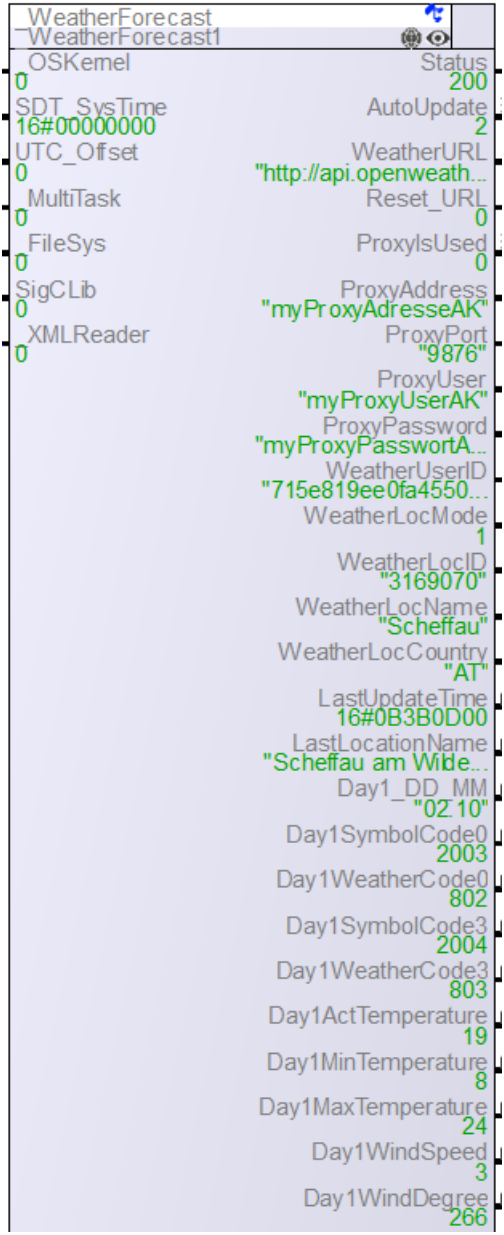
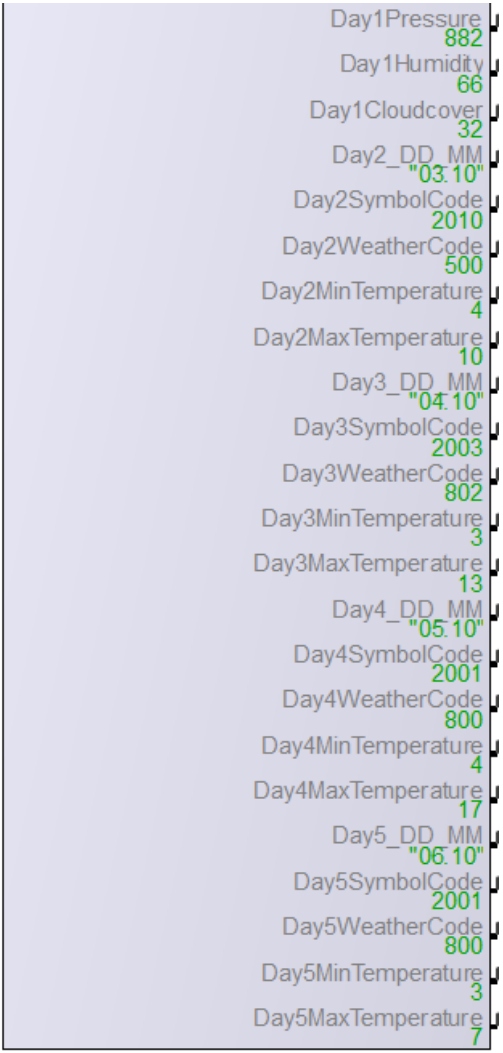
_WeatherForecast





This class can be used in an application for weather forecast. It sends a http request to an internet site and then evaluates the response (XML file). In RTK systems the file HTTPCLIENT.dlm is needed.

Interfaces

Servers

Status	A writing access to this server is a manual update command. Additionally this server shows the current status: Class internal: -1 = Error: not enough data received -2 = Error: HTTP Client error -3 = Error: Service not available -4 = Error: no connection -5 = Error writing XML file -6 = Error reading XML file -7 = Error: wrong XML format 99 = new data are processed Standard HTTP codes: 200 = HTTP transfer OK 400 = request incorrect 401 = no rights 403 = access denied 404 = not found 407 = proxy error 408 = no response 500 = server error 503 = server overloaded 504 = no response Additional codes can be found under: http://en.wikipedia.org/wiki/List_of_HTTP_status_codes
AutoUpdate	Interval for the cyclic update of the data (in minutes) 0 … no automatic update
WeatherURL	URL for the weather request In automatic update mode the URL is always created automatically based on the class servers. In **manual mode** (AutoUpdate = 0) an entered URL is not changed. With a writing access to the server “Reset_URL”, creating the URL based on the class servers can be triggered.
Reset_URL	Writing access sets the URL based on the according class servers
ProxyIsUsed	This server must be set to 1, if a proxy server is used
ProxyAddress	Address of the proxy server (e.g. "hostname.com")
ProxyPort	Port of the proxy server
ProxyUser	User name for the proxy server
ProxyPassword	Password for the proxy server

WeatherUserID	Each customer needs an own account at the weather page - the received user ID must be entered here Site: https://home.openweathermap.org/users/sign_up
WeatherLocMode	Mode for defining the place for the weather forecast 0 = use "Location per ID" (recommended by the provider!!) 1 = use "Location per Name+Country"
WeatherLocID	If the mode "Location per ID" is used: "Location ID" of the desired place for the weather data Link: http://bulk.openweathermap.org/sample/ - File "city.list.json.gz" Example: 7872210 ... Lamprechtshausen 2761367 ... Vienna 2950159 ... Berlin 3169070 ... Rome 6455259 ... Paris 2643743 ... London 5128638 ... New York 1850147 ... Tokyo 1816670 ... Beijing
WeatherLocName	If the mode "Location per Name+Country" is used: Name of the desired place for the weather data
WeatherLocCountry	If the mode "Location per Name+Country" is used: Country of the desired place for the weather data (ISO 3166 Country codes)
LastUpdateTime	Time of the last update (system time of the PLC)
LastLocationName	Name of the place of the last updated weather data
Day1_DD_MM	Day 1 (today): month and day
Day1SymbolCode0 **Day1SymbolCode3**	Day 1 (today): symbol code for "now" Day 1 (today): symbol code for "in 3 hours" Night / Day 1001 / 2001 ... clear sky 1002 / 2002 ... few clouds 1003 / 2003 ... scattered clouds 1004 / 2004 ... broken clouds 1009 / 2009 ... shower rain 1010 / 2010 ... rain 1011 / 2011 ... thunderstorm 1013 / 2013 ... snow 1050 / 2050 ... mist

Day1WeatherCode0 **Day1WeatherCode3**	Day 1 (today): detailed weather code for “now" (xml "symbol number") Day 1 (today): detailed weather code for "in 3 hours" (xml "symbol number") The following list shows the original texts of the "Weather condition codes" Site: https://openweathermap.org/weather-conditions --> Group 2xx: Thunderstorm 200 … thunderstorm with light rain 201 … thunderstorm with rain 202 … thunderstorm with heavy rain 210 … light thunderstorm 211 … thunderstorm 212 … heavy thunderstorm 221 … ragged thunderstorm 230 … thunderstorm with light drizzle 231 … thunderstorm with drizzle 232 … thunderstorm with heavy drizzle --> Group 3xx: Drizzle 300 … light intensity drizzle 301 … drizzle 302 … heavy intensity drizzle 310 … light intensity drizzle rain 311 … drizzle rain 312 … heavy intensity drizzle rain 313 … shower rain and drizzle 314 … heavy shower rain and drizzle 321 … shower drizzle --> Group 5xx: Rain 500 … light rain 501 … moderate rain 502 … heavy intensity rain 503 … very heavy rain 504 … extreme rain 511 … freezing rain 520 … light intensity shower rain 521 … shower rain 522 … heavy intensity shower rain 531 … ragged shower rain --> Group 6xx: Snow 600 … light snow 601 … snow 602 … heavy snow 611 … sleet 612 … shower sleet 615 … light rain and snow

	616 … rain and snow 620 … light shower snow 621 … shower snow 622 … heavy shower snow --> Group 7xx: Atmosphere 701 … mist 711 … smoke 721 … haze 731 … sand, dust whirls 741 … fog 751 … sand 761 … dust 762 … volcanic ash 771 … squalls 781 … tornado --> Group 800: Clear 800 … clear sky --> Group 80x: Clouds 801 … few clouds 802 … scattered clouds 803 … broken clouds 804 … overcast clouds --> Group 90x: Extreme 900 … tornado 901 … tropical storm 902 … hurricane 903 … cold 904 … hot 905 … windy 906 … hail --> Group 9xx: Additional 951 … calm 952 … light breeze 953 … gentle breeze 954 … moderate breeze 955 … fresh breeze 956 … strong breeze 957 … high wind, near gale 958 … gale 959 … severe gale 960 … storm 961 … violent storm 962 … hurricane

Day1ActTemperature	Day 1 (today): current temperature in °C
Day1MinTemperature	Day 1 (today): minimum temperature in °C
Day1MaxTemperature	Day 1 (today): maximum temperature in °C
Day1WindSpeed	Day 1 (today): current wind speed in km/h
Day1WindDegree	Day 1 (today): current wind direction in degrees
Day1Pressure	Day 1 (today): current air pressure in mbar
Day1Humidity	Day 1 (today): current humidity in %
Day1Cloudcover	Day 1 (today): current cloudcover in %
Day2_DD_MM	Day 2: month and day
Day2SymbolCode	Day 2: symbol code (listing see day 1)
Day2WeatherCode	Day 2: detailed weather code (listing see day 1)
Day2MinTemperature	Day 2: minimum temperature in °C
Day2MaxTemperature	Day 2: maximum temperature in °C
Day3_DD_MM	Day 3: month and day
Day3SymbolCode	Day 3: symbol code (listing see day 1)
Day3WeatherCode	Day 3: detailed weather code (listing see day 1)
Day3MinTemperature	Day 3: minimum temperature in °C
Day3MaxTemperature	Day 3: maximum temperature in °C
Day4_DD_MM	Day 4: month and day
Day4SymbolCode	Day 4: symbol code (listing see day 1)
Day4WeatherCode	Day 4: detailed weather code (listing see day 1)
Day4MinTemperature	Day 4: minimum temperature in °C
Day4MaxTemperature	Day 4: maximum temperature in °C
Day5_DD_MM	Day 5: month and day
Day5SymbolCode	Day 5: symbol code (listing see day 1)
Day5WeatherCode	Day 5: detailed weather code (listing see day 1)
Day5MinTemperature	Day 5: minimum temperature in °C
Day5MaxTemperature	Day 5: maximum temperature in °C

Clients

_OSKernel	Object channel to the system interface _OSKernel (created automatically)
SDT_SysTime	Connection to the object of the class _SysDateTime to the server SysTime
UTC_Offset	Offset to UTC standard time (in hours)
_MultiTask	Object channel to the system interface _MultiTask (created automatically)
_FileSys	Object channel to the system interface _FileSys (created automatically)
SigClib	Object channel to the system interface SigCLib (created automatically)
_XMLReader	Object channel to an object of the class _XMLReader

Global Methods

Init	Initializations
Background	Background task inactive in scope of delivery, cyclic function works via thread
ParallelTask	Parallel task (thread)
CyclicWork	Function for all cyclic tasks
SetWeatherURL	Creates and sets the URL for the web weather request
SetOptions	Function to set the program options (proxy string)
GetDataOverHttp	Function for sending the web weather request

Function

Cyclically or triggered manually, the class requests the weather data from the API of the following web site via http:
https://openweathermap.org/

Always five days in 3 hours intervals are requested.

The behavior can be set via different class servers.

In automatic update mode, the URL is always created automatically based on the class servers, then the weather data are requested and evaluated.

In manual mode (AutoUpdate = 0) the entered URL is not changed.
The transfer is triggered manually with a writing access to the server “Status”.
With a writing access to the server “Reset_URL”, creating the URL based on the class servers can be triggered.

So the weather data for today and the next four days are generated, which are output on the class servers (they e.g. can be displayed in the AddOn WeatherForecast).

Requirements

→ The PLC must have valid network settings, they have to be asked from the network administrator.
Some entries are needed in the **autoexec.lsl**:

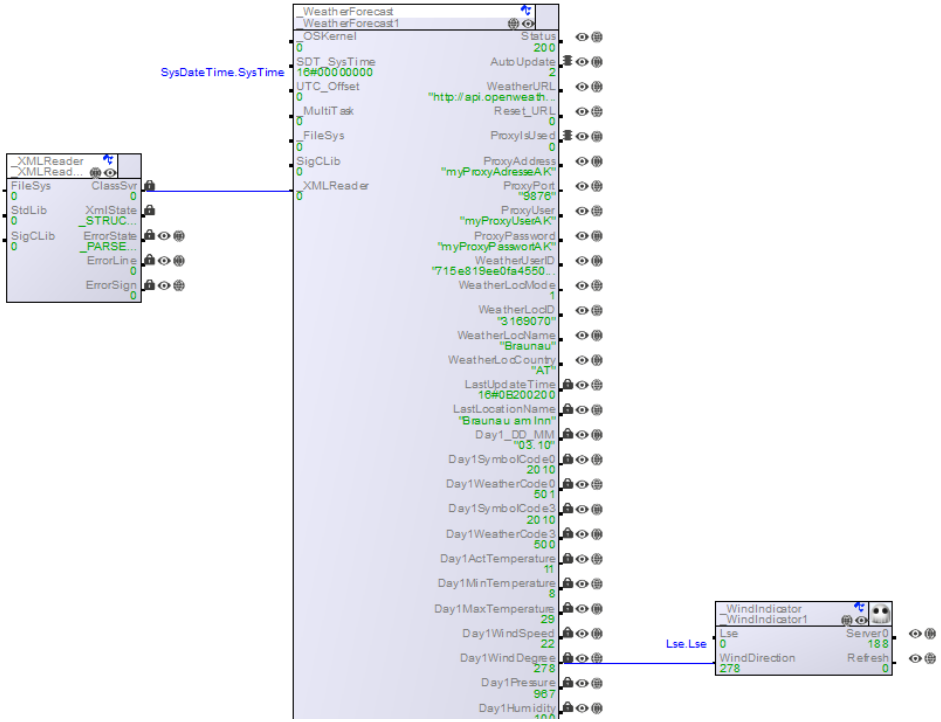
```
SET IP 1 HOSTADDR 10 100 0 209 GATEWAY 10 10 1 1
SET IP DNS 10.30.0.1
```

→ ONLY for **RTK** systems:
In order to be able to use the HTTP client interface, the following files must be copied to the directory **C:\LSLSYS**: **HTTPCLI.DLM** and **ZLIB.DLM**
The current versions are provided by the SIGMATEK support.

Defines

```
//=============================================================================================================================

#define WEATHERFC_DEF_URL_P1  "http://api.openweathermap.org/data/2.5/forecast?"   // URL Default part 1
#define WEATHERFC_DEF_URL_EXT "&units=metric&mode=xml"                             // URL Default extensions


#define WEATHERFC_XmlFile     "WeatherForecast.xml"   // Name of the XML-file

#define WEATHERFC_XmlEntries  40                      // Number of entries in the weather-xml (5 days * 8 entries/day)


#define NEW_DATA_IN_PROCESS    99                     // staus "new data will be processed"
#define HTTP_REQUEST_OK       200                     // http return code "everything ok"

//=============================================================================================================================
```

Example Application



The object "_WeatherForecast1" serves for the described weather forecast.

The class "_XMLReader" is a standard class from the Tools.Lib, it is needed for parsing the received XML files.

The class "_WindIndicator" serves for visualizing a direction arrow based on the current wind direction. It draws the direction arrow on its own, in the screen project only a numedit object is placed and then animated with the _WindIndicator object.

The application shown above is used in the AddOn "Weather Forecast" and uses the functionality of the standard template (system time).