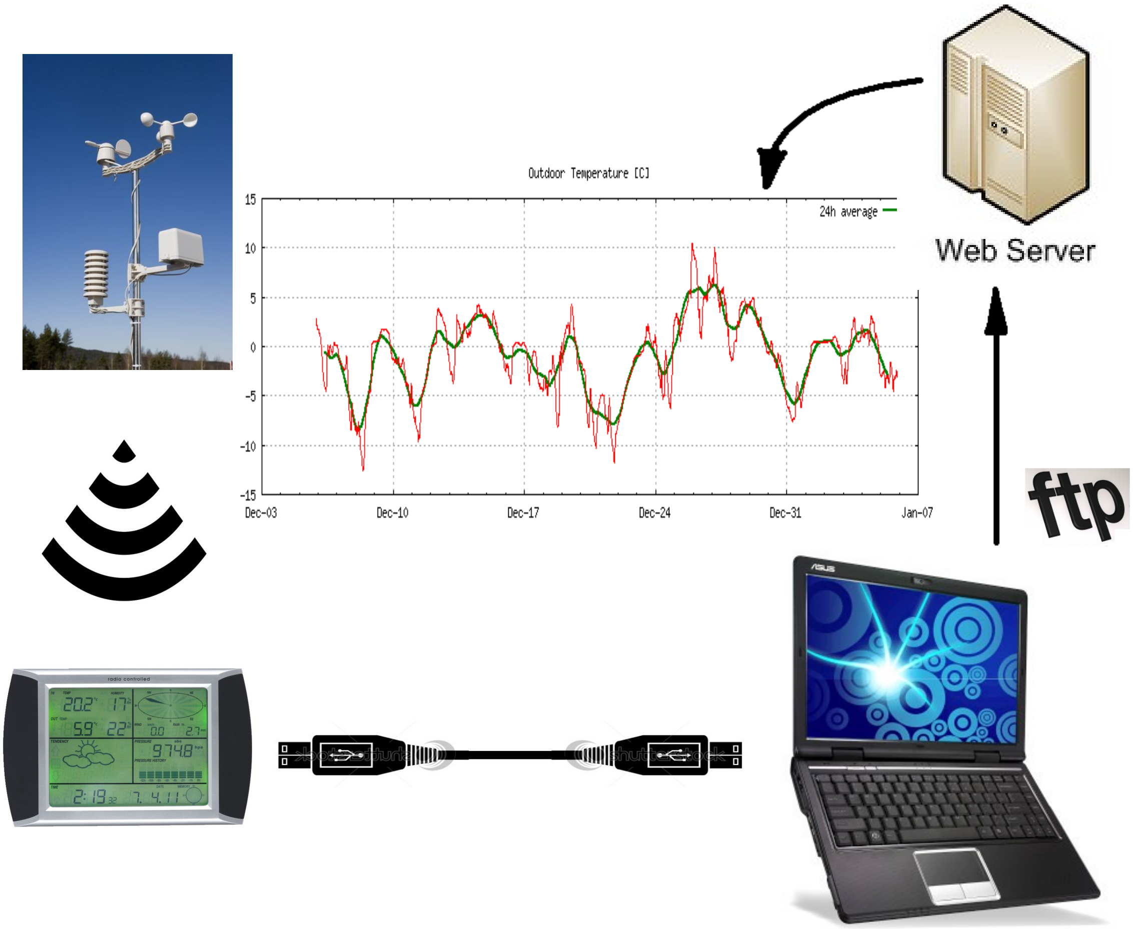
LINUX WEATHER STATION

This procedure explains how to install and use software for WH-1080 weather station on a Linux Kubuntu PC. The software collects weather data from the weather station at regular intervals, stores it in a database and produces plots that are uploaded over FTP to a website for display.

Prerequisites

To use the software you must have the following

- A weather station unit, model WH-1080 (or compatible), incl. indoor receiver with USB PC connection. This unit is available as art.no. 36-3240 at Clas Ohlson in Norway http://www.clasohlson.no/Product/Product.aspx?id=167399209
- A dedicated PC with Linux Kubuntu 11.10 installed (Ubuntu/Xubuntu etc. also ok). Other distros have not been tested, but might be ok.
- Suitable space on a web server (minimal space required), and FTP file upload access (you will need to use your FTP username and password)



Software description

Before discussing how to install the software, we briefly describe how it works, the command line parameters are as follows

```
Usage: wstation [-i] [-l] [-s] [-xd <num>] [-xp <str>] [-xr <str>] [-xh <str>] [-xu] [-co <str>]
<database filename>

  -i, --init_db            Initialise new database
  -l, --latest             Show latest data on screen (1 entry)
  -s, --store              Store latest data to database (1 entry)
  -xd, --xdays=<num>       Export <num> days to standard output
  -xe, --xelev=<str>          Export: Elevation above sea level, <str>=[m]
  -xr, --xrdat=<str>          Export: Rain level datum, <str>=[mm]
  -xh, --xhtml=<str>          Export: latest numeric values to html file, <str>=[filename]
  -xu, --xutc                 Export: use UTC time instead of local time
  -co, --conv=<str>        Convert old data <str>=WStest1
```

Do-once items (manual)

These items are things you must do once to set up the system.

- Prepare data collection on the dedicated PC
  - Configure reading weather data over USB (see below)
  - Extract supplied file structure under *~/Weather,* and edit the files
    - Initialise the database in the *~/Weather* folder, see below
    - edit *script/ws_upload.sh*, values for options -xp and -xr, see below. Make sure the database name is correct.
    - edit *script/ftp_upload.txt. S*et <ftp_user> and <ftp_password>. Edit also the target folder name as required.
- Prepare web server folder
  - create the target folder for weather station upload files. It must be consistent with the specification in the file *script/ftp_upload.txt.ftp*.
  - Copy the supplied *www/index.htm* to the web server folder. This file does never change (unless you create new plotfiles), so it is copied only once.
- Test USB connection and FTP upload
  - Manually execute script/ws_upload.sh, check if ok, see below
- When ok, start automatic acquisition of data
  - Edit crontab file for automatic execution of *ws_upload.sh*, see below

Repeated items (automatic)

The software does the following at regular intervals, e.g. every 5 min:

ws_upload.sh

- wstation
  - reads sensor data (indoor and outdoor) over USB from the indoor unit
  - Each reading, including the time, is stored in an SQLite database
  - Data time series are exported from the database to 3 text files: ws_day.txt, ws_week, ws_month.txt
  - Also exported is latest.htm, which contains numeric values of latest data
- gnuplot
  - produces png plot files of weather station data, by reading the gnuplot-ws.txt input file, which refers to the ws_day.txt, ws_week, ws_month.txt files.
- ftp
  - uploads *.png and latest.htm to a web server folder dedicated to showing weather data. The folder is typically called "wstation" and contains already index.htm referring to the named *.png and latest.htm.

Configure reading weather data over USB

First, do NOT connect the PC and the indoor unit yet, wait until you are told to do so :-) Instead, do as follows:

1. Install rules file from *~/Weather/usb* folder

   ```
   $ sudo cp  55-meteo.rules    /etc/udev/rules.d
   ```

2. Set user access permission
   The rules file declares the weather station to belong to the 'video' group. To allow your user to access the weather station device, the user must be added to the 'video' group, see also https://help.ubuntu.com/community/AddUsersHowto#Graphical_Kubuntu

   ```
   $ sudo adduser <username> video
   ```

3. Reboot the computer

4. Check usb connection
   Connect the USB cable between weather station indoor unit and the PC, and then type:

   ```
   $ dmesg
   ```

   (... lots of output omitted)

   The last line should be something like, observe the numbers in **red**.

   ```
   [ 1038.217354] generic-usb 0003:1941:8021.0005:
   hiddev0,hidraw0: USB HID v1.00 Device [HID 1941:8021] on usb-
   0000:00:1d.0-2/input0
   ```

5. The last output line is typically the last USB unit connected. Observe [HID 1941:8021], which is the vendor id and product id the software will be looking for. If you see these exact numbers, you are good to go!

Edit files under ~/Weather

The folder structure on the weather station PC should become

```
~/Weather /doc
          /www
          /script
          /usb
```

The program executable "`wstation`" is in the main folder.

Initialise the database

Open a terminal in that folder and initialise the database:

```
$ ./wstation weather_station.db -i
```

Edit *ws_upload.sh*

Go to the *script* subfolder and edit *ws_upload.sh.* Make sure the database name referred is the same as used when initialising above. Also set the -xe and -xr option values

-xe=152	This option is for specifying altitude adjustments of raw pressure down to sea level pressure. The value 152 shown here corresponds to an elevation of 152m of the weather station. The raw pressure is adjusted according to temperature and elevation, using the Barometric "equation 2" http://en.wikipedia.org/wiki/Barometric_formula
-xr=36.3	This option is for specifying rain level datum [mm] You can ignore this option if you don't have any earlier values in the database, i.e. you only need it if the database originally was created with the now obsolete WScollect. WScollect was unable to read rain data. The value to be specified is the last rain sensor level (in [mm]) **before** registration of rain levels began **in the database**. The value is subtracted in rain intensity calculations, thus avoiding artificial large jumps at the start of the rain intensity series. This value does not affect the accumulated rain series itself (orain), so the datum value can be found by producing a test time series export for the time around installation (option -xd). The datum is then the first non-zero value in the **orain** column. If the value is not specified correctly, the initial rain plot can become odd-looking and scaled wrongly. Eventually, it will still adjust itself nevertheless.

Edit *ftp_upload.txt*

Go to the *script* subfolder and edit *ftp_upload.txt,* first 2 lines. Set the website FTP address and also your username and passoword

```
open www.<website>.org
user <ftp_user>  <ftp_password>
```

Observe that FTP passive mode has been set in the input (command "passive"). This may or may not be good for your configuration, but passive mode is supposed to work around firewall problems. See http://www.ncftp.com/ncftpd/doc/misc/ftp_and_firewalls.html

More on active vs. passive mode FTP http://slacksite.com/other/ftp.html

Notice also the command

```
cd www/wstation
```

This command specifies the path on the web server to the weather station data after logging in via FTP. Edit as required.

Prepare web server folder

Using FTP or other means, create the folder on the web server corresponding to the specification in the *ftp_upload.txt* file ('cd' command).

Edit *index.htm* and change "My Weather Station" to your own name (two places). Copy the supplied *index.htm* to this folder.

Test USB connection and FTP upload

You are now be ready to test the USB connection and the FTP upload.

```
$ ./wstation weather_station.db -l
```

The output should be similar to:

```
PC time now:
        tstmp:           1325792028
        UTC:             05-Jan-2012 19:33:48
        local:           05-Jan-2012 20:33:48

Latest weather station data:

        tstmp:           1325791980             timestamp
        UTC:             05-Jan-2012 19:33:00   Coordinated Universal Time
        local:           05-Jan-2012 20:33:00   Local Time
        itemp:             20.1 C               Indoor temperature
        ihumi:             28.0 %               Indoor humidity
        otemp:             -2.5 C               Outdoor temperature
        ohumi:             72.0 %               Outdoor humidity
        opres:            966.6 hPa             Pressure
        owspd:              0.0 m/s             Average wind speed
        owgus:              0.0 m/s             Gust wind speed
        owdir:            315.0 deg             Wind dir in degrees, clockwise from North
        orain:             48.9 mm              Rain gauge
        osens:               Ok                 Outside sensor contact
```

Collect data once from weather station to database. In main folder, do

```
$ ./wstation weather_station.db -s
```

Successful run should give no terminal feedback. Next, list the data from the newly initialised database:

```
$ ./wstation weather_station.db -xd=1
```

The output should be similar to (line wraps here)

```
# wstation Gnuplot export file. UTC offset used [sec] = 3600. Pressure offset =
0

#   timestamp          itemp[C]        ihumi[%]        otemp[C]        ohumi[%]
opres[mbar]      owspd[m/s]      owgus[m/s]      owdir[deg]       orain[mm]
irain1[mm/h] irain24[mm/24h]    timestamp24h     itemp24h[C]      otemp24[C]
osens[1/0] Time

     1325795520            20.1              28             -2.5              72
966.6              0                0                0            48.9
0                0      1325795520              -nan            -nan
1 20120105 20:32:00
```

If all is well, try to run a complete data collection/upload step. Go to the script subfolder and type:

```
$ ./ws_upload.sh
```

Observe the messages and check the contents of local folder *~/Weather/www* and the target web server folder. Try viewing the plots on the web server with your web browser. If you see the recently created files (notice the time), then you are ready to enable automatic exectution.

Edit crontab file for automatic execution

`cron` is a utility for scheduling jobs under Linux, similar to Scheduled Tasks on Windows. Jobs are controlled by editing the so called "crontab" file.

Crontab – Quick Reference: http://adminschoice.com/crontab-quick-reference

To edit the crontab file in a terminal:

```
$ crontab -e
```

This opens a small editor (nano). Add the following 2 lines at bottom

```
MAILTO=""

*/5 * * * * /home/<username>/Weather/script/ws_upload.sh
```

exit the editor with Ctrl-X

The first line prevents `cron` from emailing you whenever the job is executed. This could otherwise fill up your disk with useless emails, causing everything to stop.

The second line specifies that the `ws_upload.sh` script is to be executed every 5 minutes.

Now everything should be done, watch and enjoy!