

```
1: #!/bin/bash
2: # shellcheck source=kz-common.sh
3: #####
4: # Back-up maken.
5: #
6: # Geschreven door Karel Zimmer <info@karelzimmer.nl>.
7: #####
8: PROGRAM_PATH=$(realpath "$(dirname "$0")")
9: source "$PROGRAM_PATH"/kz-common.sh
10: PROGRAM_NAME=kz-backup
11: DISPLAY_NAME=${PROGRAM_NAME/kz-/kz }
12: RELEASE_YEAR=2007
13:
14: VERSION_NUMBER=36.01.01
15: VERSION_DATE=2021-10-07
16:
17:
18: #####
19: # Global constants
20: #####
21:
22: readonly RUN_AS_SUPERUSER=true
23: readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON'e:t:'
24: readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON',exclude:,target:'
25: readonly USAGE="Gebruik: $DISPLAY_NAME [-e|--exclude=UITSLUITEN]... \
26: [-t|--target=DOELMAP]
27:                 $OPTIONS_USAGE_COMMON
28:                 [SELECTIE...]"
29: readonly HELP="Gebruik: $DISPLAY_NAME [OPTIE...] [SELECTIE...]"
30:
31: Back-up maken.
32:
33: Opties:
34:   -e --exclude=UITSLUITEN
35:                 sluit opgegeven bestand of map uit
36:   -t --target=DOELMAP
37:                 plaats back-up in de DOELMAP
38: $OPTIONS_HELP_COMMON
39:
40: Argumenten:
41:   SELECTIE      maak back-up van opgegeven mappen en bestanden"
42:
43: readonly EXCLUDEFILE_DEFAULT=/usr/local/etc/$PROGRAM_NAME-exclude-dflt
44: readonly EXCLUDEFILE_DEFAULT_CONTENT="\
45: /dev
46: /home/*/.adobe/Flash_Player/AssetCache
47: /home/*/.cache
48: /home/*/.ccache
49: /home/*/.gvfs
50: /home/*/.Private
51: /home/*/.recent-applications.xbel
52: /home/*/.recently-used.xbel
53: /home/*/snap/*/*/.cache
54: /home/*/.steam/root
55: /home/*/.thumbnails
56: /home/*/.var/app/*/cache
57: /home/*/.xsession-errors/proc
58: /run
59: /sys
60: /tmp"
61: readonly EXCLUDEFILE_OPTIONAL=/usr/local/etc/$PROGRAM_NAME-exclude-opt
62: readonly TARGET_DEFAULT_1=/media
63: readonly TARGET_DEFAULT_2=$PROGRAM_NAME
64: readonly TARGET_DEFAULT_3=$PROGRAM_NAME-$HOSTNAME-$DISTRO
65: # Conform ISO 8601, but ':' -> '.' for other filesystems not on Linux.
66: TIMESTAMP=$(date +%Y-%m-%dT%H.%M.%S); readonly TIMESTAMP
67:
68:
```

```
69: #####
70: # Global variables
71: #####
72:
73: declare -a EXCLUDE_ARGUMENT=()
74: declare ARGUMENT_SELECTION=false
75: declare -a SELECTION=()
76: declare -a SELECTION_ARGUMENT=()
77: declare -a SELECTION_DEFAULT=(' /home' )
78: declare BACKUP=''
79: declare BACKUP_CREATED=false
80: declare BACKUPFILE=''
81: declare BACKUP_TO_DELETE=''
82: declare -i SELECTIONSIZE_MACHINE=0
83: declare OPTION_EXCLUDE=false
84: declare OPTION_TARGET=false
85: declare SELECTIONSIZE_HUMAN=''
86: declare SPACE_OK=true
87: declare TARGET_ARGUMENT=''
88: declare TARGET='/mnt'
89:
90:
91: #####
92: # Functions
93: #####
94:
95: check_input() {
96:     local exclude=''
97:     local -i exclude_arg_num=0
98:     local -i select_arg_num=0
99:     local -i select_num=0
100:    local select=''
101:    local -i getopt_rc=0
102:    local parsed=''
103:
104:    parsed=$(
105:        getopt --alternative \
106:               --options "$OPTIONS_SHORT" \
107:               --longoptions "$OPTIONS_LONG" \
108:               --name "$DISPLAY_NAME" \
109:               -- "$@"
110:    ) || getopt_rc=$?
111:    if [[ $getopt_rc -ne $SUCCESS ]]; then
112:        printf '%s\n' "$USAGELINE" >&2
113:        exit $ERROR
114:    fi
115:    eval set -- "$parsed"
116:    process_common_options "$@"
117:
118:    while true; do
119:        case $1 in
120:            -e|--exclude)
121:                OPTION_EXCLUDE=true
122:                EXCLUDE_ARGUMENT[$exclude_arg_num]=$2
123:                ((++exclude_arg_num))
124:                shift 2
125:                ;;
126:            -t|--target)
127:                if $OPTION_TARGET; then
128:                    TEXT="optie '$1' eenmaal opgeven"
129:                    printf "$DISPLAY_NAME: %s\n%s\n" \
130:                           "$TEXT" \
131:                           "$USAGELINE" >&2
132:                    exit $ERROR
133:                else
134:                    OPTION_TARGET=true
135:                    TARGET_ARGUMENT=$2
136:                fi
137:            *)
138:                break
139:            esac
140:        done
141:    done
142:}
```

```
137:         shift 2
138:         ;;
139:     --)
140:         shift
141:         break
142:         ;;
143:     *)
144:         shift
145:         ;;
146:     esac
147: done
148:
149: if $OPTION_EXCLUDE; then
150:     for exclude in "${EXCLUDE_ARGUMENT[@]"; do
151:         if ! [[ -e $exclude ]]; then
152:             TEXT="exclude bestand of map '$exclude' bestaat niet"
153:             printf "$DISPLAY_NAME: %s\n%s\n" \
154:                 "$TEXT" \
155:                 "$USAGELINE" >&2
156:             exit $ERROR
157:         fi
158:     done
159: fi
160:
161: while [[ "$*" ]]; do
162:     ARGUMENT_SELECTION=true
163:     SELECTION_ARGUMENT[$select_arg_num]=$1
164:     ((++select_arg_num))
165:     shift
166: done
167:
168: check_user
169:
170: # Na check_user voor toegang tot media.
171: if $ARGUMENT_SELECTION; then
172:     for select in "${SELECTION_ARGUMENT[@]"; do
173:         if ! [[ -e $select ]]; then
174:             TEXT="selectie bestand of map '$select' bestaat niet"
175:             printf "$DISPLAY_NAME: %s\n%s\n" \
176:                 "$TEXT" \
177:                 "$USAGELINE" >&2
178:             exit $ERROR
179:         fi
180:     done
181:     for select_num in "${!SELECTION_ARGUMENT[@]"; do
182:         SELECTION[$select_num]=$ (
183:             readlink --canonicalize "${SELECTION_ARGUMENT[$select_num]}"
184:         )
185:     done
186: else
187:     for select_num in "${!SELECTION_DEFAULT[@]"; do
188:         SELECTION[$select_num]="${SELECTION_DEFAULT[$select_num]}"
189:     done
190: fi
191:
192: request_input
193: }
194:
195:
196: request_input() {
197:     local -A media=()
198:     local media2=false
199:     local media_found=false
200:     local medium=''
201:
202:     if $OPTION_TARGET; then
203:         TARGET=$TARGET_ARGUMENT
204:         media_found=true
```

```
205: else
206:     # Vul associatief array met aangekoppelde media-namen.
207:     while read -r medium; do
208:         if $media_found; then
209:             # Meer dan Ã©Ã©n aangekoppeld medium gevonden.
210:             media2=true
211:         fi
212:         media_found=true
213:         media[$medium]=$medium
214:         TARGET=$medium/$TARGET_DEFAULT_2/$TARGET_DEFAULT_3
215:     done <<(
216:         # findmnt wordt door mount aangeraden, speciaal voor in scripts, en
217:         # options=rw om alleen beschrijfbare bestandssystemen gebruiken (en
218:         # bijvoorbeeld geen iso9660; d.i. een CD of DVD).
219:         findmnt --list \
220:             --noheadings \
221:             --options=rw \
222:             --output=TARGET |
223:         grep --word-regexp \
224:             --regexp=$TARGET_DEFAULT_1 || true
225:     )
226: fi
227: if $media2; then
228:     # Meer dan Ã©Ã©n aangekoppeld medium gevonden.
229:     media_found=false
230: fi
231: if $media_found; then
232:     if ! [[ -e $TARGET ]]; then
233:         mkdir --parents \
234:             "$TARGET" |& $LOGCMD
235:         chown 1000:1000 \
236:             "$TARGET" |& $LOGCMD
237:     fi
238: else
239:     if $OPTION_GUI; then
240:         request_input_gui
241:     else
242:         request_input_tui
243:     fi
244: fi
245:
246: BACKUPFILE=$PROGRAM_NAME-$HOSTNAME-$DISTRO-$TIMESTAMP.tar
247: BACKUP=$TARGET/$BACKUPFILE
248: }
249:
250:
251: request_input_gui() {
252:     local -i dir_selected=0
253:
254:     TITLE='Kies een map om de back-up in te plaatsen'
255:     # Constructie '2> >($LOGCMD)' om stderr naar de log te krijgen.
256:     # Voorbeeld: Unable to init server: Kon niet verbinden:
257:     #     Verbinding is geweigerd
258:     #     en: (zenity:47712): Gtk-WARNING **: 10:35:49.339:
259:     #     cannot open display:
260:     TARGET="$(
261:         zenity --file-selection \
262:             --width 600 \
263:             --height 50 \
264:             --title "$TITLE" \
265:             --filename "$TARGET" \
266:             --directory 2> >($LOGCMD)
267:     )" || dir_selected=$?
268:     if [[ $dir_selected -ne $SUCCESS ]]; then
269:         warning 'Geen map geselecteerd.'
270:
271:     Geen back-up gemaakt!'
272:     exit $WARNING
```

```
273:     fi
274: }
275:
276:
277: request_input_tui() {
278:     declare medium=''
279:
280:     TEXT="Gebruik optie target om op te geven waar de back-up geplaatst moet \
281: worden."
282:     printf '%s\n' "$TEXT" >&2
283:     if $media2; then
284:         # Meer dan Ã©Ã©n aangekoppeld medium gevonden.
285:         printf '%s\n' "Voeg Ã©Ã©n van de volgende opties toe voor de nu \
286: aangekoppelde media:"
287:         for medium in "${!media[@]}"; do
288:             printf "    ${BLUE}--target %s\n${NORMAL}" \
289:                 "${media[$medium]}/${TARGET_DEFAULT_2}/${TARGET_DEFAULT_3}"
290:         done
291:     else
292:         printf '%s\n' "Voorbeeld: $DISPLAY_NAME --target DOELMAP"
293:     fi
294:     printf '%s\n' "$USAGELINE" >&2
295:     exit $SUCCESS
296: }
297:
298:
299: process_input() {
300:     local size_num=0
301:     local size_unit=''
302:
303:     printf '%s\n' "$EXCLUDEFILE_DEFAULT_CONTENT" > "$EXCLUDEFILE_DEFAULT"
304:
305:     if $OPTION_EXCLUDE; then
306:         for exclude in "${EXCLUDE_ARGUMENT[@]}"; do
307:             printf '%s\n' "$exclude" >> "$EXCLUDEFILE_OPTIONAL"
308:         done
309:     else
310:         printf '\n' > "$EXCLUDEFILE_OPTIONAL"
311:     fi
312:
313:     TEXT='Bepaal grootte van de back-up (kan even duren)...'
314:     if $OPTION_GUI; then
315:         TITLE='Back-up maken'
316:         # Met "|& zenity --progress" worden globale variabelen uit
317:         # aangeroepen functies niet doorgegeven, vandaar de
318:         # 'process substitution' met "> >(zenity...)".
319:         determine_selection_size >>(
320:             zenity --progress          \
321:                   --pulsate           \
322:                   --auto-close        \
323:                   --no-cancel         \
324:                   --width 600         \
325:                   --height 50         \
326:                   --title "$TITLE"   \
327:                   --text "$TEXT"     2> >($LOGCMD)
328:         )
329:     else
330:         info "$TEXT"
331:         determine_selection_size
332:     fi
333:
334:     TEXT='Controleer beschikbare schijfruimte (kan even duren)...'
335:     if $OPTION_GUI; then
336:         check_space >>(
337:             zenity --progress          \
338:                   --pulsate           \
339:                   --auto-close        \
340:                   --no-cancel         \
```

```
341:          --width      600      \
342:          --height     50        \
343:          --title      "$TITLE"  \
344:          --text       "$TEXT"   2> >($LOGCMD)
345:      )
346:  else
347:      info "$TEXT"
348:      check_space
349:  fi
350:
351:  if $SPACE_OK; then
352:      size_num=${SELECTIONSIZE_HUMAN%?}
353:      size_unit=${SELECTIONSIZE_HUMAN: -1}
354:      TEXT="Backup van ${SELECTION[*]} [${size_num} ${size_unit}iB] naar \
355: $TARGET/$BACKUPFILE maken..."
356:      check_on_ac_power
357:      create_backup
358:  fi
359:
360:  if ! $BACKUP_CREATED; then
361:      warning 'Geen back-up gemaakt!'
362:      exit $WARNING
363:  fi
364: }
365:
366:
367: determine_selection_size() {
368:     SELECTIONSIZE_HUMAN=$(
369:         du --apparent-size          \
370:            --block-size=1          \
371:            --exclude-from="$EXCLUDEFILE_DEFAULT" \
372:            --exclude-from="$EXCLUDEFILE_OPTIONAL" \
373:            --human-readable        \
374:            --summarize              \
375:            --total                  \
376:            "${SELECTION[@]}"        \
377:            2> >($LOGCMD)           |
378:         awk 'END{print $1}'
379:     )
380:
381:     SELECTIONSIZE_MACHINE=$(
382:         du --apparent-size          \
383:            --block-size=1          \
384:            --exclude-from="$EXCLUDEFILE_DEFAULT" \
385:            --exclude-from="$EXCLUDEFILE_OPTIONAL" \
386:            --summarize              \
387:            --total                  \
388:            "${SELECTION[@]}"        \
389:            2> >($LOGCMD)           |
390:         awk 'END{print $1}'
391:     )
392: }
393:
394:
395: check_space() {
396:     local filesys=''
397:     local free_human=''
398:     local -i free_bytes=0
399:     local -i sourcedir_size_bytes=0
400:     local mounted=''
401:
402:     sourcedir_size_bytes=$(
403:         du --apparent-size          \
404:            --block-size=1          \
405:            --exclude-from="$EXCLUDEFILE_DEFAULT" \
406:            --exclude-from="$EXCLUDEFILE_OPTIONAL" \
407:            --summarize              \
408:            --total                  \
```

```

409:         "${SELECTION[@]}" \
410:         2> >($LOGCMD) |
411:         awk 'END{print $1}'
412:     )
413:     free_bytes=$(
414:         df --block-size=1 \
415:           "$TARGET" |
416:           awk 'END{print $4}'
417:     )
418:
419:     if [[ $sourcedir_size_bytes -gt $free_bytes ]]; then
420:         free_human=$(
421:             df --human-readable \
422:               "$TARGET" |
423:               awk 'END{print $4}'
424:         )
425:         filesys=$(
426:             df "$TARGET" |
427:             awk 'END{print $1}'
428:         )
429:         mounted=$(
430:             df "$TARGET" |
431:             awk 'END{print $6}'
432:         )
433:         warning "
434: Kan geen back-up plaatsen.
435:
436: Te weinig schijfruimte op $mounted (bestandssysteem $filesys)
437: voor het plaatsen van back-up $BACKUPFILE.
438:
439: Benodigd is ${SELECTIONSIZE_HUMAN}iB, beschikbaar is ${free_human}iB.
440:
441: Maak ruimte vrij op $mounted (bestandssysteem $filesys),
442: of gebruik een ander medium met minimaal ${SELECTIONSIZE_HUMAN}iB beschikbaar
443: zoals een USB-stick of externe harddisk."
444:         SPACE_OK=false
445:     else
446:         SPACE_OK=true
447:     fi
448: }
449:
450:
451: create_backup() {
452:     local -i size_num=0
453:     local -i tar_rc=0
454:     local -i sync_rc=0
455:     local size_unit=""
456:
457:     BACKUP_TO_DELETE=$BACKUP
458:
459:     if $OPTION_GUI; then
460:         tar --create \
461:           --directory=/ \
462:           --exclude-from="$EXCLUDEFILE_DEFAULT" \
463:           --exclude-from="$EXCLUDEFILE_OPTIONAL" \
464:           --file=- \
465:           "${SELECTION[@]}" \
466:           2> >($LOGCMD) |
467:         (
468:             pv --numeric \
469:               --size="$SELECTIONSIZE_MACHINE" \
470:               > "$BACKUP"
471:         ) 2>&1 |
472:         zenity --progress \
473:           --auto-close \
474:           --no-cancel \
475:           --time-remaining \
476:           --width 600 \

```

```

477:         --height      50                \
478:         --title       "$TITLE"          \
479:         --text        "$TEXT"           2> >($LOGCMD) || tar_rc=$?
480:     else
481:         info "$TEXT"
482:         tar --create                \
483:             --directory=/          \
484:             --exclude-from="$EXCLUDEFILE_DEFAULT" \
485:             --exclude-from="$EXCLUDEFILE_OPTIONAL" \
486:             --file=-                \
487:             "${SELECTION[@]}"       \
488:             2> >($LOGCMD)           |
489:         pv --size="$SELECTIONSIZE_MACHINE" \
490:         > "$BACKUP" || tar_rc=$?
491:     fi
492:
493:     TEXT='Gegevens uit tijdelijk geheugen wegschrijven (kan even duren)...'
494:     if $OPTION_GUI; then
495:         sync |
496:         zenity --progress            \
497:               --pulsate             \
498:               --auto-close          \
499:               --no-cancel           \
500:               --width      600      \
501:               --height    50       \
502:               --title     "$TITLE" \
503:               --text      "$TEXT"   2> >($LOGCMD) || sync_rc=$?
504:     else
505:         info "$TEXT"
506:         sync || sync_rc=$?
507:     fi
508:
509:     # tar_rc=0 Succesvolle beÃ«indiging.
510:     # tar_rc=1 Sommige bestanden verschillen. Deze afsluitcode betekent dat
511:     # sommige bestanden zijn gewijzigd tijdens het archiveren en dus bevat
512:     # het resulterende archief niet de exacte kopie van de bestandsset.
513:     # tar_rc=2 Fatale fout. Dit betekent dat er een fatale, onherstelbare fout
514:     # is opgetreden.
515:     if [[ $tar_rc -eq 1 ]]; then
516:         log "Sommige bestanden zijn gewijzigd tijdens het maken van de \
517: back-up en deze wijzigingen zijn niet opgenomen in de gemaakte back-up."
518:     fi
519:     if [[ ($tar_rc -ne $SUCCESS && $tar_rc -ne 1) || $sync_rc -ne $SUCCESS ]]; then
520:         then
521:             error "Back-up maken is mislukt."
522:
523:     Opdracht om de log uit te lezen:
524:     ${BLUE}$LOGCMD_CHECK${NORMAL}
525:
526:     Geen back-up gemaakt!"
527:     exit $ERROR
528:     fi
529:
530:     BACKUP_TO_DELETE=''
531:     BACKUP_CREATED=true
532: }
533:
534:
535: term_script() {
536:     TEXT='De back-up is gemaakt.'
537:     if [[ "$TARGET" = /media/* ]]; then
538:         TEXT="$TEXT
539:
540: Koppel zelf (veilig!) het back-up-medium af."
541:     fi
542:     info "$TEXT"
543:     exit $SUCCESS
544: }

```



```
545:
546:
547: #####
548: # Main line
549: #####
550:
551: main() {
552:     init_script "$@"
553:     check_input "$@"
554:     process_input
555:     term_script
556: }
557:
558:
559: main "$@"
560:
561:
562: # EOF
```