

How to rearrange the channel list

To rearrange the channel list, use this shell script:

```
#!/bin/sh
#
# a small shell script for automatic ordering of TV channels in MythTV
#
# BIG FAT WARNING: this script was tested only with MythTV v0.23.
# Future versions of mythtv possibly use an incompatible database layout.
#
# The following workflow is recommended:
# 1) backup the current channel settings:
#     ./channel_sort_mythtv.sh dump >mythtv_channels_dump.sql
# 2) retrieve the current list of channel names
#     ./channel_sort_mythtv.sh get >channels.list
# 3) sort the channels in 'channels.list' with your favourite text editor
# 4) upload the new channel order (starting from channum=1):
#     ./channel_sort_mythtv.sh set <channel.list
#
# After another channel scan you can run this script again with the
# 'merge' argument. It reads your current channel order from stdin
# and appends all 'new' channels (taken from the database).
#
#
# Copyright 2010 Lars Kruse <devel@sumpfralle.de>
#
# This script is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 2 of the License, or
# (at your option) any later version.
#
# This script is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
```

```

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

set -eu

# you can override the mysql.txt location by providing it via an environment variable
MYSQL_CONF_FILE=${MYSQL_CONF_FILE:-}

# only used for "set" operation:
#     use this channel number for all channels, that are listed in stdin
#     set it to an empty string, if the channels, that are not listed, should remain
untouched
DEFAULT_CHANNEL=10000

# set DEBUG to a non-empty string to output the result sql statements instead of executing
them
# use something like "export DEBUG=true" before running this script to enable debug mode
DEBUG=${DEBUG:-false}
export DEBUG

# read the mysql connection settings from the myth config
# this exposes DBHostName, DBUserName, DBName and DBPassword
for mysql_config in "$MYSQL_CONF_FILE" ~/.mythtv/mysql.txt /etc/mythtv/mysql.txt; do
    if test -n "$mysql_config" && test -r "$mysql_config"; then
        . "$mysql_config"
        break
    fi
done
# temporarily disable the "unset variable check" of bash
set +u
if test -z "$DBHostName" -o -z "$DBUserName" -o -z "$DBName" -o -z "$DBPassword"; then
    echo "Failed to read the mysql settings file of MythTV." >&2
    echo "You can set the environment variable MYSQL_CONF_FILE to specify a custom
location." >&2
fi
set -u

send_mysql_statement() {

```

```

if test "$DEBUG" != "false"; then
    # only output the sql statements - don't execute them
    cat -
else
    mysql -h "$DBHostName" -u "$DBUserName" --password="$DBPassword" --skip-
column-names "$DBName"
    fi
}

show_syntax() {
    echo "Syntax: $(basename "$0") ACTION"
    echo "Possible actions:"
    echo "  get      - get all unsorted channel names from the database (to stdout)"
    echo "  set      - read the desired order of channels (by name) from stdin"
    echo "  merge    - output stdin and add all channel names (from the database)"
    echo "             that were not listed before (to stdout)"
    echo "  dump     - dump the current state of the 'channel' table (as a backup)"
    echo "  help     - show this syntax description"
    echo "Hint: the 'get' operation does to same as 'merge' with empty stdin."
}

ACTION=invalid
test $# -gt 0 && ACTION="$1"

case "$ACTION" in
    get)
        # get distinct names and ignore empty lines
        echo "SELECT DISTINCT BINARY name FROM channel ORDER BY LPAD(channum, 3, 0)
ASC;" \
            | send_mysql_statement \
            | grep -v "^$"
        ;;
    set)
        (
            # reset all channel numbers to the default value if it is not empty
            test -n "$DEFAULT_CHANNEL" && echo "update channel set
channum='$DEFAULT_CHANNEL';"
            current_num=1
            cat - | grep -v "^$" | while read channel; do
                # escape single quotes

```

```

        channel=$(echo "$channel" | sed "s/'/\\\\'/g")
        echo "update channel set channum=$current_num where BINARY
name='$channel';"

        current_num=$((current_num + 1))
            done
        ) | send_mysql_statement
    ;;
merge)
    input="$(cat -)"
    echo "$input"
    "$@" get | while read channel; do echo "$input" | grep -q "^$channel$" || echo
"$channel"; done
    ;;
dump)
    mysqldump -h "$DBHostName" -u "$DBUserName" --password="$DBPassword" "$DBName"
channel
    ;;
help|--help|invalid)
    if test "$ACTION" = "invalid"; then
        show_syntax >&2
        exit 1
    else
        show_syntax
    fi
    ;;
esac

exit 0

```

Revision #3

Created 2026-04-01 17:12:54 CEST by Philip

Updated 2026-04-13 19:31:33 CEST by Philip