_and_checks_variables/ruff.toml
494   if [[ -n "${ruff_venv}" ]]; then
495     deactivate
496   fi
497   echo "---Python end---"
498
499   echo "---PHP---"
500   echo "Running PHPMD"
501   if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
502     composer global require phpmd/phpmd
503   fi
504
505   local LFBFL_phpmd_baseline="build_and_checks_variables/"
506   LFBFL_phpmd_baseline+="phpmd_baseline.xml"
507   readonly LFBFL_phpmd_baseline
508
509   local LFBFL_temp_phpmd_baseline="build_and_checks_variables/temp/"

```

```

510 LFBFL_temp_phpmd_baseline+="phpmd_baseline.xml"
511 readonly LFBFL_temp_phpmd_baseline
512
513 local LFBFL_phpmd_rulesets="cleancode,codesize,controversial,"
514 LFBFL_phpmd_rulesets+="design,naming,unusedcode"
515 readonly LFBFL_phpmd_rulesets
516
517 phpmd --color --baseline-file "${LFBFL_phpmd_baseline}" \
518   . text "${LFBFL_phpmd_rulesets}"
519 # Saving new baseline in temp if necessary.
520 rm "${LFBFL_temp_phpmd_baseline}"
521 phpmd --color --generate-baseline \
522   --baseline-file "${LFBFL_temp_phpmd_baseline}" \
523   . text "${LFBFL_phpmd_rulesets}"
524 echo "----PHP end---"
525
526 echo "----JS---"
527 npm_lint_directories=""
528 grep_variable "${LFBFL_data_file_name}" npm_lint_directories
529 if [[ -n "${npm_lint_directories}" ]]; then
530   echo "Running ESLint"
531   local LFBFL_JS_directory
532   echo "${npm_lint_directories}" \
533     | sed -e 's/\n/\n/g' \
534     | while read -r LFBFL_JS_directory;
535   do
536     (cd "${LFBFL_JS_directory}" && npm run lint)
537   done
538 fi
539 echo "----JS end---"
540
541 echo "Analyzing too long lines"
542 too_long_code_lines "$@"
543
544 echo "Analyzing shell scripts beginnings"
545 check_shell_scripts_beginnings \
546   | relevant_grep
547
548 echo "Analyzing shell scripts indentation"
549 check_shell_scripts_indentation \
550   | relevant_grep
551
552 echo "Analyzing URLs"
553 check_URLs "$@" \
554   | relevant_grep
555

```

```

556 echo "Analyzing strange characters: hover over in doubt"
557 # shellcheck disable=SC1111
558 LFBFL_usual_characters="\x00-\x7Fàâéèëïïôûç©“”└_|«»"
559 grep --exclude-dir .git --color=always \
560   -nPr "[^${LFBFL_usual_characters}]" .
561
562 echo "Checking listed files"
563 "./${LFBFL_subdir}/update_or_check_files_names_listing.exec.sh"
564
565 echo "Creating the PDF file of the listing of the source code"
566 "./${LFBFL_subdir}/create_PDF.exec.sh" "${LFBFL_verbose}"
567
568 if [[ -f "build_and_checks_variables/post_build.sh" ]]; then
569   ./build_and_checks_variables/post_build.sh
570 fi
571
572 # -----
573 # It is at the end that all the HTML files are generated and that
574 # we can check there are no mistakes.
575 echo "Running Nu W3C HTML CSS and SVG validator"
576 upgrade_vnu_jar_time_interval_in_seconds=""
577 grep_variable "${LFBFL_data_file_name}" \
578   upgrade_vnu_jar_time_interval_in_seconds
579 if [[ "${upgrade_vnu_jar_time_interval_in_seconds}" == "wat" ]]; then
580
581   upgrade_vnu_jar_time_interval_in_seconds=${RANDOM}
582 fi
583
584 declare -i LFBFL_vnu_jar_ts
585 vnu_jar_path=""
586 grep_variable "${LFBFL_data_file_name}" vnu_jar_path
587 if [[ -f "${vnu_jar_path}" ]]; then
588   LFBFL_vnu_jar_ts=$((stat -c %Y "${vnu_jar_path}") )
589   LFBFL_current_ts=$(date +%s)
590   ((LFBFL_current_ts-=LFBFL_vnu_jar_ts))
591   ((LFBFL_current_ts-=upgrade_vnu_jar_time_interval_in_seconds))
592   if [[ LFBFL_current_ts -gt 0 ]]; then
593     echo $'!/\\ATTENTION: Nu W3C validator is too old./!\\\''
594     echo "Go get the latest version here:"
595     echo "https://github.com/validator/validator/releases"
596   fi
597   declare -r LFBFL_files_for_Nu=$((
598     find . -iregex ".*\.(css|html|svg|xhtml)" \
599       | relevant_find
600   ))
601   declare -a LFBFL_files_for_Nu2

```

```

602 mapfile -t LFBFL_files_for_Nu2 <<< "${LFBFL_files_for_Nu}"
603 readonly LFBFL_files_for_Nu2
604 java -Xss128M -jar "${vnu_jar_path}" --exit-zero-always --verbose\
605     "${LFBFL_files_for_Nu2[@]} "
606 fi
607 # -----
608
609 if [[ LFBFL_upgrade_venvs -eq 1 ]]; then
610     touch "${LFBFL_upgrade_venvs_ts_file}"
611 fi
612
613 declare -i LFBFL_popd_result
614 popd || {
615     LFBFL_popd_result=$?
616     echo "common_build_and_checks.exec.sh no popd"
617     # shellcheck disable=SC2248
618     return ${LFBFL_popd_result}
619 }
620 }
621
622 common_build_and_checks "$@"

```

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     repository_name=""
57     grep_variable "${LFBFL_data_file_name}" repository_name
58
59     abstract=""
60     grep_variable "${LFBFL_data_file_name}" abstract
61
62     acknowledgments=""
63     grep_variable "${LFBFL_data_file_name}" acknowledgments

```

```

64
65     author_email="""
66     grep_variable "${LFBFL_data_file_name}" author_email
67
68     author_full_name"""
69     grep_variable "${LFBFL_data_file_name}" author_full_name
70
71     author_website"""
72     grep_variable "${LFBFL_data_file_name}" author_website
73
74     declare -r LFBFL_current_date=$(date -I"date")
75
76     declare -r LFBFL_current_git_SHA1=$(git rev-parse HEAD)
77
78     declare -ir LFBFL_number_of_commits=$(
79         git shortlog\
80         | space_starting_lines\
81         | wc -l
82     )
83
84     declare -ir LFBFL_all_code_lines_count=$((code_lines_count_all "$@"))
85     declare -ir LFBFL_not_empty_code_lines_count=$((code_lines_count_not_empty "$@"))
86     declare -ir LFBFL_empty_code_lines_count=$((code_lines_count_empty "$@"))
87
88     local LFBFL_number_of_lines="${LFBFL_all_code_lines_count}"
89     LFBFL_number_of_lines+=" total lines,"
90     LFBFL_number_of_lines+=" ${LFBFL_not_empty_code_lines_count}"
91     LFBFL_number_of_lines+=" not empty lines,"
92     LFBFL_number_of_lines+=" ${LFBFL_empty_code_lines_count}"
93     LFBFL_number_of_lines+=" empty lines."
94     readonly LFBFL_number_of_lines
95
96     tree -a --gitignore\
97         -I "node_modules/"\
98         -I "__pycache__/"\
99         -I ".mypy_cache/"\
100        -I ".git/"\
101        | replace_non_ascii_spaces\
102        > "${LFBFL_subdir2}/temp/current_tree_light.txt"
103
104     tree -a -DFh --gitignore\
105         -I "node_modules/"\

```

```

110     -I "__pycache__/" \
111     -I ".mypy_cache/" \
112     -I ".git/" \
113     | replace_non_ascii_spaces \
114     > "${LFBFL_subdir2}/temp/current_tree.txt.temp"
115
116 local LFBFL_temp_files_listing="./${LFBFL_subdir2}/temp/"
117 LFBFL_temp_files_listing+="files_listing.tex.tpl.temp"
118 readonly LFBFL_temp_files_listing
119 : > "${LFBFL_temp_files_listing}"
120 local LFBFL_temp_files_listing2="./${LFBFL_subdir2}/temp/"
121 LFBFL_temp_files_listing2+="files_listing.html.tpl.temp"
122 readonly LFBFL_temp_files_listing2
123 : > "${LFBFL_temp_files_listing2}"
124 # HTML <li> elements, hence "lis".
125 local LFBFL_temp_files_lis
126 LFBFL_temp_files_lis="./${LFBFL_subdir2}/temp/files_lis.html.tpl"
127 readonly LFBFL_temp_files_lis
128 : > "${LFBFL_temp_files_lis}"
129 get_split_score_simple 1 70 /
130 declare -r LFBFL_score_command="${get_split_score_result}"
131 local LFBFL_score_command_properties="${get_split_score_result2}"
132 readonly LFBFL_score_command_properties
133 get_split_score_simple 1 70 ':'
134 declare -r LFBFL_score_command2="${get_split_score_result}"
135 local LFBFL_score_command_properties2="${get_split_score_result2}"
136 readonly LFBFL_score_command_properties2
137 declare -r LFBFL_suffix='%'
138 declare -i LFBFL_i=0
139 local LFBFL_cleaned_path1
140 local LFBFL_cleaned_path2
141 local LFBFL_new_lines
142 local LFBFL_new_lines2
143 local LFBFL_new_lines3
144 # Remove line returns here to keep lines short.
145 sed -Ez 's/\\n//Mg' "./${LFBFL_subdir2}/files_names_listing.txt" \
146   | grep -v '^//\' \
147   | while read -r LFBFL_file_name;
148 do
149   echo "Listing file for tex/HTML : ${LFBFL_file_name}"
150   # echo "Listing file for tex : ${LFBFL_file_name}"
151   # LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
152   LFBFL_cleaned_path1="${LFBFL_file_name//_/_}"
153   LFBFL_cleaned_path2=$(
154     echo "${LFBFL_file_name}" \
155     | sed -e 's//:/g' -e 's/\.\./g'

```

```

156 )
157 LFBFL_new_lines="${LFBFL_cleaned_path1}"
158 if [[ ${#LFBFL_new_lines} -gt 70 ]]; then
159     repeated_split_last_line "${LFBFL_new_lines}" "" 70 \
160         "${LFBFL_suffix}" "${LFBFL_score_command}" \
161         "${LFBFL_score_command_properties}" "\\" \
162     LFBFL_new_lines=${repeated_split_last_line_result}
163 fi
164 LFBFL_new_lines2="${LFBFL_cleaned_path2}"
165 if [[ ${#LFBFL_new_lines2} -gt 70 ]]; then
166     repeated_split_last_line "${LFBFL_new_lines2}" "" 70 \
167         "${LFBFL_suffix}" "${LFBFL_score_command2}" \
168         "${LFBFL_score_command_properties2}" \
169     LFBFL_new_lines2=${repeated_split_last_line_result}
170 fi
171 LFBFL_new_lines3="${LFBFL_file_name}"
172 if [[ ${#LFBFL_new_lines3} -gt 70 ]]; then
173     repeated_split_last_line "${LFBFL_new_lines3}" "" 70 \
174         "${LFBFL_suffix}" "${LFBFL_score_command}" \
175         "${LFBFL_score_command_properties}" \
176     LFBFL_new_lines3=${repeated_split_last_line_result}
177 fi
178 {
179     echo "\subsection{"
180     echo "  ${LFBFL_file_name}" \
181         | sed -e "s|  ${LFBFL_file_name}|${LFBFL_new_lines}|g" \
182             -e 's/_/\_\_/'g'
183     echo "}"
184     echo "\label{"
185     echo "  ${LFBFL_cleaned_path2}" \
186         | sed -e "s|  ${LFBFL_cleaned_path2}|${LFBFL_new_lines2}|g"
187     echo "}"
188     echo ""
189     echo "\VerbatimInput[numbers=left,xleftmargin=-5mm]{"
190     echo "  ${LFBFL_file_name}" \
191         | sed -e "s|  ${LFBFL_file_name}|${LFBFL_new_lines3}|g"
192     echo "}"
193     echo ""
194     echo ""
195 } >> "${LFBFL_temp_files_listing}"
196
197 # echo "Listing file for HTML : ${LFBFL_file_name}"
198 ((++LFBFL_i))
199 LFBFL_new_lines="${LFBFL_file_name}"
200 if [[ ${#LFBFL_file_name} -gt 70 ]]; then
201     repeated_split_last_line "${LFBFL_new_lines}" "-->" \

```

```

202     70  "<!--" "${LFBFL_score_command}" \
203     "${LFBFL_score_command_properties}" \
204     LFBFL_new_lines=${repeated_split_last_line_result}
205 fi
206 {
207     echo "<h3 id=\"subsection2.${LFBFL_i}\">2.${LFBFL_i}"
208     # shellcheck disable=SC2001
209     echo "${LFBFL_file_name}" \
210         | sed -e "s|${LFBFL_file_name}|${LFBFL_new_lines}|g"
211     echo "</h3>"
212     echo "<pre class=\"numbered_lines\">"
213     sed -e 's/&/&#x26;/g' -e 's/</&lt;/g' \
214         < "${LFBFL_file_name}"
215     echo "</pre>"
216     echo ""
217     echo ""
218 } >> "${LFBFL_temp_files_listing2}"
219 {
220     echo "      <li><a href=\"#${subsection2.${LFBFL_i}}\">"
221     # shellcheck disable=SC2001
222     echo "${LFBFL_file_name}" \
223         | sed -e "s|${LFBFL_file_name}|${LFBFL_new_lines}|g"
224     echo "        </a></li>"
225 } >> "${LFBFL_temp_files_lis}"
226 done
227 overwrite_if_not_equal "./${LFBFL_subdir2}/files_listing.tex.tpl" \
228     "${LFBFL_temp_files_listing}"
229 sed -i -e "s|\n</pre>|</pre>|g" "${LFBFL_temp_files_listing2}"
230 overwrite_if_not_equal \
231     "./${LFBFL_subdir2}/temp/files_listing.html.tpl" \
232     "${LFBFL_temp_files_listing2}"
233
234 # We verify if some lines are beyond 70 characters
235 # in current_tree_light.txt and current_tree.txt.
236 declare -ar LFBFL_trees=(
237     "current_tree_light.txt"
238     "current_tree.txt.temp"
239 )
240 local LFBFL_tree_path
241 local LFBFL_line
242 local LFBFL_prefix
243 local LFBFL_line_start
244 for LFBFL_tree in "${LFBFL_trees[@]}"; do
245     LFBFL_tree_path="${LFBFL_subdir2}/temp/${LFBFL_tree}"
246     grep '\{71\}' "${LFBFL_tree_path}" \
247         | while read -r LFBFL_line;

```

```

248 do
249     # echo "LFBFL_line: ${LFBFL_line}"
250     LFBFL_prefix=$(
251         echo "${LFBFL_line}"\
252             | sed -E -e 's/(.*)-[^-]+$/\1/g' -e 's/[^\^ ]+$/\g'\
253     )
254     LFBFL_prefix+="| "
255     # echo "LFBFL_prefix: ${LFBFL_prefix}"
256     LFBFL_file_name=$(
257         echo "${LFBFL_line}"\
258             | sed -E 's|.* ([a-zA-Z0-9\._/-]+).)$|\1|g'
259     )
260     # echo "LFBFL_file_name: ${LFBFL_file_name}"
261     LFBFL_line_start=$(
262         echo "${LFBFL_line}"\
263             | sed -E "s/(.*)[ ]*${LFBFL_file_name}/\1/g"\  

264                 -e 's/ *$/\g'
265     )
266     # echo "LFBFL_line_start: ${LFBFL_line_start}"
267     LFBFL_line=$(
268         echo "${LFBFL_line}"\  

269             | sed -E -e 's/\\[ /\\\\\\[ /g' -e 's/\\]/\\\\]/g'
270     )
271     LFBFL_new_lines="${LFBFL_prefix}${LFBFL_file_name}"
272     if [[ ${#LFBFL_new_lines} -gt 70 ]]; then
273         repeated_split_last_line "${LFBFL_new_lines}"\  

274             "${LFBFL_prefix}" 70 "" "" 3
275         LFBFL_new_lines=${repeated_split_last_line_result}
276     fi
277     sed -i -e\  

278         "s/${LFBFL_line}/${LFBFL_line_start}\n${LFBFL_new_lines}/g"\  

279         "${LFBFL_tree_path}"
280     done
281 done
282
283 overwrite_if_not_equal "${LFBFL_subdir2}/temp/current_tree.txt"\  

284     "${LFBFL_subdir2}/temp/current_tree.txt.temp" 1 1
285
286 if [[ -f "./${LFBFL_subdir2}/${repository_name}.tex.tpl" ]]; then
287     # If there is a template, we init the process from it.
288     cp "./${LFBFL_subdir2}/${repository_name}.tex.tpl"\<br/>
289         "./${LFBFL_subdir2}/temp/${repository_name}.tex.1"
290 elif [[ -f "./${LFBFL_subdir2}/${repository_name}.tex" ]]; then
291     # Otherwise if there is a tex file, we init the process from it.
292     cp "./${LFBFL_subdir2}/${repository_name}.tex"\<br/>
293         "./${LFBFL_subdir2}/temp/${repository_name}.tex.1"

```

```

294     else
295         echo "Neither .tex.tpl, nor .tex in ./${LFBFL_subdir2}/"
296     fi
297
298     # Same logic with repository HTML file.
299     if [[ -f "./${LFBFL_subdir2}/${repository_name}.html.tpl" ]]; then
300         cp "./${LFBFL_subdir2}/${repository_name}.html.tpl"\n
301             "./${LFBFL_subdir2}/temp/${repository_name}.html.1"
302     elif [[ -f "./${repository_name}.html" ]]; then
303         cp "./${repository_name}.html"\n
304             "./${LFBFL_subdir2}/temp/${repository_name}.html.1"
305     else
306         echo "Neither .html.tpl in ./${LFBFL_subdir2}/, nor .html in ./"
307     fi
308
309     declare -r LFBFL_tex_path_start=\n
310         "./${LFBFL_subdir2}/temp/${repository_name}.tex"
311     declare -r LFBFL_html_path_start=\n
312         "./${LFBFL_subdir2}/temp/${repository_name}.html"
313
314     # shellcheck disable=SC2001
315     echo "${abstract}"\
316         | sed -e 's/\n/\n/g'\
317         > "./${LFBFL_subdir2}/temp/abstract_temp"
318
319     # shellcheck disable=SC2001
320     echo "${acknowledgments}"\
321         | sed -e 's/\n/\n/g'\
322         > "./${LFBFL_subdir2}/temp/acknowledgments_temp"
323
324     # But the filling still occurs, in case the dev want to refill
325     # part of the tex/html file that he modified with @token@
326     # using some computed results.
327     # Tex filling:
328     if [[ -f "${LFBFL_tex_path_start}.1" ]]; then
329         sed -i "s|@repository_name@|${repository_name}|g"\n
330             "${LFBFL_tex_path_start}.1"
331
332         insert_file_at_token "${LFBFL_tex_path_start}.1" @abstract@\n
333             "./${LFBFL_subdir2}/temp/abstract_temp"\n
334             "${LFBFL_tex_path_start}.2"
335
336         insert_file_at_token "${LFBFL_tex_path_start}.2"@\n
337             acknowledgments@\n
338             "./${LFBFL_subdir2}/temp/acknowledgments_temp"\n
339             "${LFBFL_tex_path_start}.3"

```

```

340
341     sed -i -e "s|@author_email@|${author_email}|g" \
342             -e "s|@author_full_name@|${author_full_name}|g" \
343             -e "s|@author_website@|${author_website}|g" \
344             -e "s|@current_date@|${LFBFL_current_date}|g" \
345             -e "s|@current_git_SHA1@|${LFBFL_current_git_SHA1}|g" \
346             -e "s|@number_of_commits@|${LFBFL_number_of_commits}|g" \
347             -e "s|@number_of_lines@|${LFBFL_number_of_lines}|g" \
348             "${LFBFL_tex_path_start}.3"
349
350     pushd "./${LFBFL_subdir2}/temp/" || (echo "pushd failed" && exit)
351     sed -i \
352         -e '/@current_tree_light@/{r current_tree_light.txt' -e 'd}' \
353         -e '/@current_tree@/{r current_tree.txt' -e 'd}' \
354         "${repository_name}.tex.3"
355     popd || (echo "popd failed" && exit)
356
357     insert_file_at_token "${LFBFL_tex_path_start}.3" \
358             @files_listing_VerbatimInput@ \
359             "./${LFBFL_subdir2}/files_listing.tex.tpl" \
360             "${LFBFL_tex_path_start}.4"
361
362     overwrite_if_not_equal \
363         "./${LFBFL_subdir2}/${repository_name}.tex" \
364         "${LFBFL_tex_path_start}.4" 1
365 fi
366
367 # HTML filling:
368 if [[ -f "${LFBFL_html_path_start}.1" ]]; then
369     sed -i "s|@repository_name@|${repository_name}|g" \
370             "${LFBFL_html_path_start}.1"
371
372     insert_file_at_token "${LFBFL_html_path_start}.1" @abstract@ \
373             "./${LFBFL_subdir2}/temp/abstract_temp" \
374             "${LFBFL_html_path_start}.2"
375
376     insert_file_at_token "${LFBFL_html_path_start}.2" \
377             @acknowledgments@ \
378             "./${LFBFL_subdir2}/temp/acknowledgments_temp" \
379             "${LFBFL_html_path_start}.3"
380
381     sed -i -e "s|@author_email@|${author_email}|g" \
382             -e "s|@author_full_name@|${author_full_name}|g" \
383             -e "s|@author_website@|${author_website}|g" \
384             -e "s|@current_date@|${LFBFL_current_date}|g" \
385             -e "s|@current_git_SHA1@|${LFBFL_current_git_SHA1}|g" \

```

```

386         -e "s|@number_of_commits@|${LFBFL_number_of_commits}|g"\|
387         -e "s|@number_of_lines@|${LFBFL_number_of_lines}|g"\|
388         "${LFBFL_html_path_start}.3"
389
390     insert_file_at_token "${LFBFL_html_path_start}.3"\|
391         @files_lis@ "${LFBFL_temp_files_lis}"\|
392         "${LFBFL_html_path_start}.4"
393
394     pushd "./${LFBFL_subdir2}/temp/" || (echo "pushd failed" && exit)
395     sed -i \
396         -e '@current_tree_light@/{r current_tree_light.txt' -e 'd}' \
397         -e '@current_tree@/{r current_tree.txt' -e 'd}' \
398         "${repository_name}.html.4"
399     popd || (echo "popd failed" && exit)
400
401     insert_file_at_token "${LFBFL_html_path_start}.4"\|
402         @files_listing_HTMLPreInput@ \
403         "./${LFBFL_subdir2}/temp/files_listing.html.tpl"\|
404         "${LFBFL_html_path_start}.5"
405
406     overwrite_if_not_equal "./${repository_name}.html"\|
407         "${LFBFL_html_path_start}.5" 1
408 fi
409
410 if [[ LFBFL_verbose -eq 1 ]]; then
411     for ((i=0; i<3; i++)); do
412         pdflatex "./${LFBFL_subdir2}/${repository_name}.tex"
413         done
414     else
415         for ((i=0; i<3; i++)); do
416             pdflatex "./${LFBFL_subdir2}/${repository_name}.tex"\|
417                 > /dev/null
418             done
419     fi
420
421     LFBFL_files_to_temp=(\
422         "${repository_name}.aux"\|
423         "${repository_name}.log"\|
424         "${repository_name}.out"\|
425         "${repository_name}.toc"
426     )
427     for LFBFL_file_name in "${LFBFL_files_to_temp[@]}"; do
428         mv "${LFBFL_file_name}" "${LFBFL_subdir2}/temp/"
429     done
430 }
431

```

```
432 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() {
91      # $1=$file_name
92      # $2=$token assume that token is the only thing on his line.
93      # $3=$file_name_to_insert
94      # $4=$file_name_result
95      declare -r LFBFL_start_file_name="$1.insert_file_at_token1.temp"
96      declare -r LFBFL_end_file_name="$1.insert_file_at_token2.temp"
97      local LFBFL_result_file_name
98      if [[ -n "$4" ]]; then
99          LFBFL_result_file_name="$4"
100     else
101         LFBFL_result_file_name="$1.insert_file_at_token3.temp"
102     fi
103     readonly LFBFL_result_file_name
104     split_file_in_two "$1" "$2" "${LFBFL_start_file_name}" \
105         "${LFBFL_end_file_name}"
106     cat "${LFBFL_start_file_name}" "$3" "${LFBFL_end_file_name}" \
107         > "${LFBFL_result_file_name}"
108     rm "${LFBFL_start_file_name}" "${LFBFL_end_file_name}"
109     if [[ -z "$4" ]]; then
110         overwrite_if_not_equal "$1" "${LFBFL_result_file_name}"
111     fi
112 }

```

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     grammar_or_spell_checker_command=""
49     grep_variable "$1" grammar_or_spell_checker_command
50     declare -r LFBFL_command=$(
51         echo "${grammar_or_spell_checker_command}" \
52         | sed -Ez -e "s/\n//Mg"
53     )
54     local LFBFL_file_path
55     local LFBFL_pattern
56     local LFBFL_eval_string
57     for LFBFL_pattern in "${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             | while read -r LFBFL_file_path;
62     do

```

```

63     LFBFL_eval_string="${LFBFL_command} ${LFBFL_file_path}"
64     eval "${LFBFL_eval_string}"
65   done
66 done
67 }

```

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_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_license in "${LFBFL_licenses[@]}"; do
135     LFBFL_license_prefix="${LFBFL_source_license_prefix}"
136     LFBFL_license_prefix+="${LFBFL_license}"
137     LFBFL_license_prefix2="${LFBFL_target_license_prefix}"
138     LFBFL_license_prefix2+="${LFBFL_license}"
139     for ((i=0; i<${#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 ${LFBFL_license_file_name}.tpl created"
186             fi
187         fi
188     done
189 done
# -----
191
192 # Now that base license templates generated languages license
193 # templates in working directory, we can go there.
194 declare -i LFBFL_cd_result
195 pushd .
196 cd "${LFBFL_working_directory}" || {
197     LFBFL_cd_result=$?
198     echo "build_licenses_templates no such directory"
199     # shellcheck disable=SC2248
200     return ${LFBFL_cd_result}
201 }
202
203 local LFBFL_data_file_name="build_and_checks_variables/"
204 LFBFL_data_file_name+="repository_data.txt"
205 readonly LFBFL_data_file_name
206 repository_name=""
207 license=""
208 license2=""
209 author_full_name=""
210 grep_variable "${LFBFL_data_file_name}" repository_name
211 grep_variable "${LFBFL_data_file_name}" license
212 grep_variable "${LFBFL_data_file_name}" license2
213 grep_variable "${LFBFL_data_file_name}" author_full_name
214
215 LFBFL_license_prefix="build_and_checks_variables/temp/"
216 LFBFL_license_prefix+="license_file_header_"
217 readonly LFBFL_license_prefix
218 LFBFL_license_prefix2="${LFBFL_license_prefix}${license}"
219 readonly LFBFL_license_prefix2
220 declare -r\
221     LFBFL_license_prefix3="${LFBFL_license_prefix}${license2}"
222

```

```

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

```

```

269
270     declare -Ar LFBFL_all_block_comment_languages=( 
271         ["c"]="c"
272         ["php"]="c"
273         ["py"]="py"
274     )
275
276     local LFBFL_key
277     local LFBFL_dest
278     local LFBFL_license_file_name2
279     declare -i LFBFL_not_subfile
280     declare -i LFBFL_not_subfile2
281     for LFBFL_key in "${!LFBFL_all_block_comment_languages[@]}"; do
282         LFBFL_dest=${LFBFL_all_block_comment_languages[$LFBFL_key]}
283         LFBFL_license_file_name="${LFBFL_license_prefix2}.${LFBFL_dest}"
284         prepare_filled_license_file_block "${LFBFL_license_file_name}"
285         if [[ -n "${license2}" ]]; then
286             LFBFL_license_file_name2="${LFBFL_license_prefix3}"
287             LFBFL_license_file_name2+=".${LFBFL_dest}"
288             prepare_filled_license_file_block "${LFBFL_license_file_name2}"
289         fi
290         find . -type f -name "*.${LFBFL_key}" -printf '%P\n' \
291             | relevant_find \
292             | while read -r LFBFL_file_name;
293         do
294             is_subfile "${LFBFL_file_name}" \
295                 "${LFBFL_license_file_name}.temp"
296             LFBFL_not_subfile=$?
297             LFBFL_not_subfile2=1
298             if [[ -n "${license2}" ]]; then
299                 is_subfile "${LFBFL_file_name}" \
300                     "${LFBFL_license_file_name2}.temp"
301                 LFBFL_not_subfile2=$?
302             fi
303             if [[ LFBFL_not_subfile -ge 1 && LFBFL_not_subfile2 -ge 1 ]]; then
304                 echo "File ${LFBFL_file_name} has no/wrong license header?"
305             fi
306         done
307         rm "${LFBFL_license_file_name}.temp"
308         if [[ -n "${license2}" ]]; then
309             rm "${LFBFL_license_file_name2}.temp"
310         fi
311     done
312
313     prepare_filled_license_file_line()

```

```

315 # $1=LFBFL_license_file_name
316 sed -e "s/@repository_name@/${repository_name}/g" \
317   -e "s/@copyright_string@/${LFBFL_copyright_string}/g" \
318   "$1.tpl" \
319   > "$1.temp"
320 overwrite_if_not_equal "$1" "$1.temp"
321 declare -r LFBFL_result=$?
322 if [[ LFBFL_verbose -eq 1 ]]; then
323   if [[ LFBFL_result -eq 1 ]]; then
324     echo "License template $1 updated"
325   elif [[ LFBFL_result -eq 2 ]]; then
326     echo "License template $1 created"
327   fi
328 fi
329 }
330
331 declare -Ar LFBFL_all_line_comment_languages=(

332   ["sh"]="sh"
333   ["tex"]="tex"
334 )
335
336 for LFBFL_key in "${!LFBFL_all_line_comment_languages[@]}"; do
337   LFBFL_dest=${LFBFL_all_line_comment_languages[$LFBFL_key]}
338   LFBFL_license_file_name="${LFBFL_license_prefix2}.${LFBFL_dest}"
339   prepare_filled_license_file_line "${LFBFL_license_file_name}"
340   if [[ -n "${license2}" ]]; then
341     LFBFL_license_file_name2="${LFBFL_license_prefix3}"
342     LFBFL_license_file_name2+=".${LFBFL_dest}"
343     prepare_filled_license_file_line "${LFBFL_license_file_name2}"
344   fi
345   find . -type f -name "*.${LFBFL_key}" -printf '%P\n' \
346   | relevant_find \
347   | while read -r LFBFL_file_name;
348 do
349   is_subfile "${LFBFL_file_name}" "${LFBFL_license_file_name}"
350   LFBFL_not_subfile=$?
351   LFBFL_not_subfile2=1
352   if [[ -n "${license2}" ]]; then
353     is_subfile "${LFBFL_file_name}" "${LFBFL_license_file_name2}"
354     LFBFL_not_subfile2=$?
355   fi
356   if [[ LFBFL_not_subfile -ge 1 && LFBFL_not_subfile2 -ge 1 ]]; then
357     echo "File ${LFBFL_file_name} has no/wrong license header?"
358   fi
359 done

```

```

361     done
362
363     declare -i LFBFL_popd_result
364     popd || {
365         LFBFL_popd_result=$?
366         echo "build_licenses_templates no popd"
367         # shellcheck disable=SC2248
368         return ${LFBFL_popd_result}
369     }
370 }
371
372 build_licenses_templates "$@"

```

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 Lyaudet (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     declare -r LFBFL_regexp="(?<=^$2=).*$"
135     # echo $LFBFL_regexp
136     declare -r LFBFL_variable_value=$(grep -oP "${LFBFL_regexp}" "$1")
137     # echo $LFBFL_variable_value
138     declare -g "$2"="$LFBFL_variable_value"
139 }
140
141 repository_name=""
142 grep_variable build_and_checks_variables/repository_data.txt \
    repository_name
143
144 empty_lines() {
145     grep '^$'
146 }
147
148 not_empty_lines() {
149     grep -v '^$'
150 }
151
152 space_starting_lines() {
153     grep '^[' ]'
154 }
155
156 not_space_starting_lines() {
157     grep '^[^[:space:]]'
158 }
159
160 empty_lines_after_file_name() {
161     grep '^[:space:]$'
162 }
163
164 not_empty_lines_after_file_name() {
165     grep -v '^[:space:]$'
166 }
167
168 space_starting_lines_after_file_name() {
169

```

```

170     grep '^[:]\+\:' '
171 }
172
173 not_space_starting_lines_after_file_name() {
174     grep '^[:]\+:[^ ]'
175 }
176
177 not_JS_dependencies_find() {
178     grep -vE -e "^(|/)node_modules/" \
179         -e "^(|/)package-lock\.json$"
180 }
181
182 not_JS_dependencies_grep() {
183     grep -vE -e "^([:]+/) ?node_modules/" \
184         -e "^([:]+/) ?package-lock\.json:"
185 }
186
187 not_dependencies_find() {
188     not_JS_dependencies_find
189 }
190
191 not_dependencies_grep() {
192     not_JS_dependencies_grep
193 }
194
195 not_python_cache_find() {
196     grep -vE -e "^(|/)__pycache__/" \
197         -e "^(|/)\\.mypy_cache/" \
198         -e "^(|/)\\.ruff_cache/"
199 }
200
201 not_python_cache_grep() {
202     grep -vE -e "^([:]+/) ?__pycache__/" \
203         -e "^([:]+/) ?\\.mypy_cache/" \
204         -e "^([:]+/) ?\\.ruff_cache/"
205 }
206
207 not_cache_find() {
208     not_python_cache_find
209 }
210
211 not_cache_grep() {
212     not_python_cache_grep
213 }
214
215 not_git_find() {

```

```

216     grep -vE "^(^|/)\\.git/"
217 }
218
219 not_git_grep() {
220     grep -vE "^(^:]+/) ?\\.git/"
221 }
222
223 not_archive_find() {
224     grep -vE "(\\.gz|\\.rar|\\.tar|\\.tgz|\\.whl)$"
225 }
226
227 not_archive_grep() {
228     grep -vE "^[^:]*(.gz|\\.rar|\\.tar|\\.tgz|\\.whl):"
229 }
230
231 not_license_find() {
232     grep -vE -e "^(^|/)COPYING$" \
233             -e "^(^|/)COPYING\\.LESSER$"
234 }
235
236 not_license_grep() {
237     grep -vE -e "^(^:]+/) ?COPYING:" \
238             -e "^(^:]+/) ?COPYING\\.LESSER:"
239 }
240
241 not_main_tex_find() {
242     grep -vE "^(^|/)\$\{repository_name\}\\.tex$"
243 }
244
245 not_main_tex_grep() {
246     grep -vE "^(^:]+/) ?\$\\{repository_name\\}\\.tex:"
247 }
248
249 not_main_html_find() {
250     grep -vE "^(^|/)\$\{repository_name\}\\.html$"
251 }
252
253 not_main_html_grep() {
254     grep -vE "^(^:]+/) ?\$\\{repository_name\\}\\.html:"
255 }
256
257 not_temp_file_find() {
258     grep -vE "^(^|/)build_and_checks_variables/temp/.*$"
259 }
260
261 not_temp_file_grep() {

```

```

262     grep -vE "^([:]+/) ?build_and_checks_variables/temp/[[:]*:"
263 }
264
265 relevant_find() {
266     if [[ ! -o pipefail ]]; then
267         set -o pipefail
268         trap 'set +o pipefail' RETURN
269     fi
270     not_dependencies_find\
271         | not_cache_find\
272         | not_git_find\
273         | not_archive_find\
274         | not_temp_file_find
275 }
276
277 relevant_grep() {
278     if [[ ! -o pipefail ]]; then
279         set -o pipefail
280         trap 'set +o pipefail' RETURN
281     fi
282     not_dependencies_grep\
283         | not_cache_grep\
284         | not_git_grep\
285         | not_archive_grep\
286         | not_temp_file_grep
287 }
```

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\x{80}\x{8a}/ /g' \
50           -e 's/\xe2\x{80}\x{8b}/ /g' \
51           -e 's/\xe2\x{80}\x{8c}/ /g' \
52           -e 's/\xe2\x{80}\x{8d}/ /g' \
53           -e 's/\xe2\x{80}\x{8e}/ /g' \
54           -e 's/\xe2\x{80}\x{8f}/ /g' \
55           -e 's/\xe2\x{80}\x{aa}/ /g' \
56           -e 's/\xe2\x{80}\x{ab}/ /g' \
57           -e 's/\xe2\x{80}\x{ac}/ /g' \
58           -e 's/\xe2\x{80}\x{ad}/ /g' \
59           -e 's/\xe2\x{80}\x{ae}/ /g' \
60           -e 's/\xe2\x{80}\x{af}/ /g' \
61           -e 's/\xe2\x{81}\x{9f}/ /g' \

```

```

64      -e 's/\xe2\x81\x{aa}/ /g' \
65      -e 's/\xe2\x81\x{ab}/ /g' \
66      -e 's/\xe2\x81\x{ac}/ /g' \
67      -e 's/\xe2\x81\x{ad}/ /g' \
68      -e 's/\xe2\x81\x{ae}/ /g' \
69      -e 's/\xe2\x81\x{af}/ /g' \
70      -e 's/\xe2\x81\x{aa}/ /g' \
71      -e 's/\xe2\x81\x{ab}/ /g' \
72      -e 's/\xe2\x81\x{ac}/ /g' \
73      -e 's/\xe2\x81\x{ad}/ /g' \
74      -e 's/\xe2\x81\x{aa}/ /g' \
75      -e 's/\xe2\x81\x{ab}/ /g' \
76      -e 's/\xe2\x81\x{ac}/ /g' \
77      -e 's/\xe2\x81\x{ad}/ /g' \
78      -e 's/\xe2\x81\x{ae}/ /g' \
79      -e 's/\xe2\x81\x{af}/ /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=$?
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(?!\\s*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 "${LFBFL_file_name}" ] ] || continue
46      sed -Ez 's/(\nfrom _collections_abc[^\\n]*)/\\1\\n/Mg' \
47          "${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"
36
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=$1
153     split_line_at_most_result_end=""
154     return
155 fi
156 declare -r LFBFL_sort_command="sort --numeric-sort"
157 declare -ir LFBFL_length_minus_1=$(( ${#1} - 1 ))
158 declare -ir LFBFL_i_max=$(( ${#1} - 1 ))
159 declare -r LFBFL_min="$(
160     min "${LFBFL_sort_command}" "${LFBFL_length_minus_1}" "$2"
161 )
162 declare -i LFBFL_i
163 declare -i LFBFL_j
164 local LFBFL_previous_char
165 local LFBFL_current_char
166
167 # At the beginning, each position has score 0.
168 for ((LFBFL_i=0; LFBFL_i<=LFBFL_i_max; ++LFBFL_i)) do
169     LFBFL_positions["${LFBFL_i}"]="0"
170 done
171 LFBFL_j=$((LFBFL_i+1))
172 LFBFL_positions["${LFBFL_j}"]="0"
173
174 # If we have a forbidden previous character,
175 # we mark the forbidden positions with -1.
176 if [[ -n "$5" ]]; then
177     LFBFL_previous_char=""
178     for ((LFBFL_i=0; LFBFL_i<LFBFL_i_max; ++LFBFL_i)) do
179         LFBFL_j=$((LFBFL_i+1))
180         if [[ LFBFL_i -ge 1 ]]; then
181             LFBFL_previous_char="${LFBFL_current_char}"
182         fi
183         LFBFL_current_char="${1:${LFBFL_i}:1}"

```

```

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

```

```

229     # echo "${split_score_result}|${LFBFL_i}"
230     if [[ ${LFBFL_positions["${LFBFL_i}"]} != "-1" ]]; then
231         LFBFL_positions["${LFBFL_i}"]=$(((
232             max "${LFBFL_sort_command}" \
233                 "${LFBFL_positions["${LFBFL_i}"]}") \
234             "${split_score_result}")
235         )
236     fi
237 fi
238 eval "$3'${LFBFL_current_char}' ${LFBFL_j} 1"
239 # echo "${split_score_result}|${LFBFL_j}"
240 if [[ split_score_result -ge 1 ]]; then
241     # echo "${split_score_result}|${LFBFL_j}"
242     if [[ ${LFBFL_positions["${LFBFL_j}"]} != "-1" ]]; then
243         LFBFL_positions["${LFBFL_j}"]=$(((
244             max "${LFBFL_sort_command}" \
245                 "${LFBFL_positions["${LFBFL_j}"]}") \
246             "${split_score_result}")
247         )
248     fi
249     fi
250 done
251 fi
252 declare -ir LFBFL_max_score=$(((
253     max 'sort --numeric-sort' "${LFBFL_positions[@]}"))
254 )
255 local LFBFL_value
256 declare -i LFBFL_best_position
257 for ((LFBFL_i=0; LFBFL_i<=LFBFL_i_max; ++LFBFL_i)) do
258     local LFBFL_value=${LFBFL_positions[$LFBFL_i]}
259     # echo "${LFBFL_i} ${LFBFL_value}"
260     if [[ "${LFBFL_value}" == "${LFBFL_max_score}" ]]; then
261         LFBFL_best_position=$((LFBFL_i))
262     fi
263 done
264 # shellcheck disable=SC2248
265 split_line_at "$1" ${LFBFL_best_position}
266 split_line_at_most_result_start="${split_line_at_result_beginning}"
267 split_line_at_most_result_end="${split_line_at_result_end}"
268 # echo "$split_line_at_most_result_start"
269 # echo "$split_line_at_most_result_end"
270 }
271
272 split_last_line() {
273     # $1=$new_lines
274     # $2=$prefix

```

```

275  # $3=$max_length
276  # $4=$suffix
277  # $5=$split_score_command
278  # $6=$split_score_command_properties
279  # $7=$forbidden_previous_character ('\\' usually)
280  # In general, a prefix/suffix can be a required line prefix/suffix
281  # for all final lines like a comment prefix,
282  # or it can be a continuation line prefix/suffix applied only
283  # when splitting a line in 2 lines,
284  # or it can be a prefix/suffix for the string made of all the lines.
285  # Clearly, there is no gain to couple handling a prefix/suffix for
286  # the string made of all the lines in this function.
287  # Since we add a prefix/suffix when splitting,
288  # it is easy to see that we do not need to distinguish between
289  # required line prefix/suffix and continuation line prefix/suffix.
290  # The $2 and $4 arguments are given as concatenation of the
291  # required and continuation line prefixes/suffixes as needed by the
292  # user.
293  # Thus lines arguments should be given already prefixed
294  # and suffixed.
295  # This is logic, since it would be cumbersome to handle prefixing
296  # or not already splitted lines in this function
297  # (see repeated_split_last_line).
298  # The given max_length is the goal.
299  # This function adapts the effective max position for the split by
300  # taking into account the length of the suffix.
301 if [[ ! -o pipefail ]]; then
302     set -o pipefail
303     trap 'set +o pipefail' RETURN
304 fi
305
306 declare -g split_last_line_result="$1"
307 declare -ir LFBFL_max_length_plus=$((3 + 1))
308 declare -ir LFBFL_length2=$((3 - ${#4}))
309 declare -r LFBFL-regexp='.\{'"${LFBFL_max_length_plus}"'\}$
310 if
311     echo "$1" \
312         | sed -e 's/\n/g' \
313         | grep -q "${LFBFL-regexp}";
314 then
315     declare -r LFBFL_start=$((
316         echo "$1" \
317             | sed -e 's/\n/g' \
318             | head --lines=-1 \
319             | sed -z 's/\n/g'
320     ))

```

```

321 # echo "start: $LFBFL_start"
322 declare -r LFBFL_last_line=$(
323     echo "$1" \
324     | sed -e 's/\n/\n/g' \
325     | tail --lines=1
326 )
327 # echo "last_line: ${LFBFL_last_line}"
328 split_last_line_result=""
329 if [[ -n "${LFBFL_start}" ]]; then
330     split_last_line_result="${LFBFL_start}\n"
331 fi
332 if [[ -n "$5" ]]; then
333     split_line_at_most "${LFBFL_last_line}" "${LFBFL_length2}" \
334         "$5" "$6" "$7"
335     split_last_line_result+="${split_line_at_most_result_start}$4"
336     split_last_line_result+="\n"
337     split_last_line_result+="$2${split_line_at_most_result_end}"
338 else
339     split_last_line_result+="${LFBFL_last_line}:0:${LFBFL_length2} }"
340     split_last_line_result+="\$4\n"
341     split_last_line_result+="$2${LFBFL_last_line}: ${LFBFL_length2} }"
342 fi
343 fi
344 }
345
346 repeated_split_last_line() {
347     declare -g repeated_split_last_line_result="$1"
348     local LFBFL_i
349     for ((LFBFL_i=0; LFBFL_i<$6; ++LFBFL_i)) do
350         split_last_line "${repeated_split_last_line_result}" "$2" "$3" \
351             "$4" "$5" "$6" "$7"
352         repeated_split_last_line_result="${split_last_line_result}"
353         # echo "${split_last_line_result}"
354     done
355 }

```

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     declare -i LFBFL_verbose=0
35     if [[ "$*" == *--verbose* ]]; then
36         echo "$0 $*"
37         LFBFL_verbose=1
38     fi
39     readonly LFBFL_verbose
40
41     if [[ ! -o pipefail ]]; then
42         [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
43         set -o pipefail
44         trap 'set +o pipefail' RETURN
45     fi
46
47     get_COMMON_TEXT_FILES_GLOB_PATTERNS
48     local LFBFL_pattern
49     local LFBFL_long_line
50     local LFBFL_file_name
51     local LFBFL_line
52     local LFBFL_extension
53     local LFBFL_base_name
54     for LFBFL_pattern in "${COMMON_TEXT_FILES_GLOB_PATTERNS[@]}"; do

```

```

55  [ [ LFBFL_verbose -eq 0 ] ] \
56    || echo "Iterating on pattern: ${LFBFL_pattern}""
57  find . -type f -name "${LFBFL_pattern}" -printf '%P\n' \
58    | relevant_find \
59    | not_license_find \
60    | xargs grep -H '.\{71\}' \
61    | while read -r LFBFL_long_line;
62 do
63   LFBFL_file_name=${LFBFL_long_line%%:*} # Drop long ':' suffix
64   LFBFL_line=${LFBFL_long_line#*:} # Drop short '*' prefix
65   LFBFL_line="${LFBFL_line%\\/}" # Drop/Replace '\' suffix by ''
66   LFBFL_extension=${LFBFL_file_name##*.} # Drop long '.' prefix
67   LFBFL_base_name=${LFBFL_file_name%.*} # Drop short '..' suffix
68   if [[ "${LFBFL_extension}" == "html" ]]; then
69     if [[ -f "${LFBFL_base_name}.md" ]]; then
70       continue
71     fi
72     if grep --quiet --fixed-strings "${LFBFL_line}" \
73       "./build_and_checks_variables/temp/files_listing.html.tpl"
74     then
75       continue
76     fi
77     echo "${LFBFL_long_line}"
78   elif [[ "${LFBFL_extension}" == "md" ]]; then
79     if ! [[ -f "${LFBFL_base_name}.md.tpl" ]]; then
80       echo "${LFBFL_long_line}"
81     fi
82   else
83     echo "${LFBFL_long_line}"
84   fi
85 done
86 done
87 }

```

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 "./${LFBFL_file_name}" \
81         | cut -f1 -d' '
82     )
83     LFBFL_file_sha512_all="local LFBFL_correct_sha512\n"
84     LFBFL_file_sha512_all+="    LFBFL_correct_sha512=\""
85     LFBFL_file_sha512_all+="${LFBFL_file_sha512:0:45}'\n"
86     LFBFL_file_sha512_all+="    LFBFL_correct_sha512\+="
87     LFBFL_file_sha512_all+="${LFBFL_file_sha512:45:44}'\n"
88     LFBFL_file_sha512_all+="    LFBFL_correct_sha512\+="
89     LFBFL_file_sha512_all+="${LFBFL_file_sha512:89}'"
90     LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
91     sed -i \
92         "s|@sha512_${LFBFL_base_file_name}@|${LFBFL_file_sha512_all}|g" \
93         "./${LFBFL_common_file_name}.temp"
94 done
95
96 overwrite_if_not_equal "./${LFBFL_common_file_name}" \
97     "./${LFBFL_common_file_name}.temp"
98
99 declare -r LFBFL_main_file_name="build_and_checks.exec.sh"
100 cp "./${LFBFL_main_file_name}.tpl" "../${LFBFL_main_file_name}.temp"
101
102 LFBFL_file_sha512=$(

```

```

103     sha512sum "./${LFBFL_common_file_name}" \
104     | cut -f1 -d' '
105 )
106 LFBFL_file_sha512_all="local LFBFL_correct_sha512\n"
107 LFBFL_file_sha512_all+=" LFBFL_correct_sha512="
108 LFBFL_file_sha512_all+="${LFBFL_file_sha512:0:45}'\n"
109 LFBFL_file_sha512_all+=" LFBFL_correct_sha512\+="
110 LFBFL_file_sha512_all+="${LFBFL_file_sha512:45:44}'\n"
111 LFBFL_file_sha512_all+=" LFBFL_correct_sha512\+="
112 LFBFL_file_sha512_all+="${LFBFL_file_sha512:89}'"
113 sed -i \
114     "s|@sha512_${LFBFL_common_file_name}@|${LFBFL_file_sha512_all}|g" \
115     "../${LFBFL_main_file_name}.temp"
116
117 overwrite_if_not_equal "../${LFBFL_main_file_name}" \
118     "../${LFBFL_main_file_name}.temp"
119 }
120
121 update_common_build_and_checks "$@"

```

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
70     if [[ LFBFL_write -eq 1 ]]; then

```

```

71      : > "${LFBFL_listing}"
72  fi
73 # Remove line returns here to keep lines short.
74 sed -Ez 's/\\n//Mg' "${LFBFL_listing}" \
75   > "${LFBFL_listing}.temp"
76 shopt -s dotglob
77 get_split_score_simple 1 70 /
78 declare -r LFBFL_split_score_command="${get_split_score_result}"
79 local LFBFL_split_score_command_properties
80 LFBFL_split_score_command_properties="${get_split_score_result2}"
81 readonly LFBFL_split_score_command_properties
82 local LFBFL_suffix=\\ \\ # instead of '\\\' to avoid shellcheck SC1003
83 readonly LFBFL_suffix
84 local LFBFL_file_name
85 local LFBFL_base_file_name
86 local LFBFL_split_file_name
87 find . -type f -printf '%P\n' \
88   | relevant_find\
89   | sort\
90   | while read -r LFBFL_file_name;
91 do
92     git check-ignore -q "${LFBFL_file_name}" && continue
93     LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
94     [[ "${LFBFL_base_file_name}" != files_names_listing.txt* ]] \
95       || continue
96     if [[ "${LFBFL_base_file_name}" == *.md ]]; then
97       if [[ -f "${LFBFL_file_name}.tpl" ]]; then
98         continue
99       fi
100    fi
101   repeated_split_last_line "${LFBFL_file_name}" "" 70 \
102     "${LFBFL_suffix}" "${LFBFL_split_score_command}" \
103     "${LFBFL_split_score_command_properties}"
104   LFBFL_split_file_name="${repeated_split_last_line_result}"
105   if [[ LFBFL_write -eq 1 ]]; then
106     # shellcheck disable=SC2001
107     echo "${LFBFL_file_name}" \
108       | sed -e "s|${LFBFL_file_name}|${LFBFL_split_file_name}|g" \
109       >> "${LFBFL_listing}"
110     continue
111   fi
112   if grep -q "${LFBFL_file_name}\$" "${LFBFL_listing}.temp"; then
113     :
114   else
115     echo\
116       "The file ${LFBFL_file_name} is not listed in ${LFBFL_listing}.""

```

```

117     if [[ LFBFL_append -eq 1 ]]; then
118         # shellcheck disable=SC2001
119         echo "${LFBFL_file_name}" \
120             | sed -e "s|${LFBFL_file_name}|${LFBFL_split_file_name}|g" \
121             >> "${LFBFL_listing}"
122     fi
123     fi
124 done
125
126 while read -r LFBFL_file_name;
127 do
128     [[ "${LFBFL_file_name}" != '/*' ]] || continue
129     LFBFL_base_file_name=$(basename "${LFBFL_file_name}")
130     if ! [[ -f "${LFBFL_file_name}" ]]; then
131         echo \
132             "The non-file ${LFBFL_file_name} is listed in ${LFBFL_listing}."
133     fi
134 done < "${LFBFL_listing}.temp"
135
136 rm "${LFBFL_listing}.temp"
137 shopt -u dotglob
138 shopt -s globstar
139
140 if [[ LFBFL_verbose -eq 1 ]]; then
141     cat "${LFBFL_listing}"
142 fi
143 }
144 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/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 GPLv3+.
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="https://github.com/LLyaudet/@repository_name@/"><span
164   style="font-family:monospace"
165 >https://github.com/LLyaudet/@repository_name@/</span></a>.
166 </p>
167
168 <div>
169 <p>
170 Here is a listing of the dependencies of the code in this repository:
171 </p>
172 <ul>
173 <li> The following dependencies are GPLv3+ licensed:<br>
174 basename, bash (and its builtins), cat, chmod, cp, cut, date, diff,
175 dirname, find, grep, head, mkdir, mv, realpath, rm, sed, sha512sum,
176 shellcheck, sort, tail, wc, wget, xargs
177 <li> The following dependencies are GPLv2+ licensed:<br>
178 more, pandoc, pdftex, pdflatex, tree
179 <li> The following dependencies are GPLv2 licensed:<br>
180 git, pylint
181 <li> The following dependencies are LGPLv2.1+ licensed:<br>
182 notify-send

```

```

183 <cli> The following dependencies are MIT licensed:<br>
184 black, isort, mypy, ruff
185 <cli> The following dependencies are Apache-2.0 licensed:<br>
186 bandit
187 <cli> The following dependencies are BSD-3-Clause licensed:<br>
188 phpmd
189 <cli> php is licensed under PHP License v3.01
190 <cli> tex is licensed under Knuth license
191 <cli> pcregrep has a custom license from Cambridge University
192 </ul>
193 </div>
194
195 <p>
196 Remark: Some commands listed under GPLv3+ are POSIX
197 and can have different implementations under another license;
198 for example, uutils coreutils is MIT licensed,
199 whilst GNU coreutils is GPLv3+ licensed.
200 </p>
201
202 @files_listing_HTMLPreInput@
203
204 <p>
205 @acknowledgments@
206 </p>
207
208 <script>
209 "use strict";
210 const pres = Array.from(
211   document.getElementsByClassName('numbered_lines')
212 );
213 pres.forEach(
214   (pre) => {
215     const nb = Array.from(pre.innerText.matchAll("\n")).length + 1;
216     let numbers = "";
217     for(let i = 1; i <= nb; ++i){
218       numbers += i.toString();
219       if(i < nb){ numbers += "\n ";}
220     }
221     pre.setAttribute('line-numbers', numbers);
222   }
223 );
224 </script>
225 </html>

```

2.33 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{latexsym}
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{https://github.com/LLyaudet/@repository_name@/}.
109
110 Here is a listing of the dependencies of the code in this repository:
111 \begin{itemize}
112 \item The following dependencies are GPLv3+ licensed:\newline
113 basename, bash (and its builtins), cat, chmod, cp, cut, date, diff,
114 dirname, find, grep, head, mkdir, mv, realpath, rm, sed, sha512sum,
115 shellcheck, sort, tail, wc, wget, xargs
116 \item The following dependencies are GPLv2+ licensed:\newline
117 more, pandoc, pdftex, pdflatex, tree
118 \item The following dependencies are GPLv2 licensed:\newline
119 git, pylint
120 \item The following dependencies are LGPLv2.1+ licensed:\newline
121 notify-send
122 \item The following dependencies are MIT licensed:\newline
123 black, isort, mypy, ruff
124 \item The following dependencies are Apache-2.0 licensed:\newline
125 bandit
126 \item The following dependencies are BSD-3-Clause licensed:\newline
127 phpmd
128 \item php is licensed under PHP License v3.01
129 \item tex is licensed under Knuth license
130 \item pcregrep has a custom license from Cambridge University
131 \end{itemize}
132
133 Remark: Some commands listed under GPLv3+ are POSIX
134 and can have different implementations under another license;
135 for example, uutils coreutils is MIT licensed,
136 whilst GNU coreutils is GPLv3+ licensed.

```

```

137
138 @files_listing_VerbatimInput@
139
140 @acknowledgments@
141
142 \end{document}

```

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

```

```

176 license_file_header_AGPLv3+.tpl
177 }
178
179 \subsection{
180     build_and_checks_dependencies/licenses/_templates/%
181     license_file_header_GPLv3+.tpl
182 }
183 \label{
184     build_and_checks_dependencies:licenses_templates:%
185     license_file_header_GPLv3+tpl
186 }
187
188 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
189 build_and_checks_dependencies/licenses_templates/%
190     license_file_header_GPLv3+.tpl
191 }
192
193
194 \subsection{
195     build_and_checks_dependencies/licenses/_templates/%
196     license_file_header_LGPLv3+.tpl
197 }
198 \label{
199     build_and_checks_dependencies:licenses_templates:%
200     license_file_header_LGPLv3+tpl
201 }
202
203 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
204 build_and_checks_dependencies/licenses_templates/%
205     license_file_header_LGPLv3+.tpl
206 }
207
208
209 \subsection{
210     build_and_checks_dependencies/lines/_counts.libr.sh
211 }
212 \label{
213     build_and_checks_dependencies:lines_countslibrsh
214 }
215
216 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
217 build_and_checks_dependencies/lines_counts.libr.sh
218 }
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
277 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
278 build_and_checks_dependencies/python_isort_complement.libr.sh
279 }

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 \subsection{
318     build_and_checks_dependencies/too_long_code_lines.libr.sh
319 }
320 \label{
321     build_and_checks_dependencies:too_long_code_lineslibrsh
322 }
323
324
325 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
326 build_and_checks_dependencies/too_long_code_lines.libr.sh
327 }
328
329 \subsection{
330     build_and_checks_dependencies/%
331 update_common_build_and_checks.exec.sh
332 }
333 \label{
334     build_and_checks_dependencies:update_common_build_and_checksexecsh
335 }
336
337 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
338 build_and_checks_dependencies/update_common_build_and_checks.exec.sh
339 }
340
341
342 \subsection{
343     build_and_checks_dependencies/%
344 update_or_check_files_names_listing.exec.sh
345 }
346 \label{
347     build_and_checks_dependencies:%
348 update_or_check_files_names_listingexecsh
349 }
350
351 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
352 build_and_checks_dependencies/%
353 update_or_check_files_names_listing.exec.sh
354 }
355
356
357 \subsection{
358     build_and_checks_variables/.gitignore
359 }
```

```

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/DevOrSysAdminScripts.html.tpl
396 }
397 \label{
398   build_and_checks_variables:DevOrSysAdminScriptsshtmltpl
399 }
400
401 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
402 build_and_checks_variables/DevOrSysAdminScripts.html.tpl
403 }
404
405

```

```

406 \subsection{
407     build\_and\_checks\_variables/DevOrSysAdminScripts.tex.tpl
408 }
409 \label{
410     build_and_checks_variables:DevOrSysAdminScriptstextpl
411 }
412
413 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
414 build_and_checks_variables/DevOrSysAdminScripts.tex.tpl
415 }
416
417
418 \subsection{
419     build\_and\_checks\_variables/files\_listing.tex.tpl
420 }
421 \label{
422     build_and_checks_variables:files_listingtextpl
423 }
424
425 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
426 build_and_checks_variables/files_listing.tex.tpl
427 }
428
429
430 \subsection{
431     build\_and\_checks\_variables/files\_names\_listing.txt
432 }
433 \label{
434     build_and_checks_variables:files_names_listingtxt
435 }
436
437 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
438 build_and_checks_variables/files_names_listing.txt
439 }
440
441
442 \subsection{
443     build\_and\_checks\_variables/licenses\_notes.txt
444 }
445 \label{
446     build_and_checks_variables:licenses_notestxt
447 }
448
449 \VerbatimInput[numbers=left,xleftmargin=-5mm]{
450 build_and_checks_variables/licenses_notes.txt
451 }

```

```

452
453
454 \subsection{
455     build\_and\_checks\_variables/phpmd\_baseline.xml
456 }
457 \label{
458     build_and_checks_variables:phpmd_baselinexml
459 }
460
461 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
462 build_and_checks_variables/phpmd_baseline.xml
463 }
464
465
466 \subsection{
467     build\_and\_checks\_variables/pylintrc.toml
468 }
469 \label{
470     build_and_checks_variables:pylintrctoml
471 }
472 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
473 build_and_checks_variables/pylintrc.toml
474 }
475
476
477 \subsection{
478     build\_and\_checks\_variables/repository\_data.txt
479 }
480 \label{
481     build_and_checks_variables:repository_datatxt
482 }
483
484 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
485 build_and_checks_variables/repository_data.txt
486 }
487
488
489 \subsection{
490     build\_and\_checks\_variables/ruff.toml
491 }
492 \label{
493     build_and_checks_variables:rufftoml
494 }
495
496 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
497

```

```

498 build_and_checks_variables/ruff.toml
499 }
500
501
502 \subsection{
503     build\_and\_checks\_variables/shellcheck.ini
504 }
505 \label{
506     build_and_checks_variables:shellcheckini
507 }
508
509 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
510 build_and_checks_variables/shellcheck.ini
511 }
512
513
514 \subsection{
515     build\_and\_checks\_variables/shellcheck\_notes.txt
516 }
517 \label{
518     build_and_checks_variables:shellcheck_notestxt
519 }
520
521 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
522 build_and_checks_variables/shellcheck_notes.txt
523 }
524
525
526 \subsection{
527     build\_and\_checks.exec.sh
528 }
529 \label{
530     build_and_checksexecsh
531 }
532
533 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
534 build_and_checks.exec.sh
535 }
536
537
538 \subsection{
539     COPYING
540 }
541 \label{
542     COPYING
543 }

```

```

544 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
545 COPYING
546 }
548
549 \subsection{
550     COPYING.LESSER
551 }
553 \label{
554     COPYINGLESSER
555 }
556
557 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
558 COPYING.LESSER
559 }
560
561 \subsection{
562     git\_helpers/git\_checks.libr.sh
563 }
565 \label{
566     git_helpers:git_checkslibrsh
567 }
568
569 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
570 git_helpers/git_checks.libr.sh
571 }
572
573
574 \subsection{
575     git\_helpers/post-commit
576 }
577 \label{
578     git_helpers:post-commit
579 }
580
581 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
582 git_helpers/post-commit
583 }
584
585
586 \subsection{
587     git\_helpers/post\_commit\_\_renaming\_\_comment.libr.sh
588 }
589 \label{

```

```

590     git_helpers:post_commit_renaming_commentlibrsh
591 }
592
593 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
594 git_helpers/post_commit_renaming_comment.libr.sh
595 }
596
597 \subsection{
598     git\_\_helpers/pre-commit
599 }
600 \label{
601     git_helpers:pre-commit
602 }
603
604 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
605 git_helpers/pre-commit
606 }
607
608
609 \subsection{
610     LiensUtiles.txt
611 }
612 \label{
613     LiensUtilestxt
614 }
615
616 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
617 LiensUtiles.txt
618 }
619
620
621 \subsection{
622     README.md.tpl
623 }
624 \label{
625     READMEmdtpl
626 }
627
628 \VerbatimInput[numbers=left,xleftmargin=-5mm] {
629 README.md.tpl
630 }
631
632
633 \subsection{
634     unicode.libr.php
635 }
```

```

636 }
637 \label{
638     unicodelibrphp
639 }
640
641 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
642     unicode.libr.php
643 }
644
645
646 \subsection{
647     upgrade\_build\_and\_checks.exec.sh
648 }
649 \label{
650     upgrade_build_and_checksexecsh
651 }
652
653 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
654     upgrade_build_and_checks.exec.sh
655 }
656
657
658 \subsection{
659     wget\_sha512.libr.sh
660 }
661 \label{
662     wget_sha512librsh
663 }
664
665 \VerbatimInput [numbers=left,xleftmargin=-5mm] {
666     wget_sha512.libr.sh
667 }
668
669

```

2.35 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 LGPLv3+.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/DevOrSysAdminScripts.html.tpl
69 // build_and_checks_variables/DevOrSysAdminScripts.tex
70 build_and_checks_variables/DevOrSysAdminScripts.tex.tpl
71 build_and_checks_variables/files_listing.tex.tpl
72 build_and_checks_variables/files_names_listing.txt
73 build_and_checks_variables/licenses_notes.txt
74 build_and_checks_variables/phpmd_baseline.xml
75 build_and_checks_variables/pylintrc.toml
76 build_and_checks_variables/repository_data.txt
77 build_and_checks_variables/ruff.toml
78 build_and_checks_variables/shellcheck.ini
79 build_and_checks_variables/shellcheck_notes.txt
80 // build_and_checks_variables/upgrade_venvs_ts
81 build_and_checks.exec.sh
82 COPYING
83 COPYING.LESSER
84 // DevOrSysAdminScripts.html
85 // DevOrSysAdminScripts.pdf
86 // EncoreBlasé20240517-20240518.txt
87 git_helpers/git_checks.libr.sh
88 git_helpers/post-commit
89 git_helpers/post_commit_renaming_comment.libr.sh
90 git_helpers/pre-commit
91 LiensUtiles.txt
92 // README.html
93 // README.md
94 README.md.tpl
95 unicode.libr.php
96 upgrade_build_and_checks.exec.sh
97 wget_sha512.libr.sh

```

2.36 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 GPLv3+.
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 LGPLv2.1+ 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.
```

2.37 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_dependenc_
5       <violation rule="PHPMD\Rule\Controversial\CamelCaseVariableName" file="build_and_
6         <violation rule="PHPMD\Rule\Controversial\CamelCaseParameterName" file="unicode.i_
7           <violation rule="PHPMD\Rule\Naming\LongVariable" file="unicode.libr.php"/>
8           <violation rule="PHPMD\Rule\Controversial\CamelCaseVariableName" file="unicode.li_
9             <violation rule="PHPMD\Rule\CleanCode\MissingImport" file="unicode.libr.php"/>
10             <violation rule="PHPMD\Rule\CyclomaticComplexity" file="unicode.libr.php"/>
11             <violation rule="PHPMD\Rule\Design\NpathComplexity" file="unicode.libr.php"/>
12   </phpmd-baseline>
```

2.38 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 module-rgx = "^[_a-zA-Z0-9+]+$"
7 [tool pylint.reports]
8 output-format = "colorized"
```

2.39 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 grammar_or_spell_checker_command=more
26
27
28 Why one venv for each Python dependency ?
29 Because no language of my knowledge did it right to handle all
30 possible dependency conflicts.
31 So in case of such a conflict, one specialized venv can get you out
32 of any problem.
33 The best solution would have been to allow install of different
```

```

34 versions of the same package, and have a mechanism:
35 - first intent, use the latest available version of a package among
36 those that satisfy the declared dependency constraints of the package
37 that has this dependency;
38 - second intent, check a dependency overload configuration file
39 with tree structure: if calling python package is A version 1.1.1
40 with dependency B itself with dependency C, use B 2.2.2 and C 3.3.3;
41 you could have along that another rule saying that when called
42 directly B 2.2.2 should use C 4.4.4;
43 and even more complicated:
44 if calling python package is A version 1.1.1 with dependency B and D
45 both themselves with dependency C, use B 2.2.2, D 0.1.0,
46 and C 3.3.3 for B, and C 4.4.4 for D;
47 once you've seen that it can never be more complicated than that once
48 the same version number guarantees that the code is the same,
49 you create a config language "à la CSS" for cascading dependencies ;)
50 XD fit into JSON the dumbest possible way and you stop bothering
51 about dependency conflicts for the rest of history of informatics.
52 {
53     "A": {
54         "1.1.1": {
55             "B": {
56                 "2.2.2": {
57                     "C": {
58                         // Dumb, you did not found it by yourself? Failed exam.
59                         "3.3.3" : {}
60                     }
61                 }
62             },
63             "D": {
64                 "0.1.0": {
65                     "C": {
66                         // Dumb, you did not found it by yourself? Failed exam.
67                         "4.4.4" : {}
68                     }
69                 }
70             }
71         },
72         "B": {
73             "2.2.2": {
74                 "C": {
75                     // Dumb, you did not found it by yourself? Failed exam.
76                     "4.4.4" : {}
77                 }
78             }
79         }
}

```

```

80    }
81 Wonderful someone rediscovered nested structures XD
82 Seriously so many engineers or computer scientists are dumb to cry.
83
84 bandit_venv=../venv_bandit
85 black_venv=../venv_black
86 isort_venv=../venv_isort
87 mypy_venv=../venv_mypy
88 pylint_venv=../venv_pylint
89 ruff_venv=../venv_ruff
90
91
92 upgrade venvs can be auto or ask, any other value is interpreted
93 as ask.
94
95 upgrade_venvs=ask
96
97
98 time interval can be 0 to always upgrade or ask to upgrade,
99 or a number at most bash max int
100 https://superuser.com/questions/1030122/
101 what-is-the-maximum-value-of-a-numeric-bash-shell-variable
102 it should be enough to simulate never "upgrade or ask for upgrade"
103 unless you claim to live many billions years ;)
104 If it is wat, this is a random int XD (see RANDOM).
105 Note that the probability that it upgrades your dependencies is not
106 1/2, it increases over time, and is 1 after slightly
107 less than one day.
108 If it is watyouwant?, see RANDOM XD.
109
110 upgrade_venvs_time_interval_in_seconds=wat
111
112
113 For the Nu HTML, CSS, and SVG checker.
114
115 vnu_jar_path=../jar_venv/vnu.jar
116
117 Not fully implemented yet, it will only warn that vnu jar was not
118 updated since some time.
119 "wat" works, but "watyouwant?" joke was not repeated ;).
120
121 upgrade_vnu_jar_time_interval_in_seconds=604800

```

2.40 build_and_checks_variables/ruff.toml

```
1 [lint]
2 select = ["ALL", "D211"]
3 ignore = ["D203", "D212", "D404"]
4 [lint.per-file-ignores]
5 "build_and_checks_dependencies/licenses_templates/*.py" = ["INP001"]
```

2.41 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
```

2.42 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$"

```

2.43 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='4ac98c6888998ebf45fe166fcaecd5b104de8624abfec'
71   LFBFL_correct_sha512+='38ad0ba7442ef32c0e6b944a87344b6dfb5626895b44'

```

```

72 LFBFL_correct_sha512+='b6c5db1d098363c1ee33ba1556bdfb2ae214c89'
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="${LFBFL_variables_directory}/"
91     LFBFL_data_file_name+="repository_data.txt"
92     grep_variable "${LFBFL_data_file_name}" repository_name
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 "${LFBFL_variables_directory}/${repository_name}.tex"
97     touch "${LFBFL_working_directory}/${repository_name}.pdf"
98     touch "${LFBFL_working_directory}/${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 "$@"

```

2.44 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

```

Preamble

10 The GNU General Public License is a free, copyleft license for
11 software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

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

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.

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.

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.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of 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.

70

71 TERMS AND CONDITIONS

72

73 0. Definitions.

74

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

76

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

79

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

83

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

88

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

91

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

98

99 To "convey" a work means any kind of propagation that enables other

100 parties to make or receive copies. Mere interaction with a user through
101 a computer network, with no transfer of a copy, is not conveying.

102
103 An interactive user interface displays "Appropriate Legal Notices"
104 to the extent that it includes a convenient and prominently visible
105 feature that (1) displays an appropriate copyright notice, and (2)
106 tells the user that there is no warranty for the work (except to the
107 extent that warranties are provided), that licensees may convey the
108 work under this License, and how to view a copy of this License. If
109 the interface presents a list of user commands or options, such as a
110 menu, a prominent item in the list meets this criterion.

111
112 1. Source Code.

113
114 The "source code" for a work means the preferred form of the work
115 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.

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.

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.

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.

629 To do so, attach the following notices to the program. It is safest
630 to attach them to the start of each source file to most effectively
631 state the exclusion of warranty; and each file should have at least
632 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>

636
637 This program is free software: you can redistribute it and/or modify
638 it under the terms of the GNU General Public License as published by
639 the Free Software Foundation, either version 3 of the License, or
640 (at your option) any later version.

641
642 This program is distributed in the hope that it will be useful,
643 but WITHOUT ANY WARRANTY; without even the implied warranty of
644 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
645 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:

```
652     <program> Copyright (C) <year> <name of author>
653     This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
654     This is free software, and you are welcome to redistribute it
655     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/>.

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

2.45 COPYING.LESSER

21
22 An "Application" is any work that makes use of an interface provided
23 by the Library, but which is not otherwise based on the Library.
24 Defining a subclass of a class defined by the Library is deemed a mode
25 of using an interface provided by the Library.

26
27 A "Combined Work" is a work produced by combining or linking an
28 Application with the Library. The particular version of the Library
29 with which the Combined Work was made is also called the "Linked
30 Version".

31
32 The "Minimal Corresponding Source" for a Combined Work means the
33 Corresponding Source for the Combined Work, excluding any source code
34 for portions of the Combined Work that, considered in isolation, are
35 based on the Application, and not on the Linked Version.

36
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.
```

2.46 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 }

```

2.47 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
38 commit_a_file_renamed_comment --verbose --log-directory-change
39 exit 0

```

2.48 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      declare -i LFBFL_verbose=0
67      if [[ "$*" == *--verbose* ]]; then
68          echo "$0 commit_a_file_renamed_comment $*"
69          LFBFL_verbose=1
70      fi
71      readonly LFBFL_verbose
72
73      if [[ ! -o pipefail ]]; then
74          [[ LFBFL_verbose -eq 1 ]] && echo "pipefail option activated"
75          set -o pipefail
76          trap 'set +o pipefail' RETURN
77      fi
78
79      declare -i LFBFL_log_directory_change=0
80      if [[ "$*" == *--log-directory-change* ]]; then
81          LFBFL_log_directory_change=1
82      fi
83      readonly LFBFL_log_directory_change
84
85      declare -i LFBFL_max_comment_line_length=70
86      if [[ -n "$3" ]]; then
87          LFBFL_max_comment_line_length=$((3))
88      fi
89      readonly LFBFL_max_comment_line_length
90      if [[ LFBFL_verbose -eq 1 ]]; then
91          echo "LFBFL_log_directory_change: ${LFBFL_log_directory_change}"
92          echo "LFBFL_max_comment_line_length:" \
93              "${LFBFL_max_comment_line_length}"
94      fi
95      #-----
96
97      declare -r LFBFL_renaming_lines=$((
98          git log -p -1 \
99          | grep '^diff --git' -A 1
100     ))
101     if [[ LFBFL_verbose -eq 1 ]]; then
102         echo "LFBFL_renaming_lines: ${LFBFL_renaming_lines}"
103     fi
104
105    declare -a LFBFL_renaming_lines_array
106    mapfile -t LFBFL_renaming_lines_array <<< "${LFBFL_renaming_lines}"
107    readonly LFBFL_renaming_lines_array
108
109    declare -r LFBFL_timestamp=$(date --iso-8601="minute")

```

```

110 if [[ LFBFL_verbose -eq 1 ]]; then
111     echo "LFBFL_timestamp: ${LFBFL_timestamp}"
112 fi
113 get_split_score_simple 1 "${LFBFL_max_comment_line_length}" _
114 LFBFL_split_score_command1="${get_split_score_result}"
115 LFBFL_split_score_command_properties1="${get_split_score_result2}" /
116 get_split_score_simple 1 "${LFBFL_max_comment_line_length}" /
117 LFBFL_split_score_command2="${get_split_score_result}"
118 LFBFL_split_score_command_properties2="${get_split_score_result2}"
119 declare -i LFBFL_renaming_happened=0
120 declare -i LFBFL_i
121 declare -i LFBFL_j
122 # declare -r LFBFL_special_file_name="files_names_listing.txt"
123 declare -r LFBFL_sfn="files_names_listing.txt"
124 local LFBFL_diff_line
125 local LFBFL_similarity_line
126 local LFBFL_old_file_path
127 local LFBFL_old_file_name
128 local LFBFL_old_file_directory
129 local LFBFL_new_file_path
130 local LFBFL_new_file_name
131 local LFBFL_new_file_directory
132 local LFBFL_useful_file_name
133 local LFBFL_extension
134 local LFBFL_comment_prefix
135 local LFBFL_comment_prefix2
136 local LFBFL_old_file_name2
137 local LFBFL_old_file_directory2
138 local LFBFL_new_file_name2
139 local LFBFL_new_file_directory2
140 local LFBFL_new_comment
141 declare -i LFBFL_copyright_line_number
142 declare -r LFBFL_send_summary_2="ATTENTION: No line with copyright"
143 declare -i LFBFL_line_count
144 declare -i LFBFL_lines_after
145 local LFBFL_temp_file_path
146 for ((LFBFL_i=0; LFBFL_i<${#LFBFL_renaming_lines_array[@]};)); do
147     LFBFL_diff_line="${LFBFL_renaming_lines_array[$LFBFL_i]}"
148     LFBFL_i=$((LFBFL_i + 1))
149     LFBFL_similarity_line="${LFBFL_renaming_lines_array[$LFBFL_i]}"
150     LFBFL_i=$((LFBFL_i + 1))
151     # We need to increment 3 times since grep adds "==" lines between.
152     LFBFL_i=$((LFBFL_i + 1))
153     if [[ LFBFL_verbose -eq 1 ]]; then
154         echo "LFBFL_diff_line: ${LFBFL_diff_line}"
155         echo "LFBFL_similarity_line: ${LFBFL_similarity_line}"

```

```

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

```

```

202     echo "LFBFL_useful_file_name: ${LFBFL_useful_file_name}"
203     echo "LFBFL_extension: ${LFBFL_extension}"
204   fi
205 done
206 LFBFL_comment_prefix=""
207 if [[ "${LFBFL_extension}" == "sh" ]]; then
208   LFBFL_comment_prefix="# "
209 fi
210 if [[ "${LFBFL_extension}" == "sql" ]]; then
211   LFBFL_comment_prefix="-- "
212 fi
213 if [[ "${LFBFL_extension}" == "tex" ]]; then
214   LFBFL_comment_prefix="% "
215 fi
216 if [[ "${LFBFL_useful_file_name}" == "${LFBFL_sfn}" ]]; then
217   LFBFL_comment_prefix="// "
218 fi
219 if [[ LFBFL_verbose -eq 1 ]]; then
220   echo "LFBFL_comment_prefix: ${LFBFL_comment_prefix}"
221 fi
222 #-----
223 # Split if necessary old_file_name, new_file_name, etc. to have
224 # comment lines below max_comment_line_length.
225 LFBFL_comment_prefix2="${LFBFL_comment_prefix}''"
226 # ----
227 LFBFL_old_file_name2="${LFBFL_comment_prefix2}"
228 LFBFL_old_file_name2+="${LFBFL_old_file_name}"
229 repeated_split_last_line "${LFBFL_old_file_name2}" \
230   "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
231   "'"' "${LFBFL_split_score_command1}" \
232   "${LFBFL_split_score_command_properties1}"
233 LFBFL_old_file_name2="${repeated_split_last_line_result}"
234 # ----
235 LFBFL_old_file_directory2="${LFBFL_comment_prefix2}"
236 LFBFL_old_file_directory2+="${LFBFL_old_file_directory}"
237 repeated_split_last_line "${LFBFL_old_file_directory2}" \
238   "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
239   "'"' "${LFBFL_split_score_command2}" \
240   "${LFBFL_split_score_command_properties2}"
241 LFBFL_old_file_directory2="${repeated_split_last_line_result}"
242 # ----
243 LFBFL_new_file_name2="${LFBFL_comment_prefix2}"
244 LFBFL_new_file_name2+="${LFBFL_new_file_name}"
245 repeated_split_last_line "${LFBFL_new_file_name2}" \
246   "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
247   "'"' "${LFBFL_split_score_command1}" \

```

```

248     "${LFBFL_split_score_command_properties1}"
249 LFBFL_new_file_name2="${repeated_split_last_line_result}"
250 # -----
251 LFBFL_new_file_directory2="${LFBFL_comment_prefix2}"
252 LFBFL_new_file_directory2+="${LFBFL_new_file_directory}"
253 repeated_split_last_line "${LFBFL_new_file_directory2}" \
254     "${LFBFL_comment_prefix2}" "${LFBFL_max_comment_line_length}" \
255     '' "${LFBFL_split_score_command2}" \
256     "${LFBFL_split_score_command_properties2}"
257 LFBFL_new_file_directory2="${repeated_split_last_line_result}"
258 # Create new comments strings. -----
259 LFBFL_new_comment=""
260 if [[ "${LFBFL_new_file_name2}" != "${LFBFL_old_file_name2}" ]];
261 then
262     LFBFL_new_comment="${LFBFL_comment_prefix}"
263     LFBFL_new_comment+=" ${LFBFL_timestamp}"
264     LFBFL_new_comment+=": This file was renamed from\n"
265     LFBFL_new_comment+=" ${LFBFL_old_file_name2}''\n"
266     LFBFL_new_comment+=" ${LFBFL_comment_prefix}to\n"
267     LFBFL_new_comment+=" ${LFBFL_new_file_name2}''."
268 fi
269 if [[ LFBFL_log_directory_change -eq 1 ]] && [[ \
270     "${LFBFL_new_file_directory}" \
271     != "${LFBFL_old_file_directory}" \
272 ]]; then
273     LFBFL_new_comment+="\n${LFBFL_comment_prefix}"
274     LFBFL_new_comment+=" ${LFBFL_timestamp}"
275     LFBFL_new_comment+=": This file was moved from\n"
276     LFBFL_new_comment+=" ${LFBFL_old_file_directory2}''\n"
277     LFBFL_new_comment+=" ${LFBFL_comment_prefix}to\n"
278     LFBFL_new_comment+=" ${LFBFL_new_file_directory2}''."
279 fi
280 if [[ LFBFL_verbose -eq 1 ]]; then
281     echo "LFBFL_new_comment: ${LFBFL_new_comment}"
282 fi
283 # Find line with ©Copyright. -----
284 LFBFL_copyright_line_number=${(
285     grep -n '©Copyright' "${LFBFL_new_file_path}" \
286     | head --lines=1 \
287     | cut -d ':' -f 1
288 )
289 if [[ LFBFL_verbose -eq 1 ]]; then
290     echo "LFBFL_copyright_line_number:" \
291         "${LFBFL_copyright_line_number}"
292 fi
293 if [[ LFBFL_copyright_line_number -eq 0 ]]; then

```

```

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

```

2.49 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

```

2.50 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 -----

```

2.51 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/llyaudet/DevOrSysAdminScripts/badge
51
52 [CodeFactor-package-page]: https://www.codefactor.io/repository/github/llyaudet/DevOrSysAdminScripts
53
54 [Codacy-badge]: https://app.codacy.com/project/badge/Grade/3dd6ef85e8db4460991d1ebc404b68dd
55
56 [Codacy-package-page]: https://app.codacy.com/gh/LLyaudet/DevOrSysAdminScripts/dashboard?utm\_source=gh&utm\_medium=referral&utm\_content=&utm\_campaign=Badge\_grade
57
58 [GitHub-top-language-badge]: https://img.shields.io/github/
59
60
61
62
63

```

```

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

```

2.52 **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 = (int) (
44             (
45                 $i_code_point_in_decimal_notation
46                 - $i_last_byte_significant_bits
47             ) / (2**6)
48         );
49         $s_result = chr(
50             $i_first_byte_base_value
51             + $i_first_byte_significant_bits
52         );
53         $s_result .= chr(
54             $i_continuation_base_value

```

```

55         + $i_last_byte_significant_bits
56     );
57     return $s_result;
58 }
59 // 1110xxxx 10xxxxxxxx 10xxxxxxxx
60 if($i_code_point_in_decimal_notation < 256 * 256) {
61     $i_first_byte_base_value = 224;
62     $arr_arr_data_per_byte_reverse = [
63         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
64         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
65         ["i_bits" => 4, "i_base_value" => $i_first_byte_base_value],
66     ];
67     $i_rest = $i_code_point_in_decimal_notation;
68     $s_result = "";
69     foreach($arr_arr_data_per_byte_reverse as $arr_data) {
70         // _Pragma unroll
71         // J'aurais pu "roller" les 4 cas des ifs.
72         // Mais pour les perfs, il faudrait tout unroller vu le nombre
73         // d'itérations. Et j'aimerais bien comparer du code unrollé
74         // aux 2 niveaux à la main avec celui unrollé par le compilateur
75         // sur les 2 niveaux. Je préjuge/présuppose qu'il unroll
76         // l'intérieur d'abord (pour ne pas refaire le même boulot)
77         // puis l'extérieur et qu'il oublie de réoptimiser l'intérieur
78         // en fonction de l'extérieur unrollé. Est-ce que j'ai raison ?
79     $i_significant_bits = $i_rest % (2**$arr_data["i_bits"]);
80     $i_rest = (int) (
81         ($i_rest - $i_significant_bits) / (2**$arr_data["i_bits"])
82     );
83     $s_result = ( // .= or =
84         chr($i_significant_bits + $arr_data["i_base_value"])
85         . $s_result
86     );
87 }
88     return $s_result;
89 }
89 // 11110xxx 10xxxxxxxx 10xxxxxxxx 10xxxxxxxx
90 if($i_code_point_in_decimal_notation < 256 * 256 * (2**5)) {
91     $i_first_byte_base_value = 240;
92     $arr_arr_data_per_byte_reverse = [
93         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
94         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
95         ["i_bits" => 6, "i_base_value" => $i_continuation_base_value],
96         ["i_bits" => 3, "i_base_value" => $i_first_byte_base_value],
97     ];
98     $i_rest = $i_code_point_in_decimal_notation;
99     $s_result = "";
100 }
```

```

101    foreach($arr_arr_data_per_byte_reverse as $arr_data) {
102        // _Pragma unroll
103        $i_significant_bits = $i_rest % (2**$arr_data["i_bits"]);
104        $i_rest = (int) (
105            ($i_rest - $i_significant_bits) / (2**$arr_data["i_bits"])
106        );
107        $s_result = ( // .= or =
108            chr($i_significant_bits + $arr_data["i_base_value"]))
109            . $s_result
110        );
111    }
112    return $s_result;
113}
114 throw new Exception(
115     "UTF-8 avec jusqu'à 6 octets a été abandonné il y a longtemps."
116     ." Autre anecdote \"débile\" : MySQL a de base UTF-8 avec jusqu'à"
117     ." 3 octets au lieu de 4 :face-palm:."
118     ." Du coup, ça peut donner quelques \"surprises\"."
119     ." Sinon, y'a aussi ASCII dans MySQL."
120     ." Et comme ça joue sur la mémoire et les performances,"
121     ." pourquoi ils n'ont pas fait ASCII, UTF-8-2, UTF-8-3, UTF-8-4 ?"
122     ." :)"
123 );
124}
125
126
127
128 function hexa_code_point_to_UTF8(
129     $s_code_point_in_hexadecimal_notation
130 ) {
131     $s_code_point_in_hexadecimal_notation = strtoupper(
132         $s_code_point_in_hexadecimal_notation
133     );
134     $i_code_point_in_decimal_notation = 0;
135     $i_ord_0 = ord('0');
136     $i_ord_A = ord('A');
137     $i_digit_offset = -$i_ord_0;
138     $i_letter_offset = 10 - $i_ord_A;
139     for(
140         $i = 0, $i_max = strlen($s_code_point_in_hexadecimal_notation);
141         $i < $i_max;
142         ++$i
143     ) {
144         $s_hexa_digit = $s_code_point_in_hexadecimal_notation[$i];
145         $i_code_point_in_decimal_notation *= 16;
146         if(ctype_digit($s_hexa_digit)) {

```

```

147     $i_code_point_in_decimal_notation += (
148         ord($s_hexa_digit) + $i_digit_offset
149     );
150     continue;
151 }
152 if(ctype_xdigit($s_hexa_digit)){
153     $i_code_point_in_decimal_notation += (
154         ord($s_hexa_digit) + $i_letter_offset
155     );
156     continue;
157 }
158 throw new Exception("Not an hexadecimal digit.");
159 }
160 return decimal_code_point_to_UTF8(
161     $i_code_point_in_decimal_notation
162 );
163 }
164
165
166
167 function check_string_is_valid_ASCII($s_string){
168     for($i = 0, $i_max = strlen($s_string); $i < $i_max; ++$i){
169         if(ord($s_string[$i]) > 127){
170             throw new Exception(
171                 "Non-ASCII character found at octet position "
172                 . $i
173                 . ", value "
174                 . ord($s_string[$i])
175                 . "."
176             );
177         }
178     }
179     return true;
180 }
181
182
183
184 function check_file_is_valid_ASCII($s_file_name){
185     $s_string = file_get_contents($s_file_name);
186     if($s_string === false){
187         throw new Exception("File ".$s_file_name." not found.");
188     }
189     return check_string_is_valid_ASCII($s_string);
190 }
191
192

```

```

193
194 function check_string_is_valid_UTF8($s_string) {
195     $i_continuation_octet_needed = 0;
196     $i_character_start_position = 0;
197     for($i = 0, $i_max = strlen($s_string); $i < $i_max; ++$i){
198         $i_current_octet = ord($s_string[$i]);
199         if($i_continuation_octet_needed > 0){
200             if($i_current_octet < 128 || $i_current_octet >= 192){
201                 throw new Exception(
202                     "Non-UTF8 character found at start position "
203                     . $i_character_start_position
204                     . ", octet at position "
205                     . $i
206                     . " has value "
207                     . $i_current_octet
208                     . " which is not a continuation octet."
209                 );
210             }
211             --$i_continuation_octet_needed;
212             continue;
213         }
214
215         $i_character_start_position = $i;
216         if($i_current_octet < 128){ // 0xxxxxxx ASCII
217             continue;
218         }
219         if($i_current_octet >= 128 && $i_current_octet < 192){
220             throw new Exception(
221                 "Non-UTF8 character found at start position "
222                 . $i_character_start_position
223                 . ", octet at position "
224                 . $i
225                 . " has value "
226                 . $i_current_octet
227                 . " which is a continuation octet."
228             );
229         }
230         if($i_current_octet >= 192 && $i_current_octet < 224){
231             // 110xxxxx 10xxxxxx
232             $i_continuation_octet_needed = 1;
233             continue;
234         }
235         if($i_current_octet >= 224 && $i_current_octet < 240){
236             // 1110xxxx 10xxxxxx 10xxxxxx
237             $i_continuation_octet_needed = 2;
238             continue;

```

```

239     }
240     if($i_current_octet >= 240 && $i_current_octet < 248) {
241         // 11110xxx 10xxxxxx 10xxxxxx 10xxxxxx
242         $i_continuation_octet_needed = 3;
243         continue;
244     }
245     throw new Exception(
246         "Non-UTF8 character found at start position "
247         .$.i_character_start_position
248         .", octet at position "
249         .$.i
250         ." has value "
251         .$.i_current_octet
252         ." which is invalid."
253     );
254 }
255 return true;
256 }
257
258
259
260 function check_file_is_valid_UTF8($s_file_name) {
261     $s_string = file_get_contents($s_file_name);
262     if($s_string === false) {
263         throw new Exception("File ".$s_file_name." not found.");
264     }
265     return check_string_is_valid_UTF8($s_string);
266 }
267
268
269 /*
270 ?>
271 <?php
272 require_once("./unicode.libr.php");
273 var_dump(hexa_code_point_to_UTF8("002B"));
274 var_dump(hexa_code_point_to_UTF8("00E6"));
275 var_dump(hexa_code_point_to_UTF8("1400"));
276 var_dump(hexa_code_point_to_UTF8("10111"));
277 // https://en.wikibooks.org/wiki/Unicode/Character_reference/2000-2FFF
278 // Some invisible unicode characters that are a problem:
279 // 2000-200F 2028-202F 205F 2060-206F
280 // $arr = [
281     '2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007',
282     '2008', '2009', '200A', '200B', '200C', '200D', '200E', '200F',
283     '2028', '2029', '202A', '202B', '202C', '202D', '202E', '202F',

```

```

285     '205F',
286     '2060', '2061', '2062', '2063', '2064', '2065', '2066', '2067',
287     '2068', '2069', '206A', '206B', '206C', '206D', '206E', '206F',
288 ];
289 $arr_s = [];
290 $arr_s2 = [];
291 $arr_s3 = [];
292 foreach($arr as $s_hexa) {
293     $s = hexa_code_point_to_UTF8($s_hexa);
294     $arr_s []= $s;
295     if(strlen($s) === 2) {
296         $s2 = '\x'.bin2hex($s[0]).'\x'.bin2hex($s[1]);
297     }
298     if(strlen($s) === 3) {
299         $s2 = '\x'.bin2hex($s[0]).'\x'.bin2hex($s[1])
300             .'\'\x'.bin2hex($s[2]);
301     }
302     $arr_s2 []= $s2;
303     $s3 = " | sed -e 's/" . $s2 . "/g' \\";
304     $arr_s3 []= $s3;
305     echo($s3."\n");
306 }
307 // */
308

```

2.53 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     "${LFBFL_fixed_point_build}"
44 }
45
46 upgrade_build_and_checks "$@"

```

2.54 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 !