Notification to all logged in X users
broadcast
#!/bin/bash

# Broadcast v1.0
#
# A sleazy, dirty, creepy hack to broadcast a message to every single connected user within an X environment.
# Feel free to enhance and share any enhancement!
# everchere@everchere.com
#

# Checking the number of parameters given:
if [ $# -ne 3 ]; then echo "Wrong number of parameters:";echo "Usage: $0 TITLE MESSAGE DURATION";exit 1; fi;
# First parameter is the title of the info box
TITLE="$1"
# Second parameter is the message to be broadcasted
MSG="$2"
# Third parameter is the timeout duration for the Info box.
DURATION="$3"

# Null-Initializes the list of ADVertised CLienTs
ADVCLT=""

# Grabs the name of one of the connected users from which to grab a list of connected displays
USERNAME=`w -h | head -1 | cut -d' ' -f1`

# Grabs the list of connected users
USERLIST=`w -h | cut -d' ' -f1 | sort -d -u | grep \[^root\]`

# Adds :0 to the list of the displays, since it might not be grabbed later if $USERNAME is a Network User.
XAUTHOP=":0 

# Grabs the list of the connected displays from $USERNAME and removes duplicates
XAUTHOP="$XAUTHOP 'su $USERNAME -c -l 'xauth list $DISPLAY' | sed 's/.*\([0-9]\+\).*/\1/g' | sort -d -u"

# Core Stuff:
# For each Connected display...
for i in $XAUTHOP
do
  # Try to display the message with each user's environment.
  # That's dirty and kind of "brutal", but I couldn't find a way to associate a display to a user in a clean way...
  for j in $USERLIST
do
    # Try to display for the user only if he was not already advertised.
    if ! (echo "$ADVCLT" | grep $j)
      then
        su $j -c -l "export DISPLAY='localhost$\{i\}'; zenity --info --title='$TITLE' --text='$MSG' --timeout $DURATION"
      fi
    # Same thing here, for non-network users.
    if ! (echo "$ADVCLT" | grep $j)
      then
        su $j -c -l "export DISPLAY='$\{i\}'; zenity --info --title='$TITLE' --text='$MSG' --timeout $DURATION" & & ADVCLT="$ADVCLT $j"
      fi
  done
done