Weather-station 気象観測所

maXbox Starter 147 - Get a Weatherbox.

"Non ridere, non lugere, neque detestari, sed intelligere.[1] - Spinoza.

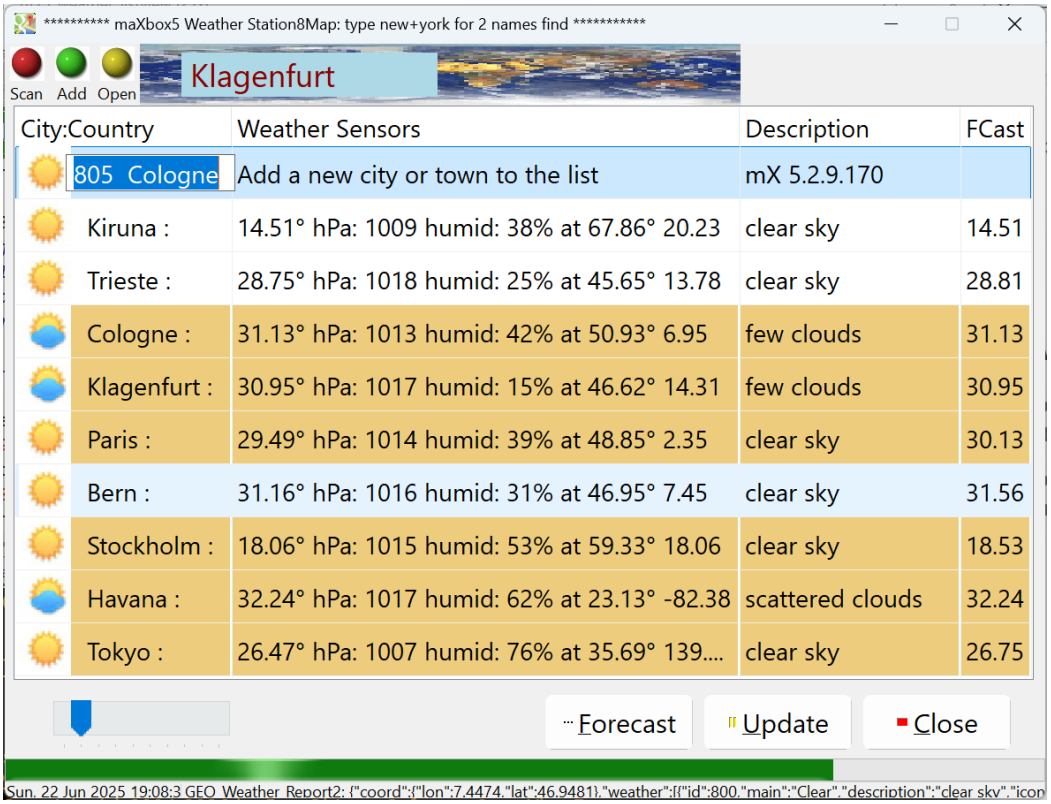
Source: **1413_services5jcl_1.pas**
1415_weather_listview52.txt

https://sourceforge.net/projects/maxbox5/files/examples/1415_weather_listview52.txt/download

The Open Weather Station App (Windows only as Weatherbox) is ready for use for free just to load as a script in maXbox.

The information generated by the OWS every REST-call is the following:

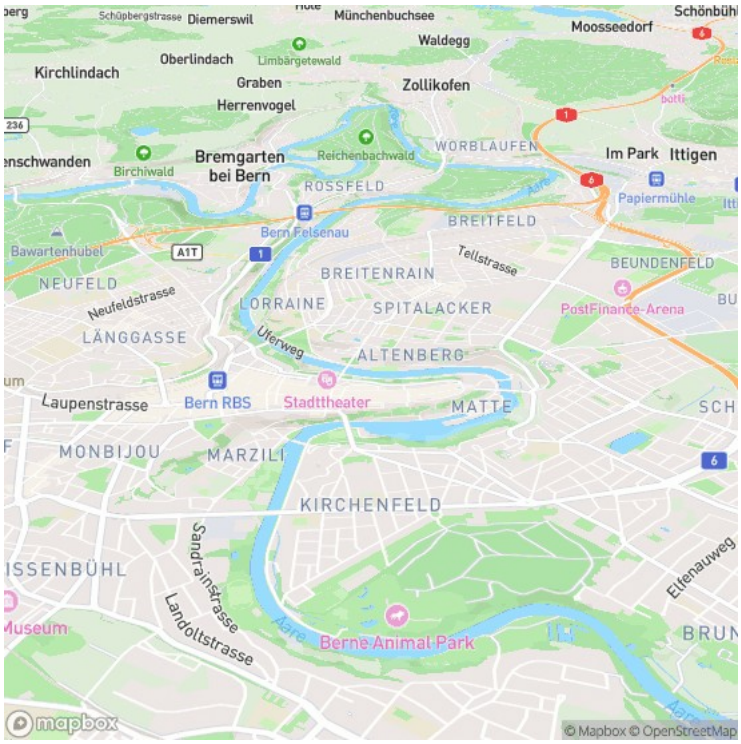
- Location
- temperature (C°)
- absolute atmospheric pressure (Pascal)
- relative humidity (%)
- coordinates (GPS notation)
- description
- Forecast (FCast)



1415_weatherboxScreenshot2025-06-22_190916.png

1 (ethik. 9, 13) - Not to laugh, not to cry, not to hate, but to understand. -

As an addition you can *dblclick* on a location in the listview and you get a map of the environment.
The current implementation is my personal approach to the challenges I have faced during several years of dealing with a lot of unexpected scenarios in the field of arduino sensors as well as REST-Apis like **OpenWeatherMap**, Thingspeak and **mapbox**.



Mapbox API integration

So we use two REST services OpenWeatherMap, mapbox (OpenStreetMap) or another custom cloud service of your preference.
Note: the API Keys delivered with the script are for demo purpose only, please register your own free Key!

```
//breitschbox key https://home.openweathermap.org/api_keys
  UrlWeatherReport25_Forecast=
     'http://api.openweathermap.org/data/2.5/forecast?q=%s&units=metric&APPID'+
                                              '=your own API Key';

{@To get access weather API you need own API key whatever account you chose!!}
https://home.openweathermap.org/api_keys
 {
  "message": "Not Authorized",
  "error_detail": "Direct access not allowed"
  }
```

For commercial use OpenWeather provides hyperlocal minute-by-minute forecasts, historical data, current conditions, and weather forecasts ranging from short-term to annual for any location worldwide. All data is accessible via industry-standard APIs.
OpenWeather delivers also reliable forecasts worldwide, covering both remote and densely populated areas.
For additional functionality, please consider the generous professional product collections.

In maXbox we first call the API and then extract the data with the help of a Regex, Json and simple copy routines:

```
const WeatherREX =
   // kairo test ---> Al 'Atabah  ----> [\w\s']
   '"main":"([\w\s]+)".*"description":"([\w\s]+)".*"temp":([\d\.]+).*'+
   '"temp_max":([\d\.\-]+).*"pressure":([0-9]+).*"humidity":([0-
9]+).*"name":"([\w]+)"';


function GetGeoWeather(const location: string;
                       const UrlRestAPI: string): string;
  var  lStream: TStringStream;  asyn: TSynwInfo;
    //dl: TDownloadURL;
begin
  lStream:= TStringStream.Create('');
  try
     try
       HTTP_GetStream(Format(UrlRestAPI,[location]),lStream);
     except
       //if something wrong try using a backup server.
       //writeln('html back: '+GetURLAsString('http://api.openweathermap.org'));
       writeln('OWeather_Map Exception: '+Gethtm(UrlWeatherReport25))
     end;
     lStream.Seek(0,0);
     result:= 'GEO_Weather_Report2: '+(lStream.ReadString(lStream.Size));
  finally
    lStream.Free;
  end;
end;
```

Note: when no internet is available you get:
Dec: without internet fail safe
debug: 332- 4294967295 err:0
debug: 333-Socket Error # 11001
Host not found. 856 err:20

Bring location data to life with beautiful base maps, versatile upload and design tools and cross-platform rendering. **Mapbox** helps build engaging maps and scripts that delight users and bring them back for more.typo or a tricky logic error, knowing how to debug efficiently is an important skill.

```
function GetGeoInfoMap5save(const lat,lon, zoom: double; asize: integer;
                            const UrlGeoLookupInfo, apath: string): string;
  var
    pngStream: TMemoryStream;
    internalzoomf: double;
    MainMenu1: TMainMenu;
    File1: TMenuItem;
  begin
    pngStream:= TMemoryStream.Create;
      try
        HttpGet(Format(UrlGeoLookupInfo,[lon,lat,zoom, asize-50,asize-50]),
                       pngStream);
      except
        //lHTTP.Get1(Format(UrlGeoLookupInfo2,[IpAddress]), lStream);
        writeln(ExceptionToString(ExceptionType, ExceptionParam));
      end;
```

```
    try
      writ('size of geosat pic '+itoa(pngstream.size));
      pngStream.Position:= 0;
      pngStream.SaveToFile(apath);
      APATHGLOB:= apath;
    finally
     //Dispose;
     //Free;
     pngStream.Free;
  end;
end;
```

The program then runs until it encounters a stopping condition, such as hitting a breakpoint set with **F5**. In which case, the DE sends a breakpoint event to the debug session. The breakpoint event is a stopping event, and the DE again waits for a user response.
When you set a breakpoint with F5 (you can set more than one) then you start the debug session with Debug Run till the breakpoint it waits, then you step further with Debug Run or **Ctrl+ F9** like continue. If the debug session is to ignore a particular stopping event, the debug session calls the program's Continue method. If the program was stepping into, over, or out of a function when it encountered the stopping condition, then it continues the step.

Also set the breakpoint in a single function works:

```
function StartServiceByName(const AServer,AServiceName: String):Boolean;
var
  ServiceHandle,
  SCMHandle: SC_HANDLE;
  P: PChar;
begin
  P:= nil;
  Result:= False;
  SCMHandle:= OpenSCManager(PChar(AServer), nil, SC_MANAGER_ALL_ACCESS);
  if SCMHandle <> 0 then
  try
    ServiceHandle:= OpenService(SCMHandle, Pchar(AServiceName),
                                                SERVICE_ALL_ACCESS);
    if ServiceHandle <> 0 then
      Result:= StartService(ServiceHandle, 0, P);

    CloseServiceHandle(ServiceHandle);
  finally
    CloseServiceHandle(SCMHandle);
  end;
end;
```

We call that from the main:

```
if StartServiceByName('DESKTOP-BTLKHKF','ALG') then
writ('ALG started...');
writ('stat of ALG '+
     itoa(ord(GetServiceStatusByName('DESKTOP-BTLKHKF','ALG'))));
  sleep(500)
  writ('stat of ALG '+
              itoa(ord(GetServiceStatusByName('DESKTOP-BTLKHKF','ALG'))));
  //toogle 2___________
  //StopServiceByName('DESKTOP-BTLKHKF','ALG');
```

Example Usage

MaxMatrix Time/Space:

The multiplication of past x future is a vector with the function:= known = f(changeable) [y=f(x)] as distance over time, so **distance** is a function of time: **d=f(t)**.

Conclusion

OpenWeather provides accurate weather data by drawing on trusted sources such as weather stations, satellites, radar systems, and advanced models like GFS, ECMWF, and its own OWHL™. These partnerships with the UK Met Office and other national meteorological agencies boost reliability, with updates released every 10 minutes to ensure precision.
Remarkably, the OWHL™ model reports less than 1% of temperature predictions deviating by more than 5°C. Widely recognised as a leading provider of short-term forecasts, OpenWeather acknowledges that accuracy diminishes over longer periods due to atmospheric complexity. Thus, while short-term forecasts remain highly reliable, it is prudent to view predictions beyond a week with a degree of caution.
With a free plan you have some limits:

- Hourly forecast: unavailable
- Daily forecast: unavailable
- Calls per minute: 60
- 3 hour forecast: 5 days

Script:
https://sourceforge.net/projects/maxbox5/files/examples/1415_weather_listview52.txt/download

References:
https://docs.mapbox.com/help/glossary/access-token/

OpenWeatherMap API guide - OpenWeatherMap

Maps | Mapbox

Tutor 39_2 mapbox Coding

maXbox Primer MAPS and GPS

気象ステーション.

Doc and Tool: maXbox5 - Manage Files at SourceForge.net

Max Kleiner 23/06/2025

maXbox4exe
Digitally signed by maXbox4exe
Date: 2025.06.23 11:25:50 +02'00'