

```
1  #!/bin/bash
2  # shellcheck source=common.sh
3  # #####
4  # Bestand: restore
5  # Doel: Zet back-up (reservekopie) terug
6  # Gebruik: Met starter 'Restore' (aanbevolen)
7  # of: Met starter 'Menu', kies 'Menu Opdrachten' (aanbevolen)
8  # of: In het terminalvenster:
9  # /opt/kzscripts/restore
10 # of:
11 # kzrestore
12 # Auteur: Karel Zimmer (https://karelzimmer.nl, info@karelzimmer.nl)
13 # -----
14 # Auteursrecht (c) 2007-2020 Karel Zimmer.
15 #
16 # Dit programma is vrije software: u mag het herdistribueren en/of wijzigen
17 # onder de voorwaarden van de GNU Algemene Publieke Licentie zoals gepubliceerd
18 # door de Free Software Foundation, onder versie 3 van de Licentie of (naar uw
19 # keuze) elke latere versie.
20 #
21 # Dit programma is gedistribueerd in de hoop dat het nuttig zal zijn maar ZONDER
22 # ENIGE GARANTIE; zelfs zonder de impliciete garanties die GEBRUIKELIJK ZIJN IN
23 # DE HANDEL of voor BRUIKBAARHEID VOOR EEN SPECIFIEK DOEL.
24 # Zie de GNU Algemene Publieke Licentie voor meer details.
25 #
26 # U hoort een kopie van de GNU Algemene Publieke Licentie te hebben ontvangen
27 # samen met dit programma. Als dat niet het geval is, zie
28 # http://www.gnu.org/licenses/.
29 # #####
30 readonly REL_NUM=36.03.01
31 readonly REL_DAT=2020-03-28
32 readonly REL_MSG='Teksten gewijzigd'
33
34 # #####
35 # Instellingen
36 # #####
37 source "$(dirname "$(realpath "$0")")/common.sh
38
39 # -----
40 # Globale constanten
41 # -----
42 declare -ir BUPNUMLEN=2
43 readonly EXCLUDEFILE_DEFAULT=/root/.$PROGRAMME-exclude-dflt
44 readonly EXCLUDEFILE_DEFAULT_CONTENT="\
45 $(find /etc/skel -type f -exec basename '{}' \; | awk '$0="/home/*/"$0')
46 # Hieboven worden de bestanden opgenomen die bij het aanmaken van een
47 # gebruiker worden neergezet in $HOME. Meestal zijn dit de bestanden
48 # .bash_logout, .bashrc, examples.desktop, en .profile (bij Ubuntu).
49 # Deze bestanden terugzetten bij een nieuwe versie van Linux betekent dat
50 # deze bestanden worden teruggezet naar de oude versie; niet wenselijk.
51 readonly EXCLUDEFILE_OPTIONAL=/root/.$PROGRAMME-exclude-opt
52 readonly SELECTION_DEFAULT=''
53 readonly RUN_AS_SUPERUSER=true
54 readonly SOURCE_DEFAULT=/media
55 readonly TARGET_DEFAULT=/
56
57 readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON's:t:x:'
58 readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON',source:,target:,exclude:'
59 readonly OPTIONS_TAB_COMPLETION=$OPTIONS_TAB_COMPLETION_COMMON" --source \
60 --target --exclude"
61 readonly USAGE="Gebruik: $PROGRAMME2 [-s|--source=BRON] \
62 [-t|--target=DOEL] [-x|--exclude=UITSLUITEN]..."
63 $OPTIONS_USAGE_FILLER$OPTIONS_USAGE_COMMON
64 $OPTIONS_USAGE_FILLER[SELECTIE...]
65
66 $OPTIONS_LONG_SHORT
67
68 $ARGUMENT_LONG_SHORT"
69 readonly HELP="Gebruik: $PROGRAMME2 [OPTIE...] [--] [SELECTIE...]"
```

```

70
71  Zet back-up (reservekopie) terug.
72
73  Opties:
74    $OPTIONS_LONG_SHORT
75
76    $ARGUMENT_LONG_SHORT
77
78    -s --source=BRON    Zoek back-upbestand vanaf map BRON
79    -t --target=DOEL   Zet back-up terug in map DOEL
80    -x --exclude=UITSLUITEN
81                               Sluit uit bestand of map in UITSLUITEN
82  $OPTIONS_HELP_COMMON
83
84  Argument:
85    SELECTIE           Zet opgegeven mappen en bestanden terug"
86
87  # -----
88  # Globale variabelen
89  # -----
90  declare -a EXCLUDE_ARGUMENT=''
91  declare ARGUMENT_SELECTION=false
92  declare -a SELECTION=''
93  declare -a SELECTION_ARGUMENT=''
94  declare BACKUP=''
95  declare BACKUPSIZE_HUMAN=''
96  declare -i BACKUPSIZE_MACHINE=0
97  declare -i TAR_RC=0
98  declare OPTION_EXCLUDE=false
99  declare OPTION_SOURCE=false
100 declare OPTION_TARGET=false
101 declare SOURCE=''
102 declare SOURCE_ARGUMENT=''
103 declare TARGET=''
104 declare TARGET_ARGUMENT=''
105
106 # #####
107 # Functies
108 # #####
109 controleer_invoer() {
110     local -i exclude_arg_num=0
111     local -i getopt_rc=0
112     local -i select_arg_num=0
113     local -i select_num=0
114     local parsed=''
115
116     set +o errexit
117     parsed=$(getopt --alternative \
118                 --options "$OPTIONS_SHORT" \
119                 --longoptions "$OPTIONS_LONG" \
120                 --name "$PROGNAME" \
121                 -- "$@")
122     getopt_rc=$?
123     set -o errexit
124     if [[ $getopt_rc -ne 0 ]]; then
125         printf '%s\n' "$HELPLINE" >&2
126         quiet; exit $ERROR
127     fi
128     eval set -- "$parsed"
129     verwerk_algemene_opties "$@"
130
131     while true; do
132         case $1 in
133             -s|--source)
134                 if $OPTION_SOURCE; then
135                     printf "$PROGNAME: %s\n%s\n" "optie '$1' éénmaal opgeven" \
136 "$HELPLINE" >&2
137                     quiet; exit $ERROR
138                 else

```

```
139             OPTION_SOURCE=true
140             SOURCE_ARGUMENT=$2
141         fi
142         shift 2
143         ;;
144     -t|--target)
145         if $OPTION_TARGET; then
146             printf "$PROGNAME: %s\n%s\n" "optie '$1' éénmaal opgeven" \
147 "$HELPLINE" >&2
148             quiet; exit $ERROR
149         else
150             OPTION_TARGET=true
151             TARGET_ARGUMENT=$2
152         fi
153         shift 2
154         ;;
155     -x|--exclude)
156         OPTION_EXCLUDE=true
157         EXCLUDE_ARGUMENT[$exclude_arg_num]=$2
158         (( ++exclude_arg_num ))
159         shift 2
160         ;;
161     --)
162         shift
163         break
164         ;;
165     *)
166         shift
167         ;;
168     esac
169 done
170
171 while [[ "$*" ]]; do
172     ARGUMENT_SELECTION=true
173     SELECTION_ARGUMENT[$select_arg_num]=$1
174     (( ++select_arg_num ))
175     shift
176 done
177
178 if $OPTION_SOURCE; then
179     SOURCE=$SOURCE_ARGUMENT
180     if [[ ! -d $SOURCE ]]; then
181         printf "$PROGNAME: %s\n%s\n" "map '$SOURCE' bestaat niet" \
182 "$HELPLINE" >&2
183         quiet; exit $ERROR
184     fi
185 else
186     SOURCE=$SOURCE_DEFAULT
187 fi
188
189 if $OPTION_TARGET; then
190     if [[ ! -d $TARGET_ARGUMENT ]]; then
191         printf "$PROGNAME: %s\n%s\n" "map '$TARGET_ARGUMENT' bestaat niet" \
192 "$HELPLINE" >&2
193         quiet; exit $ERROR
194     fi
195     TARGET=$TARGET_ARGUMENT
196 else
197     TARGET=$TARGET_DEFAULT
198 fi
199
200 if $ARGUMENT_SELECTION; then
201     for select_num in "${!SELECTION_ARGUMENT[@]}; do
202         if [[ ${SELECTION_ARGUMENT[$select_num]:0:1} = '/' ]]; then
203             SELECTION[$select_num]=${SELECTION_ARGUMENT[$select_num]:1}
204         else
205             SELECTION[$select_num]=${SELECTION_ARGUMENT[$select_num]}
206         fi
207     done
```

```
208     else
209         SELECTION[0]=$SELECTION_DEFAULT
210     fi
211
212     controleer_gebruiker
213     vraag_invoer
214 }
215
216 vraag_invoer() {
217     local title="$PROGNAME"
218
219     if $OPTION_GUI; then
220         selecteer_backup_gui
221     else
222         selecteer_backup_tui
223     fi
224 }
225
226 selecteer_backup_gui() {
227     local title="$PROGNAME - selecteer een back-upbestand"
228     local filename="{SOURCE_ARGUMENT:-/media}"
229
230     set +o errexit
231     BACKUP="$(
232         zenity --file-selection          \
233              --width=600                \
234              --height=50               \
235              --title="$title"          \
236              --filename="$filename"    \
237              --file-filter=*.tar       \
238              2> /dev/null
239     )"
240     # shellcheck disable=SC2181
241     if [[ $? -ne 0 ]]; then
242         quiet; exit $WARNING
243     fi
244     set -o errexit
245 }
246
247 selecteer_backup_tui() {
248     local range=""
249     local -a buparray
250     local -i bupnum=0
251
252     toon_tekst "Er wordt naar back-ups gezocht vanaf $SOURCE.
253 Het medium waarop de back-up staat dient aangekoppeld te zijn.
254 Voorbeelden van een medium zijn een (externe) (netwerk)schijf,
255 USB-stick, CD, of DVD.
256 Zoek nu naar back-ups van deze computer..."
257
258     toon_backups "$HOSTNAME"
259
260     if [[ $bupnum -eq 0 ]]; then
261         toon_waarschuwingstekst 'Geen back-ups gevonden van deze computer.'
262         REPLY='A'
263     fi
264
265     if [[ $bupnum -gt 0 ]]; then
266         printf '\n%s\n' \
267 "Kies nummer, of A om te zoeken naar alle back-ups, en S om te stoppen."
268         if [[ $bupnum -eq 1 ]]; then
269             range=""
270         else
271             range="- $bupnum"
272         fi
273         while true; do
274             read -rp "Terugzetten back-up nummer [1$range, of A, of S, \
275 standaard 1]: " </dev/tty
276             case $REPLY in
```

```

277         '' )
278             BACKUP=${buparray[1]}
279             break
280             ;;
281         a|A)
282             break
283             ;;
284         s|S)
285             printf '%s\n' 'Gestopt.'
286             quiet; exit $SUCCESS
287             ;;
288         *[^[:digit:]]*)
289             echo -e "${UP_ONE_LINE}${ERASE_LINE}"
290             continue
291             ;;
292         *)
293             if [[ $REPLY -ge 1 && $REPLY -le $bupnum ]] 2> /dev/null
294             then
295                 BACKUP=${buparray[$REPLY]}
296                 break
297             else
298                 echo -e "${UP_ONE_LINE}${ERASE_LINE}"
299                 continue
300             fi
301             ;;
302         esac
303     done
304 fi
305
306 if [[ $REPLY = [A,a] ]]; then
307     printf '\n%s\n' "Zoek nu naar back-ups van alle computers..."
308     bupnum=0
309     toon_backups ''
310     if [[ $bupnum -eq 0 ]]; then
311         toon_waarschuwingstekst 'Geen back-ups gevonden.'
312         quiet; exit $WARNING
313     elif [[ $bupnum -eq 1 ]]; then
314         range=''
315     else
316         range="- $bupnum"
317     fi
318     printf '\n%s\n' "Kies nummer, of S om te stoppen."
319     while true; do
320         read -rp "Terugzetten back-up nummer [1$range, of S, \
321 standaard S]: " </dev/tty
322         case $REPLY in
323             s|S|'')
324                 printf '%s\n' 'Gestopt.'
325                 quiet; exit $SUCCESS
326                 ;;
327             *[^[:digit:]]*)
328                 echo -e "${UP_ONE_LINE}${ERASE_LINE}"
329                 continue
330                 ;;
331             *)
332                 if [[ $REPLY -ge 1 && $REPLY -le $bupnum ]] 2> /dev/null
333                 then
334                     BACKUP=${buparray[$REPLY]}
335                     break
336                 else
337                     echo -e "${UP_ONE_LINE}${ERASE_LINE}"
338                     continue
339                 fi
340                 ;;
341             esac
342         done
343     fi
344 }
345

```

```

346 toon_backups() {
347     local search4=${1-search4?}
348     local backup=''
349     local dirname=''
350     local filename=''
351     local filesize_human=''
352
353     # Zoek vanaf bronmap ("$SOURCE") naar back-ups. Find zoals hier gebruikt
354     # levert, gescheiden door een TAB, back-upnaam en directory waarin de
355     # back-up zich bevindt. De sort sorteert op back-upnaam aflopend, dat is
356     # van nieuw naar oud.
357
358     while read -r backup; do
359         filename=$(echo "$backup" | cut --fields=1)
360         dirname=$(echo "$backup" | cut --fields=2)
361         filesize_human=$(
362             du --apparent-size          \
363                 --block-size=1         \
364                 --human-readable       \
365                 "$dirname/$filename"   |
366             awk '{print $1}'
367         )
368
369         (( ++bupnum ))
370         buparray[$bupnum]=$dirname/$filename
371
372         if [[ $bupnum -eq 1 ]]; then
373             if [[ -z $search4 ]]; then
374                 printf '\n%s\n' "De volgende back-ups zijn gevonden:"
375             else
376                 printf '\n%s\n' \
377 "De volgende back-ups van $search4 zijn gevonden, van nieuw naar oud:"
378             fi
379         fi
380
381         printf '%+*s %s\t%s\t%s\n'          \
382             $BUPNUMLEN $bupnum             \
383             "$(basename "$filename" ".tar)" \
384             "$filesize_human" "iB" "$dirname"
385
386         done << (
387             find $SOURCE                    \
388                 -type f                    \
389                 -noleaf                    \
390                 -name 'backup*"$search4"*.tar' \
391                 -printf '%f\t%h\n'         \
392                 2> /dev/null              |
393             sort --reverse
394         )
395     }
396
397     verwerk_invoer() {
398         local select=''
399         local size_num=0
400         local size_unit=''
401         local text=''
402         local title="$PROGNAME"
403
404         echo "$EXCLUDEFILE_DEFAULT_CONTENT" > "$EXCLUDEFILE_DEFAULT"
405
406         if $OPTION_EXCLUDE; then
407             for exclude in "${EXCLUDE_ARGUMENT[@]}; do
408                 echo "$exclude" >> "$EXCLUDEFILE_OPTIONAL"
409             done
410         else
411             cp /dev/null "$EXCLUDEFILE_OPTIONAL"
412         fi
413
414         text='Controleer beschikbare schijfruimte...'
415         if $OPTION_GUI; then

```

```

415     # Met "|& zenity --progress" worden globale variabelen uit
416     # aangeroepen functies niet doorgegeven, vandaar de
417     # 'process substitution' met "> >(zenity ...)".
418     controleer_schijfruimte > >(
419         zenity --progress \
420             --pulsate \
421             --auto-close \
422             --no-cancel \
423             --width=600 \
424             --height=50 \
425             --title="$title" \
426             --text="$text" \
427             2> /dev/null)
428     else
429         toon_tekst "$text"
430         controleer_schijfruimte
431     fi
432
433     bepaal_bestandsgrootte
434
435     if $ARGUMENT_SELECTION; then
436         select='Selectie'
437     else
438         select='Alles'
439     fi
440     size_num=${BACKUPSIZE_HUMAN%?}
441     size_unit=${BACKUPSIZE_HUMAN: -1}
442     text=" uit back-up $(basename "$BACKUP") [$size_num ${size_unit}iB] \
443     wordt teruggezet"
444     zet_backup_terug
445
446     if [[ $TAR_RC -eq 0 ]]; then
447         toon_tekst "$select uit back-up $(basename "$BACKUP") [$size_num \
448     ${size_unit}iB] is teruggezet."
449     fi
450 }
451
452 controleer_schijfruimte() {
453     local filesys=''
454     local free_human=''
455     local -i free_bytes=0
456     local -i need_bytes=0
457     local mounted=''
458     local need_human=''
459
460     need_bytes=$(
461         du --apparent-size \
462             --block-size=1 \
463             --exclude-from="$EXCLUDEFILE_DEFAULT" \
464             --exclude-from="$EXCLUDEFILE_OPTIONAL" \
465             --summarize \
466             --total \
467             "$BACKUP" \
468         awk 'END{print $1}'
469     )
470     free_bytes=$(df --block-size=1 "$TARGET" | awk 'END{print $4}')
471
472     if [[ $need_bytes -gt $free_bytes ]]; then
473         filesys=$(df "$TARGET" | awk 'END{print $1}')
474         mounted=$(df "$TARGET" | awk 'END{print $6}')
475         need_human=$(
476             du --apparent-size \
477                 --block-size=1 \
478                 --exclude-from="$EXCLUDEFILE_DEFAULT" \
479                 --exclude-from="$EXCLUDEFILE_OPTIONAL" \
480                 --human-readable \
481                 --summarize \
482                 --total \
483                 "$BACKUP" \

```

```

484         awk 'END{print $1}'
485     )
486     free_human=$(df --human-readable "$TARGET" | awk 'END{print $4}')
487
488     if [[ ${SELECTION[0]} = "$SELECTION_DEFAULT" ]]; then
489         toon_fouttekst "Te weinig schijfruimte op $mounted \
490 (bestandssysteem $filesystem)
491 voor het terugzetten van ALLES in $(basename "$BACKUP") naar $TARGET.
492 Overweeg het gebruik van optie SELECTIE.
493 $HELPLINE"
494         quiet; exit $ERROR
495     fi
496     toon_waarschuwingstekst "Mogelijk te weinig schijfruimte (hangt af van \
497 de selectie)
498 op $mounted (bestandssysteem $filesystem)
499 voor het terugzetten van ${SELECTION[*]} in $(basename "$BACKUP") naar \
500 $TARGET).
501 De HELE back-up is ${need_human}iB groot, beschikbaar is ${free_human}iB."
502     fi
503 }
504
505 bepaal_bestandsgrootte() {
506     BACKUPSIZE_HUMAN=$(
507         du --apparent-size \
508            --block-size=1 \
509            --human-readable \
510            "$BACKUP" |
511         awk '{print $1}'
512     )
513
514     BACKUPSIZE_MACHINE=$(
515         du --apparent-size \
516            --block-size=1 \
517            "$BACKUP" |
518         awk '{print $1}'
519     )
520 }
521
522 zet_backup_terug() {
523     set +o errexit
524     if $OPTION_GUI; then
525         (
526             pv --size="$BACKUPSIZE_MACHINE" \
527                --numeric \
528                < "$BACKUP" |
529             tar --extract \
530                --directory="$TARGET" \
531                --exclude-from="$EXCLUDEFILE_DEFAULT" \
532                --exclude-from="$EXCLUDEFILE_OPTIONAL" \
533                --file=- \
534                "${SELECTION[@]}" \
535                2> /dev/null
536         ) 2>&1 |
537         zenity --progress \
538                --width=600 \
539                --height=50 \
540                --title="$title" \
541                --text="$select$text" \
542                --auto-close \
543                --no-cancel \
544                --time-remaining \
545                2> /dev/null
546         TAR_RC=$?
547     else
548         toon_tekst "$select$text ..."
549         pv --size="$BACKUPSIZE_MACHINE" \
550            < "$BACKUP" |
551         tar --extract \
552            --directory="$TARGET" \

```



```
553         --exclude-from="$EXCLUDEFILE_DEFAULT" \
554         --exclude-from="$EXCLUDEFILE_OPTIONAL" \
555         --file=- \
556         "${SELECTION[@]}" \
557         2> /dev/null
558     TAR_RC=$?
559     fi
560     if ! $ARGUMENT_SELECTION && [[ $TARGET = '/' ]]; then
561         zet_backup_favorieten_terug
562     fi
563     set -o errexit
564     sync
565 }
566
567 zet_backup_favorieten_terug() {
568     local user=''
569
570     for dir in /home/*; do
571         user=$(basename "$dir")
572         sudo su --login "$user" \
573             --command "gsettings set org.gnome.shell favorite-apps < \
574 /home/$user/.backup-gnome-favorites" |& $LOGCMD
575     done
576 }
577
578 term_script() {
579     if [[ $TAR_RC -eq 0 ]]; then
580         toon_succetekst "De back-up is teruggezet."
581         quiet; exit $SUCCESS
582     else
583         toon_waarschuwingstekst "De back-up is niet of slechts gedeeltelijk \
584 teruggezet."
585         quiet; exit $WARNING
586     fi
587 }
588
589 verwerk_script() {
590     controleer_invoer "$@"
591     verwerk_invoer
592 }
593
594 # #####
595 # Hoofdlijn
596 # #####
597 main() {
598     init_script
599     verwerk_script "$@"
600     term_script
601 }
602
603 main "$@"
604
605 # Einde script
```