

```
1 #!/bin/bash
2 # shellcheck source=kzcommon.sh
3 #-----#
4 # Scripts distribueren en uploaden. #
5 # #
6 # Geschreven door Karel Zimmer <info@karelzimmer.nl>. #
7 # #
8 # Auteursrecht (c) 2009-2020 Karel Zimmer. #
9 # GNU Algemene Publieke Licentie <https://www.gnu.org/licenses/gpl.html>. #
10 # #
11 # RelNum=58.05.07 #
12 # RelDat=2020-11-16 #
13 #-----#
14
15 #-----#
16 # Global constants #
17 #-----#
18 source "${dirname "$0"}/kzcommon.sh
19 readonly THIS_YEAR=$(date +%Y)
20 readonly ARCHIVELOG=$HOME/kzscripts.arch/$THIS_YEAR/CHANGELOG_$THIS_YEAR
21 readonly ARCHIVELOG_GZ=$ARCHIVELOG.gz
22 readonly CHANGELOG=$HOME/kzscripts/CHANGELOG
23 readonly CHANGELOG_GZ=$CHANGELOG.gz
24 readonly CHECKLISTSDIR=$HOME/Documenten/Checklists
25 readonly CHECKLISTSARCHDIR=$HOME/Documenten/Checklists.arch
26 readonly DEBARCHDIR=$HOME/kzscripts.deb.arch/$THIS_YEAR
27 readonly DEBDIR=$HOME/kzscripts.deb
28 readonly DEBNAME=kzscripts_365_all.deb
29 readonly DIFF=$HOME/.$PROGRAMME.diff
30 readonly LASTRUN=$HOME/.$PROGRAMME.lastrun
31 readonly LAST_YEAR=$((THIS_YEAR - 1))
32 readonly REPORT=$HOME/.$PROGRAMME.rpt
33 readonly SCRIPTSARCHDIR=$HOME/kzscripts.arch/$THIS_YEAR
34 readonly SCRIPTSDIR=$HOME/kzscripts
35 readonly SITEDIR=$HOME/uploads/karelzimmer.nl/httpdocs
36 readonly TARNAME=kzscripts.tar.gz
37
38 # @public
39 # De SOURCE_UPLOADS[TAR] en [DEB] worden door dit script gegenereerd.
40 readonly -A SOURCE_UPLOADS=(
41     [TAR]=$SITEDIR/apps/kzscripts
42     [CHECKLISTS]=$HOME/Documenten/Checklists
43     [DEB]=$SITEDIR/apps/kzscripts
44     [PDFS_1]=$HOME/Documenten
45     [PDFS_2]=$HOME/Documenten/Checklists
46     [SCRIPTS]=$SCRIPTSDIR
47 )
48 readonly -A TARGET_UPLOADS=(
49     [KZSCRIPTS]=$SITEDIR/apps/kzscripts
50     [CHECKLISTS]=$SITEDIR/data/linux/documents
51     [PDFS]=$SITEDIR/data/linux/pdfs
52     [SCRIPTS]=$SITEDIR/data/linux/scripts
53 )
54
55 # @public
56 readonly UPLOADS=$HOME/uploads
57 readonly UPLOADS_OLD=$HOME/uploads.old
58
59 # @public
60 # De SOURCE_DEB[BIN] wordt gevuld door SCRIPTSDIR.
61 # De BACKGROUND_PROP wordt door dit script gegenereerd.
62 # Enkele mappen en bestanden (zoals pixmaps en control) staan al in SOURCE_DEB.
63 # Te vullen mappen zijn relatief ten opzichte van $DEBDIR.
64 readonly -A SOURCE_DEB=(
65     [APPLICATIONS]=/usr/share/applications
66     [BACKGROUNDS]=/usr/share/backgrounds
```

```
67     [BACKGROUND_PROP]=/usr/share/gnome-background-properties
68         [BIN]=/usr/bin
69         [DOC]=/usr/share/doc/kzscripts
70         [MAN]=/usr/share/man/man1
71     [POLICY_1]=/usr/share/polkit-1/actions
72     [README]=/usr/share/doc/kzscripts/README
73     [TABCOMPL]=/usr/share/bash-completion/completions
74 )
75 readonly TARGET_DEB=${TARGET_UPLOADS[KZSCRIPTS]}
76
77 # @public
78 # Te synchroniseren mappen en bestanden zijn relatief ten opzichte van $HOME.
79 readonly -A SOURCE_TAR=(
80     [CHECKLISTS-1]=Documenten/Checklists
81     [CHECKLISTS-2]=Documenten/Checklists.arch
82     [DEB-1]=kzscripts.deb
83     [DEB-2]=kzscripts.deb.arch
84     [ONTWIKKELSCRIPTS]=scripts
85     [PROGRAMMAS]="Programma's"
86     [SCRIPTSARCHIEF_1]=kzscripts.arch/$THIS_YEAR
87     [SCRIPTSARCHIEF_2]=kzscripts.arch/$LAST_YEAR.tar.xz
88     [SCRIPTS]=kzscripts
89     [TODO]=ToDo
90 )
91 readonly TARGET_TAR=${TARGET_UPLOADS[KZSCRIPTS]}
92
93 # @private
94 # Te synchroniseren mappen zijn relatief ten opzichte van $HOME.
95 readonly -A SOURCE_GOOGLE_DRIVE=(
96     [ADRESSEN]=Documenten/Adressen
97     [CAH]=Documenten/CaH
98     [GEBRUIKERSFOTO]=Afbeeldingen/Gebruikersfoto
99     [SCANS]=Afbeeldingen/Scans
100     [SLEUTELBOSSEN]=Documenten/Karel/Sleutelbossen
101 )
102 readonly TARGET_GOOGLE_DRIVE=google-drive://karel.zimmer@gmail.com
103
104 # Bij aanpassingen ook .completion aanpassen!
105 readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON'elm:r'
106 readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON',edit,list,message:,reset'
107 readonly USAGE="Gebruik: $PROGNAME -m|--message VRIJGAVEBERICHT [-e|--edit]
108     [-l|--list] [-r|--reset]
109     $OPTIONS_USAGE_COMMON
110
111 $OPTIONS_LONG_SHORT"
112 readonly HELP="Gebruik: $PROGNAME [OPTIE...]"
113
114 Scripts distribueren en uploaden.
115
116 Opties:
117     $OPTIONS_LONG_SHORT
118
119     -e --edit           Wijzig verschillenbestand
120     -l --list          Toon overzicht wijzigingen
121     -m --message       Vrijgavebericht opgeven
122     -r --reset         Maak verschillenbestand leeg
123 $OPTIONS_HELP_COMMON
124
125 Argument:
126     VRIJGAVEBERICHT   Vrijgavebericht, geef '=' op voor vorig vrijgavebericht"
127
128 #-----#
129 # Global variables                                     #
130 #-----#
131 declare MESSAGE_ARGUMENT=' '
132 declare OPTION_EDIT=false
```

```
133 declare OPTION_LIST=false
134 declare OPTION_MESSAGE=false
135 declare OPTION_RESET=false
136 declare REL_MSG=''
137
138 #-----#
139 # Functions #
140 #-----#
141 check_input() {
142     local -i getopt_rc=0
143     local parsed=''
144     local karel_full_name=''
145     local ubuntu_full_name=''
146
147     parsed=$(
148         getopt --alternative \
149                --options "$OPTIONS_SHORT" \
150                --longoptions "$OPTIONS_LONG" \
151                --name "$PROGNAME" \
152                -- "$@"
153     ) || getopt_rc=$?
154     if [[ $getopt_rc -ne 0 ]]; then
155         printf '%s\n' "$USAGELINE" >&2
156         QUIET=true exit $ERROR
157     fi
158     eval set -- "$parsed"
159     process_general_options "$@"
160
161     while true; do
162         case $1 in
163             -e|--edit)
164                 OPTION_EDIT=true
165                 shift
166                 ;;
167             -l|--list)
168                 OPTION_LIST=true
169                 shift
170                 ;;
171             -m|--message)
172                 if $OPTION_MESSAGE; then
173                     printf "$PROGNAME: %s\n%s\n" \
174                            "optie '$1' éénmaal opgeven" "$USAGELINE" >&2
175                     QUIET=true exit $ERROR
176                 else
177                     OPTION_MESSAGE=true
178                     MESSAGE_ARGUMENT=$2
179                 fi
180                 shift 2
181                 ;;
182             -r|--reset)
183                 OPTION_RESET=true
184                 shift
185                 ;;
186             --)
187                 shift
188                 break
189                 ;;
190             *)
191                 shift
192                 ;;
193         esac
194     done
195
196     if [[ "$*" ]]; then
197         printf "$PROGNAME: %s\n%s\n" 'geen argumenten opgeven' "$USAGELINE" >&2
198         QUIET=true exit $ERROR
199     fi
200 }
```

```

199     elif $OPTION_MESSAGE && [[ -z $MESSAGE_ARGUMENT ]]; then
200         printf "$PROGNAME: %s\n%s\n" \
201             "optie '--message' vereist een niet-leeg argument" \
202             "$USAGELINE" >&2
203     QUIET=true exit $ERROR
204 fi
205
206 karel_full_name=$(
207     getent passwd karel |
208     awk -F: '{print $5}' |
209     awk -F, '{print $1}' || true
210 )
211 ubuntu_full_name=$(
212     getent passwd ubuntu |
213     awk -F: '{print $5}' |
214     awk -F, '{print $1}' || true
215 )
216 if ! [[ ($USER = karel && $karel_full_name = 'Karel Zimmer') ||
217     ($USER = ubuntu && $ubuntu_full_name = 'Live session user') ]]; then
218     printf "$PROGNAME: %s\n" 'alleen uitvoeren als ontwikkelaar' \
219 "$USAGELINE" >&2
220     QUIET=true exit $ERROR
221 fi
222
223 # Een non-gui script gestart met optie gui.
224 if $OPTION_GUI; then
225     OPTION_GUI=false
226     TERMINAL=true
227 fi
228
229 check_user
230 }
231
232 process_input() {
233     if ! [[ -e "$LASTRUN" ]]; then
234         touch "$LASTRUN"
235     fi
236
237     if ! [[ -e "$HOME/${SOURCE_TAR[SCRIPTSARCHIEF_2]}" ]]; then
238         info "Wisselng van het jaar $LAST_YEAR naar $THIS_YEAR."
239         info "Maak zelf ${SOURCE_TAR[SCRIPTSARCHIEF_2]} aan, en"
240         info "maak zelf $CHANGELÖG.gz leeg."
241         exit 0
242     fi
243
244     process_option_edit_reset $OPTION_EDIT $OPTION_RESET
245
246     if $OPTION_LIST; then
247         process_option_list
248         QUIET=true exit $SUCCESS
249     fi
250
251     if ! $OPTION_MESSAGE; then
252         read -rp \
253             'Vrijgavebericht, '=' voor vorig vrijgavebericht [=]: ' \
254             < /dev/tty
255         if [[ -z $REPLY ]]; then
256             REL_MSG='='
257         else
258             REL_MSG=$REPLY
259         fi
260     else
261         REL_MSG=$MESSAGE_ARGUMENT
262     fi
263
264     check_scripts

```

```
265     if [[ $? -ne $SUCCESS ]]; then
266         warning "
267 Herstel alle fouten hierboven en start daarna weer $PROGRAMME."
268         QUIET=true exit $ERROR
269     fi
270
271     archive_scripts
272     create_scriptspackage
273     create_scriptsarchive
274     synchronize_uploads
275     determine_changes
276     synchronize_uploads_old
277     synchronize_google_drive
278     show_changes
279     upload_website
280     install_scriptspackage
281 }
282
283 process_option_list() {
284     if [[ -s "$REPORT" ]]; then
285         cat "$REPORT"
286         while true; do
287             read -rp 'Resultaat afdrukken? [j/N]: ' < /dev/tty
288             case $REPLY in
289                 j*|J*)
290                     lpr "$REPORT" |& $LOGCMD
291                     info 'Overzicht is verzonden naar de printer.'
292                     break
293                     ;;
294                 n*|N*|'')
295                     break
296                     ;;
297                 *)
298                     printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
299                     continue
300                     ;;
301             esac
302         done
303     else
304         info 'Het rapportbestand is leeg.'
305     fi
306 }
307
308 process_option_edit_reset() {
309     local editopt=${1:-false}
310     local resetopt=${2:-false}
311
312     if $editopt && $resetopt; then
313         printf '%s\n' \
314             "Specificeer optie '--reset' of optie '--edit', niet allebei." \
315             >&2
316         QUIET=true exit $ERROR
317     fi
318
319     if $editopt; then
320         gedit "$DIFF"
321         QUIET=true exit $SUCCESS
322     fi
323
324     if $resetopt; then
325         truncate --size=0 "$DIFF" "$REPORT" |& $LOGCMD
326         info 'Verschillenbestand is leeggemaakt.'
327     fi
328 }
329
330 archive_scripts() {
```

```

331     local archivefile=''
332     local release_date=''
333     local release_number=''
334     local scriptfile=''
335
336     remove_backups
337     archive_upfront
338
339     printf "%s\n" 'Archiveer scripts ...'
340
341     if ! [[ -d $SCRIPTSARCHDIR ]]; then
342         mkdir --parents \
343             "$SCRIPTSARCHDIR" |& $LOGCMD
344     fi
345     if [[ -e $CHANGELOG_GZ ]]; then
346         gunzip --force \
347             "$CHANGELOG_GZ" |& $LOGCMD
348     fi
349
350     # Te verwerken scripts staan in SCRIPTSDIR.
351     cd "$SCRIPTSDIR"
352     for scriptfile in *; do
353
354         if [[ $scriptfile = CHANGELOG* ]]; then
355             continue
356         fi
357
358         check_archivefolder "$SCRIPTSARCHDIR/$scriptfile"
359
360         release_number=$(
361             awk -FRelNum= \
362                 '/RelNum={print $2;exit}' \
363                 "$scriptfile"
364         )
365         release_number=${release_number:0:8}
366         release_date=$(
367             awk -FRelDat= \
368                 '/RelDat={print $2;exit}' \
369                 "$scriptfile"
370         )
371         release_date=${release_date:0:10}
372         archivefile=${scriptfile}_${release_date}V$release_number"
373
374         # Archiveer het bestand als deze nieuw of gewijzigd is.
375         if ! diff "$scriptfile" \
376             "$SCRIPTSARCHDIR/$scriptfile/$archivefile" \
377             &> /dev/null; then
378             cp --update \
379                 --preserve \
380                 "$scriptfile" \
381                 "$SCRIPTSARCHDIR/$scriptfile/$archivefile" |& $LOGCMD
382             chmod 'a-x' \
383                 "$SCRIPTSARCHDIR/$scriptfile/$archivefile" |& $LOGCMD
384             printf "%s\n" \
385                 "$scriptfile:$release_number:$release_date:\
386 $REL_MSG" \
387                 >> "$CHANGELOG"
388             if grep --quiet \
389                 --word-regexp \
390                 --regexp='^\.TH ' \
391                 "$scriptfile"; then
392                 process_manpage
393             fi
394         fi
395     done
396 }

```

```
397
398 remove_backups() {
399     local dir=''
400
401     for dir in $UPLOADS; do
402         find "$dir" \
403             -name '*~' -delete -or \
404             -name '*.*~' -delete -or \
405             -name '*.bak' -delete |& $LOGCMD
406     done
407 }
408
409 archive_upfront() {
410     if ! [[ -d $DEBARCHDIR ]]; then
411         mkdir --parents \
412             "$DEBARCHDIR" |& $LOGCMD
413     fi
414     rsync --archive \
415         "$DEBDIR"/ \
416         "$DEBARCHDIR" |& $LOGCMD
417
418     # Vanuit gedit.
419     if ls "${SOURCE_UPLOADS[PDFS_1]}/*.pdf &> /dev/null; then
420         mv "${SOURCE_UPLOADS[PDFS_1]}/*.pdf \
421             "${TARGET_UPLOADS[PDFS]}" |& $LOGCMD
422     fi
423
424     # Vanuit LibreOffice.
425     if ls "${SOURCE_UPLOADS[PDFS_2]}/*.pdf &> /dev/null; then
426         mv "${SOURCE_UPLOADS[PDFS_2]}/*.pdf \
427             "${TARGET_UPLOADS[PDFS]}" |& $LOGCMD
428     fi
429
430     if ! [[ -d $CHECKLISTSARCHDIR ]]; then
431         mkdir --parents \
432             "$CHECKLISTSARCHDIR" |& $LOGCMD
433     fi
434     rsync --archive \
435         "$CHECKLISTSDIR"/ \
436         "$CHECKLISTSARCHDIR" |& $LOGCMD
437 }
438
439 check_archivefolder() {
440     local archivedir=${1:-archivedir?}
441
442     if ! [[ -d $archivedir ]]; then
443         mkdir "$archivedir" |& $LOGCMD
444     fi
445 }
446
447 process_manpage() {
448     man --troff \
449         "$scriptfile" \
450     ps2pdf - \
451         "${TARGET_UPLOADS[PDFS]}/${scriptfile}.pdf" |& $LOGCMD
452 }
453
454 check_scripts() {
455     local -i check_rc=0
456
457     info 'Controleer scripts (kzivpsh) ...'
458     "$SCRIPTSDIR/kzivpsh" || check_rc=$?
459     if [[ $check_rc -ne $SUCCESS ]]; then
460         warning ""
461         Herstel alle meldingen hierboven en start daarna weer $PROGRAMME."
462         QUIET=true exit $check_rc

```

```

463     fi
464 }
465
466 create_scriptspackage() {
467     # Te verwerken scripts staan in SCRIPTSDIR.
468     cd "$SCRIPTSDIR"
469     chmod 'u=rwx,g=rx,o=rx' -- *
470     chmod 'a-x'             -- *.*
471
472     sort --version-sort \
473         --unique \
474         "$CHANGELOG" \
475         --output="$CHANGELOG" |& $LOGCMD
476     gzip --best \
477         --force \
478         "$CHANGELOG" |& $LOGCMD
479     cp "$CHANGELOG_GZ" \
480        "$ARCHIVELOG_GZ" |& $LOGCMD
481     chmod 'u=rw,g=r,o=r' \
482         -- \
483         "$ARCHIVELOG_GZ" \
484         "$CHANGELOG_GZ"
485
486     # Deze rsync is inclusief CHANGELOG.gz, *.7, *.completion, *.desktop, en
487     # *.policy.
488     rsync --archive \
489         --delete \
490         "$SCRIPTSDIR"/ \
491         "$DEBDIR${SOURCE_DEB[BIN]}" |& $LOGCMD
492     printf "%s\n" 'Maak scriptspakket ...'
493     process_scriptspackage_files
494     create_wallpapapers_configuration_file
495
496     # Voor Ubuntu 12.04 LTS en Ubuntu 14.04 LTS.
497     # -Zgzip toegevoegd, xz niet ondersteund.
498     fakeroot dpkg-deb \
499         -Zgzip \
500         --build \
501         "$DEBDIR" \
502         "$TARGET_DEB/$DEBNAME" |& $LOGCMD
503 }
504
505 process_scriptspackage_files() {
506     local basename=''
507     local file=''
508     local newfile=''
509     local prefix='nl.karelzimmer.'
510
511     # Changelog (.gz)
512     mv "$DEBDIR${SOURCE_DEB[BIN]}" /CHANGELOG.gz \
513        "$DEBDIR${SOURCE_DEB[DOC]}" |& $LOGCMD
514
515     # Man-pagina's (.7)
516     rm --force \
517        "$DEBDIR${SOURCE_DEB[MAN]}"/* \
518     rsync --archive \
519         --delete \
520         --remove-source-files \
521         "$DEBDIR${SOURCE_DEB[BIN]}"/*.1 \
522         "$DEBDIR${SOURCE_DEB[MAN]}" |& $LOGCMD
523     gzip --best \
524         --force \
525         "$DEBDIR${SOURCE_DEB[MAN]}"/*.1 |& $LOGCMD
526
527     # Tab-completion-scripts (.completion)
528     rm --force \

```



```

529     "$DEBDIR${SOURCE_DEB[TABCOMPL]}"/*
530 for file in "$DEBDIR${SOURCE_DEB[BIN]}"/*.completion; do
531     basename=$(basename "$file" .completion)
532     mv "$file" \
533     "$DEBDIR${SOURCE_DEB[TABCOMPL]}/$basename" |& $LOGCMD
534 done
535
536 # Bureaublad-configuratiebestanden (.desktop)
537 rm --force \
538     "$DEBDIR${SOURCE_DEB[APPLICATIONS]}"/*
539 for file in "$DEBDIR${SOURCE_DEB[BIN]}"/*.desktop; do
540     basename=$(basename "$file")
541     newfile="$DEBDIR${SOURCE_DEB[APPLICATIONS]}/$prefix$basename"
542     mv "$file" \
543     "$newfile" |& $LOGCMD
544 done
545
546 # PolicyKit actiedefinitiebestanden (.policy)
547 rm --force \
548     "$DEBDIR${SOURCE_DEB[POLICY_1]}"/*
549 for file in "$DEBDIR${SOURCE_DEB[BIN]}"/*.policy; do
550     basename=$(basename "$file")
551     newfile="$DEBDIR${SOURCE_DEB[POLICY_1]}/$prefix$basename"
552     mv "$file" \
553     "$newfile" |& $LOGCMD
554 done
555 }
556
557 create_wallpapers_configuration_file() {
558     local wallpaper='
559     local wallpapers_config_file=kzscripts-wallpapers.xml
560
561     {
562     printf '%s\n' '<?xml version="1.0" encoding="UTF-8"?>'
563     printf '%s\n' '<!DOCTYPE wallpapers SYSTEM "gnome-wp-list.dtd">'
564     printf '%s\n' '<wallpapers>'
565     } > "$DEBDIR${SOURCE_DEB[BACKGROUND_PROP]}/$wallpapers_config_file"
566
567     # Te verwerken bureaubladachtergronden staan in SOURCE_DEB[BACKGROUNDS].
568     cd "$DEBDIR${SOURCE_DEB[BACKGROUNDS]}"
569
570     for wallpaper in *; do
571         printf '%s\n' "    <wallpaper>"
572         printf '%s\n' "    <name>${wallpaper%.*}</name>"
573         printf '%s\n' "    <filename>${SOURCE_DEB[BACKGROUNDS]}/$wallpaper</filename>"
574         printf '%s\n' "    <options>zoom</options>"
575         printf '%s\n' "    <pcolor>#000000</pcolor>"
576         printf '%s\n' "    <scolor>#000000</scolor>"
577         printf '%s\n' "    <shade_type>solid</shade_type>"
578         printf '%s\n' "</wallpaper>" \
579         >> "$DEBDIR${SOURCE_DEB[BACKGROUND_PROP]}/$wallpapers_config_file"
580     done
581     printf '%s\n' "    </wallpapers>" \
582     >> "$DEBDIR${SOURCE_DEB[BACKGROUND_PROP]}/$wallpapers_config_file"
583 }
584
585
586 create_scriptsarchive() {
587     # Te synchroniseren mappen en bestanden in SOURCE_TAR zijn relatief ten
588     # opzichte van $HOME.
589     cd "$HOME"
590
591     printf "%s\n" 'Maak scriptsarchieef ...'
592
593     tar --create \
594     --gzip \

```

```

595     "${SOURCE_TAR[@]}" \
596     --file="$TARGET_TAR/$TARNAME" |& $LOGCMD
597 }
598
599 synchronize_uploads() {
600     printf "%s\n" 'Synchroniseer uploads ...'
601
602     rsync --archive \
603         --delete \
604         "${SOURCE_UPLOADS[CHECKLISTS]}/" \
605         "${TARGET_UPLOADS[CHECKLISTS]}" |& $LOGCMD
606     rsync --archive \
607         --delete \
608         "${SOURCE_UPLOADS[SCRIPTS]}/" \
609         "${TARGET_UPLOADS[SCRIPTS]}" |& $LOGCMD
610 }
611
612 determine_changes() {
613     touch "$DIFF" |& $LOGCMD
614     diff --brief \
615         --recursive \
616         "$UPLOADS" \
617         "$UPLOADS_OLD" \
618         >> "$DIFF" || true
619 }
620
621 synchronize_uploads_old() {
622     printf "%s\n" 'Synchroniseer uploads-oud ...'
623
624     rsync --archive \
625         --delete \
626         "$UPLOADS/" \
627         "$UPLOADS_OLD" |& $LOGCMD
628 }
629
630 synchronize_google_drive() {
631     local dir=''
632     local file=''
633
634     if ! [[ $HOSTNAME = pc06 ]]; then
635         printf "%s\n" 'Synchroniseer Google Drive overslaan (niet op pc06).'
636         return $SUCCESS
637     fi
638
639     printf "%s\n" 'Synchroniseer Google Drive ...'
640     if gio info "$TARGET_GOOGLE_DRIVE" &> /dev/null; then
641         gio mount \
642             --unmount \
643             "$TARGET_GOOGLE_DRIVE"
644         gio mount \
645             "$TARGET_GOOGLE_DRIVE"
646     else
647         gio mount \
648             "$TARGET_GOOGLE_DRIVE"
649     fi
650
651     # Te synchroniseren mappen en bestanden in SOURCE_GOOGLE_DRIVE zijn relatief
652     # ten opzichte van $HOME.
653     cd "$HOME"
654
655     info 'Zoek mappen/bestanden voor Google Drive ...'
656     while read -r file; do
657         if [[ -d "$file" ]]; then
658             if ! gio info "$TARGET_GOOGLE_DRIVE/$file" &> /dev/null; then
659                 info "Aanmaken $file ..."
660                 gio mkdir \

```

```

661         --parent          \
662         "$TARGET_GOOGLE_DRIVE/$file"
663     fi
664 else
665     info "Uploaden $file ..."
666     gio copy                \
667     "$file"                \
668     "$TARGET_GOOGLE_DRIVE/${dirname "$file"}"
669 fi
670 done < <(
671     find "${SOURCE_GOOGLE_DRIVE[@]}" \
672     -newer "$LASTRUN" \
673     -print
674 )
675
676 gio mount                \
677 --unmount                \
678 "$TARGET_GOOGLE_DRIVE"
679 }
680
681 show_changes() {
682     local hdr=' '
683     local -i num_of_changes=0
684     local record=' '
685     local temp_record=' '
686
687     touch "$DIFF.temp1"
688     # Wijzig verschillenlijst (van DIFF naar DIFF.temp1)
689     # -----
690     # Verwijder aan het begin 'Bestanden ' en alles vanaf ' en /'.
691     # Verwijder aan het begin 'Alleen in ' en vervang '<pad>: <bestand>' door
692     # '<pad>/<bestand>'.
693     while read -r record; do
694         temp_record=$record
695
696         # Sluit bestanden uit die altijd wijzigen.
697         if [[ $temp_record == *CHANGELOG* ]]; then
698             continue
699         elif [[ $temp_record == *$DEBNAME* ]]; then
700             continue
701         elif [[ $temp_record == *$TARNAME* ]]; then
702             continue
703         fi
704
705         temp_record=${temp_record#Bestanden }
706         temp_record=${temp_record% en /*}
707
708         temp_record=${temp_record#Alleen in }
709         temp_record=${temp_record//: /\//}
710         temp_record=$(basename "$temp_record")
711
712         printf '%s\n' "_ $temp_record" >> "$DIFF.temp1"
713     done < "$DIFF"
714
715     sort --unique          \
716     "$DIFF.temp1" >> "$DIFF.temp2"
717
718     num_of_changes=$(wc --lines < "$DIFF.temp2")
719     hdr="Overzicht wijzigingen ($num_of_changes)"
720
721     touch "$REPORT"
722     pr --date-format='%a %d %b %Y' \
723     --form-feed \
724     --header="$hdr" \
725     "$DIFF.temp2" |
726     tee "$REPORT"

```

```
727     if [[ $num_of_changes -eq 0 ]]; then
728         info 'Niets gewijzigd.'
729     fi
730
731     rm "$DIFF".temp* |& $LOGCMD
732 }
733
734 upload_website() {
735     local ftp_set='set ssl:verify-certificate no'
736     local ftp_from=$SITEDIR
737     local ftp_to=/httpdocs
738     local ftp_opts='--reverse --delete --verbose'
739     local ftp_cmd="mirror $ftp_opts $ftp_from $ftp_to"
740     local ftp_host=server106.hosting2go.nl
741     local ftp_user=kzimmer
742     local ftp_login=$HOME/.kz-$ftp_host
743
744     # Push scriptspakket, scriptsarchief, website.
745     if ! [[ -f $ftp_login ]]; then
746         read -rsp "FTP-wachtwoord voor $ftp_host: "
747         printf '%s\n' "$REPLY" > "$ftp_login"
748         printf '\n'
749         chmod 'u=rw,g=,o=' "$ftp_login" |& $LOGCMD
750     fi
751     info 'Upload website ...'
752     if ! lftp --user "$ftp_user" \
753         --password "$(cat "$ftp_login")" \
754         -e "$ftp_set; $ftp_cmd; exit" \
755         "$ftp_host" |& $LOGCMD; then
756         warning 'Uploaden website is niet uitgevoerd.'
757         QUIET=true exit $WARNING
758     fi
759 }
760
761 install_scriptspackage() {
762     info 'Installeer scriptspakket (kzgetscripts) ...'
763     # Constructie '2> >($LOGCMD)' om stderr naar de log te krijgen.
764     if ! wget -O- karelzimmer.nl/s 2> >($LOGCMD) | bash; then
765         warning 'Installeren scriptspakket is niet uitgevoerd.'
766         QUIET=true exit $WARNING
767     fi
768 }
769
770 term_script() {
771     touch "$LASTRUN"
772     QUIET=true exit $SUCCESS
773 }
774
775 #-----#
776 # Main line #
777 #-----#
778 main() {
779     init_script
780     check_input "$@"
781     process_input
782     term_script
783 }
784
785 main "$@"
786
787 # EOF
```