Tool for Communication

applications & TOOLS

HTML Generator as OPC DA Client for simple plant visualization via Standard Web Browser

SIEMENS

Tool

Note

The function blocks and tools are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The function blocks and tools do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are correctly used. The introduced examples do not relieve you of the responsibility of safely and professionally using, installing, operating and servicing equipment. Using these function examples and tools within the examples, you accept that Siemens cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these function examples and tools at any time without prior notice. If there are any deviations between the recommendations provided in these examples and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

Warranty, Liability and Support

We accept no liability for information contained in this document.

Any claims against us – based on whatever legal reason – resulting from the use of the function blocks and tools, information, programs, engineering and performance data etc., described in this example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above provisions do not imply a change in the burden of proof to your detriment.

For questions about this document please use the following
e-mail-address:

mailto:csweb@ad.siemens.de

Foreword

In this example we introduce fully functional and tested automation configurations based on A&D standard products for simple, fast and inexpensive implementation of automation tasks.

The function blocks or tools used here facilitate realizing a certain partial function from this example. Apart from a list of all required hardware and software components and a description of the way they are connected to each other, the examples include the tested tools or function blocks. This ensures that the functionalities described here can be reset in a short period of time and thus also be used as a basis for individual expansions.

Table of Contents

Table of Contents 3

1 Automation Description 4

2 Setup 7

3 General Function Mechanisms 8

4 Required Hardware and Software Components 10

Components for an S7-200 configuration 10

Components for an S7-300 configuration 10

Components for an S7-400 configuration 10

PC station 10

Configuration software/tools 11

5 Function Method of the HTML OPC Generator 12

5.1 Function Chain of the OPC HTML Generator 12

5.2 User interface of the HTML OPC Generator 13

5.3 The HTML template 17

5.4 The dummy 19

5.5 The CSS-file 21

6 Installation and Operation of the Sample/Tool 24

Preliminary remark 24

6.1 Installation of Hardware and Software 24

6.2 Configuration of the PG/PC 25

6.3 Installation of the user program 25

6.4 Operating the example 27

7 Basic Performance Data 30

OPC HTML Generator 30

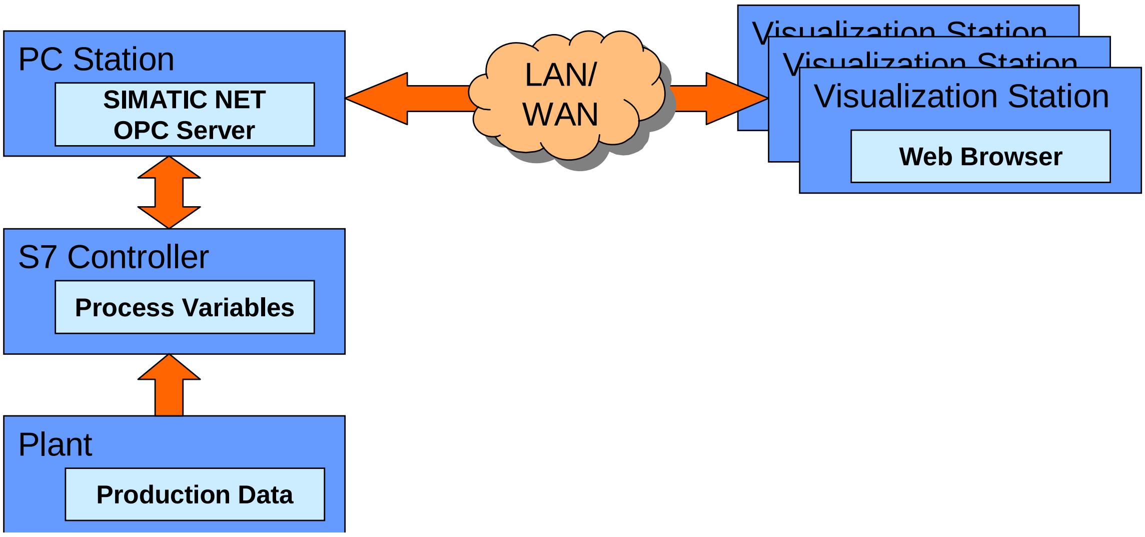
8 History 30

1 Automation Description

Automation Task

A production plant is to be monitored with one or several simple visualization stations as cost-efficiently as possible. Standardized coupling methods (OPC) and user interfaces (web browser) are to be used, which can be employed in a LAN as well as a WAN.

Figure 1-1



Automation Solution

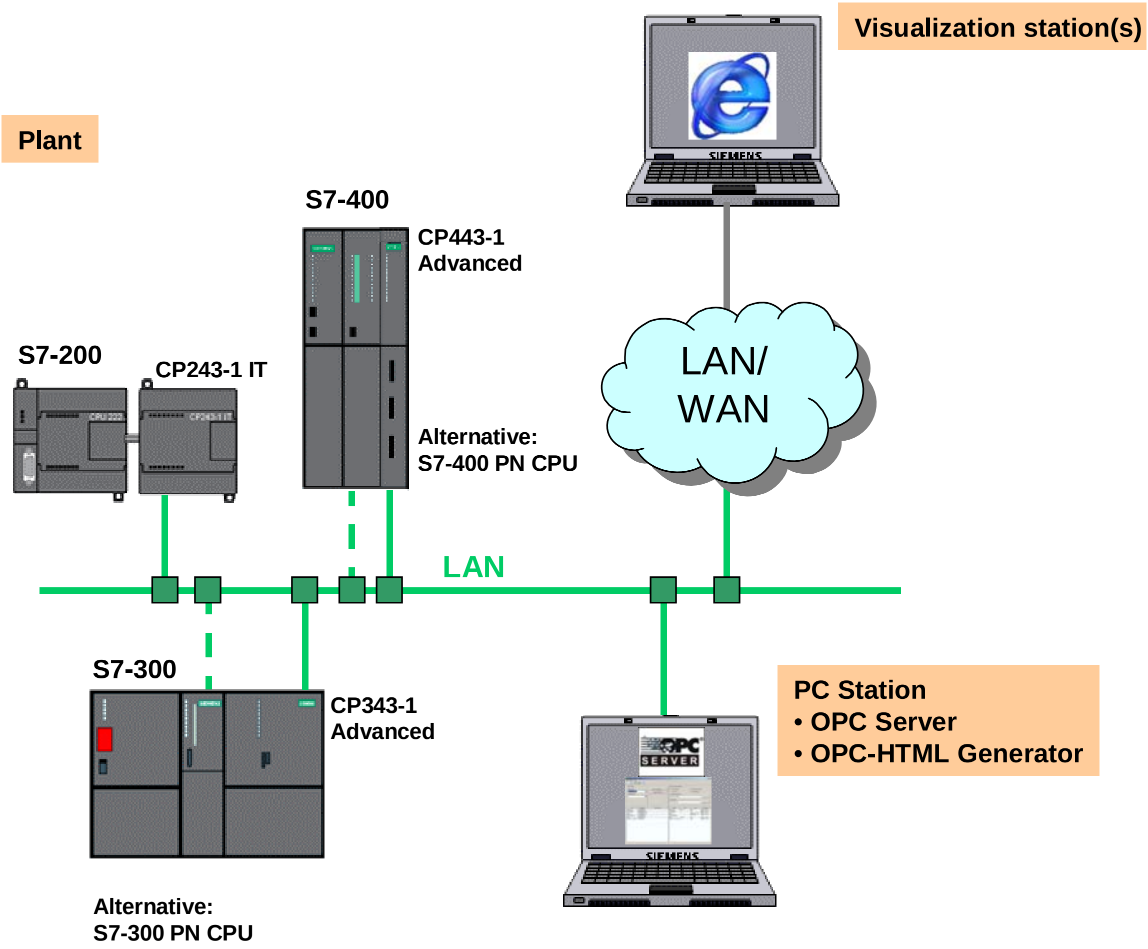
In this example of a small weather station, we show you a simple option of visualizing S7 process variables on a web browser using the HTML generator in an industrial Ethernet environment.

The application on hand on the PC station of the plant is an OPC-DA-Client and periodically generates a HTML file which represents configurable OPC variables. This HTML file can be viewed by each standard web browser which is capable of XHTML and CSS (e.g. directly via a LAN, or via a web server).

No previous programming knowledge is required. The application is already programmed completely and needs only be configured with your desired variables.

The following graphic shows different SIMATIC HW-Configurations based on Industrial Ethernet the OPC HTML Generator can be used with.

Figure 1-2



Note

The plant can, with other bus systems and corresponding interfaces ,also be connected to the OPC server in the PC station (PROFIBUS, MPI)

Application areas / Customer benefits

This tool/function block can be used

- for all simple visualization tasks in the lower performance range (only monitoring), such as
  - Building control systems
  - Monitoring of remote systems / machines, etc.

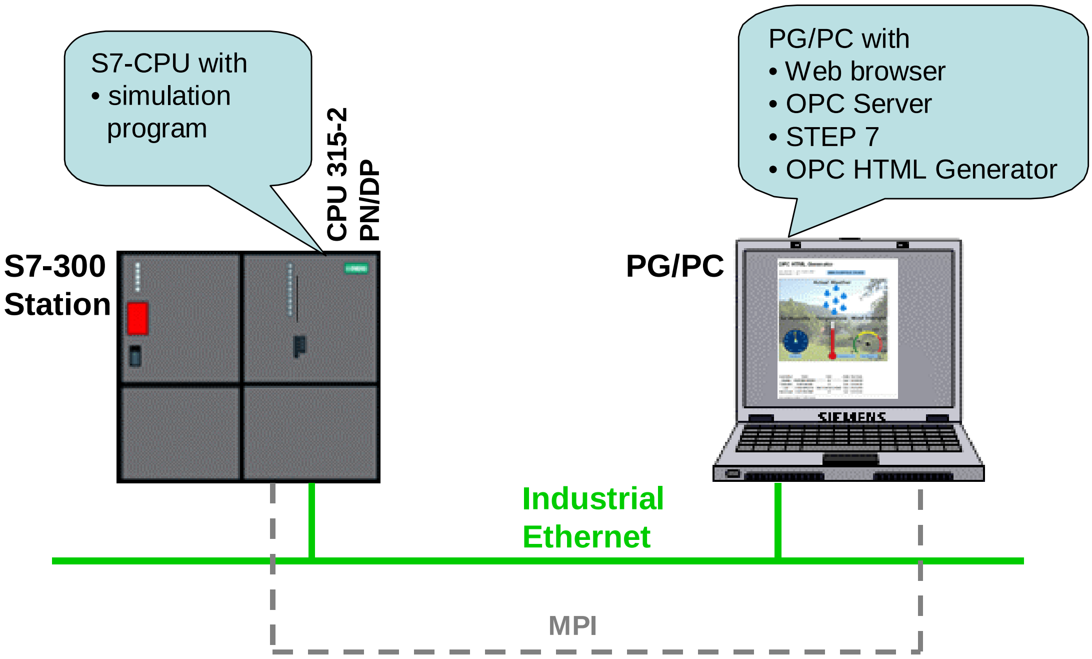
This solution has the following advantages for the user:

- no previous programming skills required
- runnable on SIMATIC S7-200/300/400 (via CPs/ PN-CPUs)
- can be based on standard web browsers
- no OPC-Client installation necessary on the visualization stations
- individual design of the HTML page based on a template
- safe read-only access to OPC-Server without DCOM settings possible

2 Setup

The following figure shows the setup example with a S7-300 station and PROFINET S7-CPU.

Figure 2-1



In this example the OPC connection occurs via Ethernet with a Profinet CPU. However, there is also the option of establishing an OPC connection via Profibus or MPI.

3 General Function Mechanisms

What is OPC?

OPC is a software interface independent of manufacturer, based on COM/DCOM, which enables data exchange between hardware and software also from different manufacturers. The OPC Foundation, an interest group of well-known manufacturers, created these four OPC specifications.

Table 3-1

Specification	Application
OPC Data Access (DA)	Access to process data
OPC Alarm& Events (A&E)	Interface for event-based information including acknowledgement
OPC Historical Data Access (HDA)	Function for archived data
OPC Data eXchange (DX)	Server to server cross-communication.

This example uses exclusively the “OPC Data Access” specification.

OPC is a client/server architecture. Manufacturers for modules providing process data provide an OPC server with their modules which interfaces to the respective data source. An OPC client contacts the server and reads or writes the data.

OPC DA interface

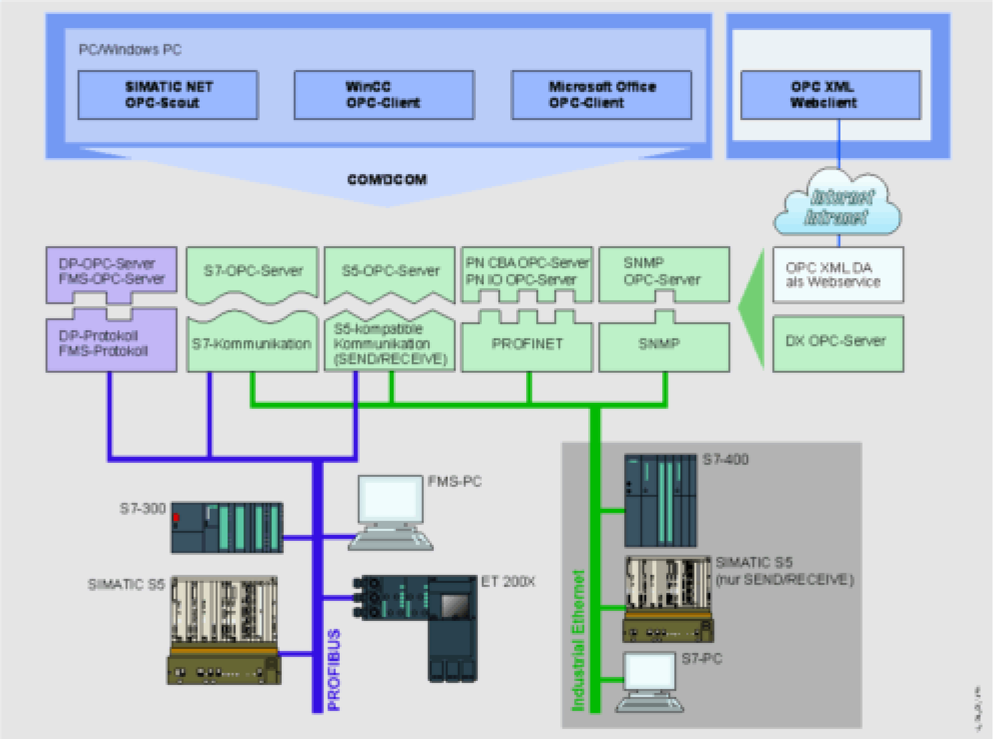
The OPC DA specification was the first OPC standard. It is used to exchange process data between controller and HMI devices or other clients. The read data contains data type, time stamp and quality, the statements on the quality of the value. The clients can read process data as well as write commands to the OPC server. The server then forwards the control data to the controller.

SIMATIC NET OPC server

The following graphic illustrates a structure with different SIMATIC NET OPC servers and the respective protocol drivers.

For this application the S7 OPC server was used. An S7 communication is established with an S7-300 via Ethernet.

Figure 3-1



The SIMATIC NET OPC server provides the following accesses:

- Industrial Ethernet
  - S7 communication
  - S5-compatible communication
  - PROFINET
  - SNMP
- PROFIBUS
  - DP protocol
  - FMS protocol
  - S7 communication
  - S5-compatible communication

4 Required Hardware and Software Components

For this example we only list the components of a PN solution. The example is also possible via PROFIBUS connection or via MPI. At the PC Station the respective interface cards must then be installed.

Components for an S7-200 configuration

Table 4-1

Component	Qty.	MLFB / Order number	Note
Power Supply SITOP	1	6EP1332-1SH31	or similar power supply
CPU 222	1	6ES7212-1AB23-0XB0	or similar
CP243-1 IT	1	6GK7243-1GX00-0XE0	Alternative: PROFIBUS connection

Components for an S7-300 configuration

Table 4-2

Component	Qty.	MLFB / Order number	Note
PS 307 5A	1	6ES7307-1EA00-0AA0	or similar power supply
CPU 315-2 PN/DP	1	6ES7315-2EG10-0AB0	Alternative: take normal CPU and establish a connection to the PG/PC via MPI/PROFIBUS-CP/Ethernet-CP . Types: 315-2 DP, 313C, CPU 317-2DP

Components for an S7-400 configuration

Table 4-3

Component	Qty.	MLFB / Order number	Note
PS 407 4A	1	6ES7407-0DA02-0AA0	or similar power supply
CPU 414-3 PN/DP	1	6ES7414-3EM05-0AB0	Alternative: take normal CPU and establish a connection to the PG/PC via MPI/PROFIBUS-CP/Ethernet-CP . Types: CPU 414-3, CPU 416-3, CPU 417-4

PC station

Table 4-4

Component	Qty.	MLFB /Order number	Note
CP 1613 Ethernet PCI card	1	6GK1161-3AA00	Or similar Ethernet card
CP 5611 A2 PCI-KARTE		6GK1561-1AA01	Optional: Only if OPC connection runs via Profibus/MPI.

Configuration software/tools

Table 4-5

Component	Qty.	MLFB / Order number	Note
SIMATIC STEP 7 Version 5.4 + SP1	1	6ES7810-4CC08-0YA5	or higher
SIMATIC NET SOFTWARE EDITION 2005	1	6GK1704-0AA07-3AA0	or higher OPC-Server, OCX Data Control
SIMATIC NET IE SOFTNET-S7/2006	1	6GK1704-1CW64-3AA0	Software license
SIMATIC NET PB SOFTNET-DP/2006		6GK1704-5DW64-3AA0	Optional for PROFIBUS, MPI, etc.
Webbrowser	1		e.g. Internet Explorer by Microsoft

Example files and projects

The software examples are available on the HTML page from which you downloaded this document.

Clicking the link below directly displays the download page of this document.

http://support.automation.siemens.com/WW/view/en/24272005

The download is a ZIP file which can be unzipped with any zip-program. The zip.-file contains the following files:

Table 4-6

File name	Contents
S7_html_OPC.zip	Contains the archived STEP 7 project
\htmlgen	Folder contains the OPC HTML Generator and the example project

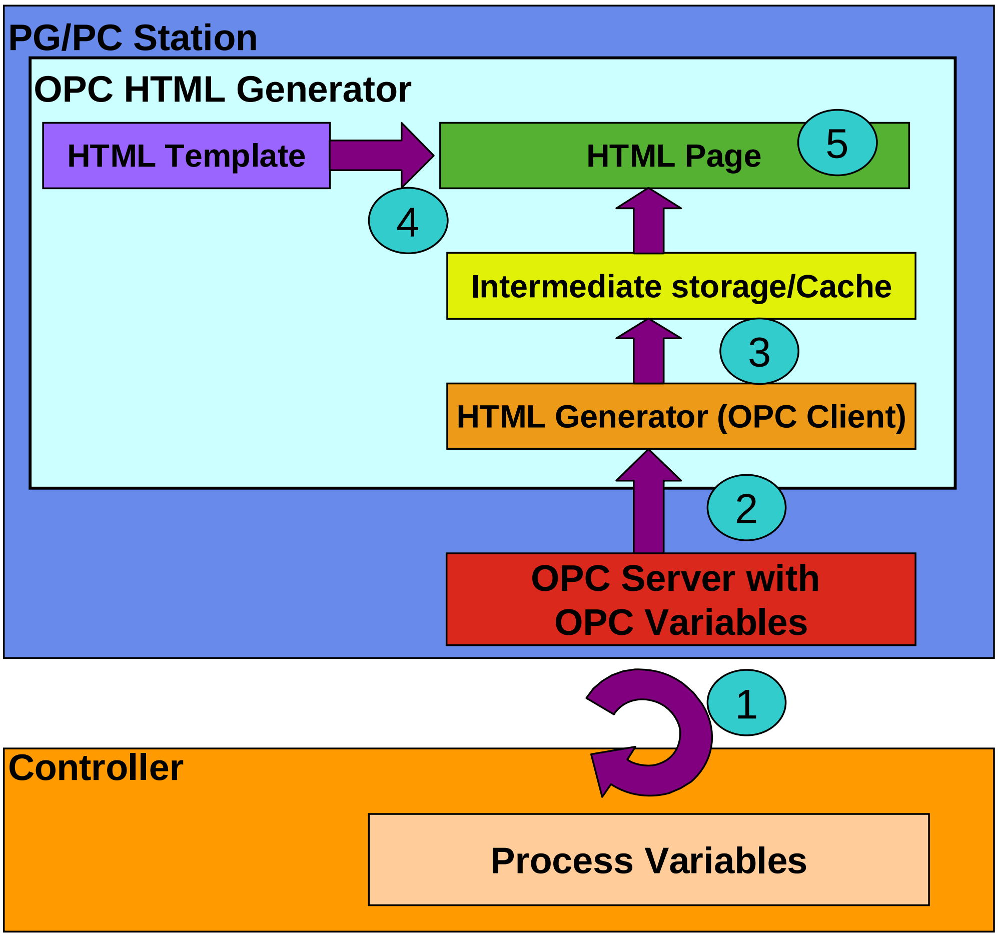
5 Function Method of the HTML OPC Generator

5.1 Function Chain of the OPC HTML Generator

Display

The following figure illustrates the interaction of the individual components.

Figure 5-1



Description

The numbers have the following meaning:

Table 5-1

Number	Description
1	The OPC server continuously monitors the assigned process variables for.
2	The HTML generator connects with a specified OPC server as OPC client and reads the values of the variables via the Data Access interface. Thereby only the changed values are transferred.
3	The read data are saved in a cache.
4	In selectable intervals, the variables are downloaded from the cache and an internal copy of the HTML template is generated. This copy is filled up with the variables. The template is configured with dummies assigned to a certain OPC variable. These dummies are now replaced by current variable values.
5	The result is saved in a new HTML page which can be called via a web browser.

Note

This individual HTML page can now be read by any XHTML capable web browser.

If a folder on a Web server (e.g. IIS server) is specified as path name, the HTML page can be made available for the internet.

5.2 User interface of the HTML OPC Generator

The user interface has a menu, a toolbar and a work space. The work space is divided into two sections – OPC configuration and HTML template configuration. The division can be configured using a splitter.

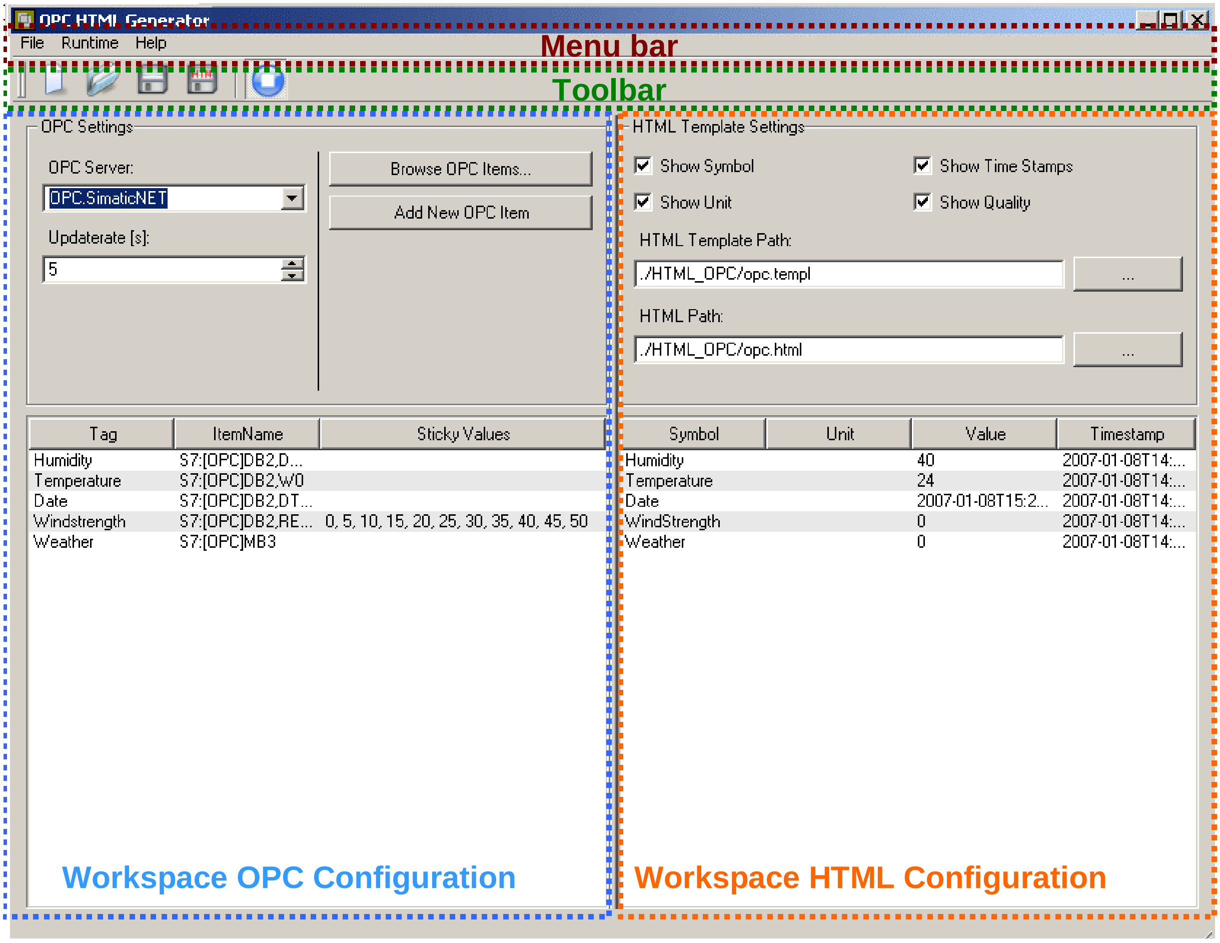
File format

All settings configured on the user interface can be saved in a configuration file. Several configurations can be managed in separate configuration files. The file format for saving is **.ini**.

User interface

The figure below shows the setup of the user interface:

Figure 5-2



OPC Settings

OPC Server: OPC.SimaticNET

Updaterate [s]: 5

HTML Template Settings

☑ Show Symbol ☑ Show Time Stamps

☑ Show Unit ☑ Show Quality

HTML Template Path: ./HTML_OPC/opc.templ

HTML Path: ./HTML_OPC/opc.html

Tag	ItemName	Sticky Values
Humidity	S7:[OPC]DB2,D...	
Temperature	S7:[OPC]DB2,W0	
Date	S7:[OPC]DB2,DT...	
Windstrength	S7:[OPC]DB2,RE...	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Weather	S7:[OPC]MB3	

Symbol	Unit	Value	Timestamp
Humidity		40	2007-01-08T14:...
Temperature		24	2007-01-08T14:...
Date		2007-01-08T15:2...	2007-01-08T14:...
WindStrength		0	2007-01-08T14:...
Weather		0	2007-01-08T14:...

The menu bar

Various functions have been assigned to the menu items of the user interface. The following table lists the individual menu items with submenu and describes their meaning.

Table 5-2

Menu	Function / Meaning
• File - New - Open - Save - SaveAs - Save HTML Template	 Generates a new empty configuration without name. Opens a saved configuration file. Saves the current configuration Saves the configuration under a selectable name and place. Generates a new HTML template on the basis of the configuration.
• Runtime - Start/Stop	 Starts the Runtime. The connection to the OPC server is established and the HTML file cyclically generated.

Menu	Function / Meaning
• Help	
- Contents	Operator manual for the HTML OPC Generator
- Index	Searches for a key word in the user manual
- About	Displays the About Dialog
- AboutQt	Displays the About Dialog for the used third-party product Qt -A

The toolbar

The toolbar enables a faster access to the most important functions of the menu.

Table 5-3

Symbol	Function / Meaning
🗋	This symbol is identical with File -> New
📂	This symbol is identical with File -> Open
💾	This symbol is identical with File -> Save
HTM	This symbol is identical with File -> Save HTML Template
▶	This symbol is identical with Runtime -> Start
⏹	This symbol is identical with Runtime -> Stop

Work space

OPC configuration

The left part of this work space is determined for the OPC configuration. The OPC server with the selection box **OPC Server** is selected.

The input field **Updaterate(s)** defines the speed at which the HTML page is automatically updated via “Reload“. For the OPC communication, a sensible update rate is derived from this.

Using the **Browser OPC Items…** button, OPC items can be selected and accepted in the list. The **Add New OPC Item** button adds a new entry to the list for manual input.

The columns of the list have the following meaning:

Table 5-4

Column name	Description	Data relevance
Day	Name of the dummy for the OPC variable. This dummy is located in the HTML template and is during runtime replaced with the current value of the variable	Default name; can be additionally edited by the user
ItemName	Definition of the OPC Item ID, from which values are to be read	Automatically inserted; can be additionally edited by the user
Sticky Values	Establishing a value range; for an imprecise variable value, it is rounded and replaced by the next suitable value from this range.	Definition by the user:

HTML template configuration

The right part of this work space is used for configuration of the HTML page. Using checkboxes, it can be selected whether the symbol name(Symbol), the unit(Unit), the time stamp(Timestamp) and the quality(Quality) of the OPC variables shall be displayed on the HTML page. In the **HTML Template Path** input field the filling location for the HTML template is selected. In **HTML Path** for that HTML file, which the HTML Generator generates and updates automatically.

The columns of the list have the following meaning:

Table 5-5

Column name	Description	Data relevance
Symbol	Symbolic name for the variables	Default name; can be additionally edited by the user
Unit	Unit of the variables	Can be additionally edited by the user
value	Current value of the variable.	Provided by the OPC server
Timestamp	The time stamp of the variable value	Provided by the OPC server

The lists of the variables from the side of the **OPC Configuration** and **HTML Template Configuration** are synchronous. This means, that when entering a new OPC variable it also appears in the list under HTML Template Configuration or when deleting a variable it disappears from both lists.

5.3 The HTML template

The HTML template serves as a template for the generated HTML page. It contains a table-oriented layout an dummies for the values to be displayed. During runtime, these dummies are replaced from the template with the current values of the OPC variables and written into a HTML file.

There are two possibilities for generating the HTML template

- Automatic generation by the OPC HTML Generator
- User-defined generation of the HTML template

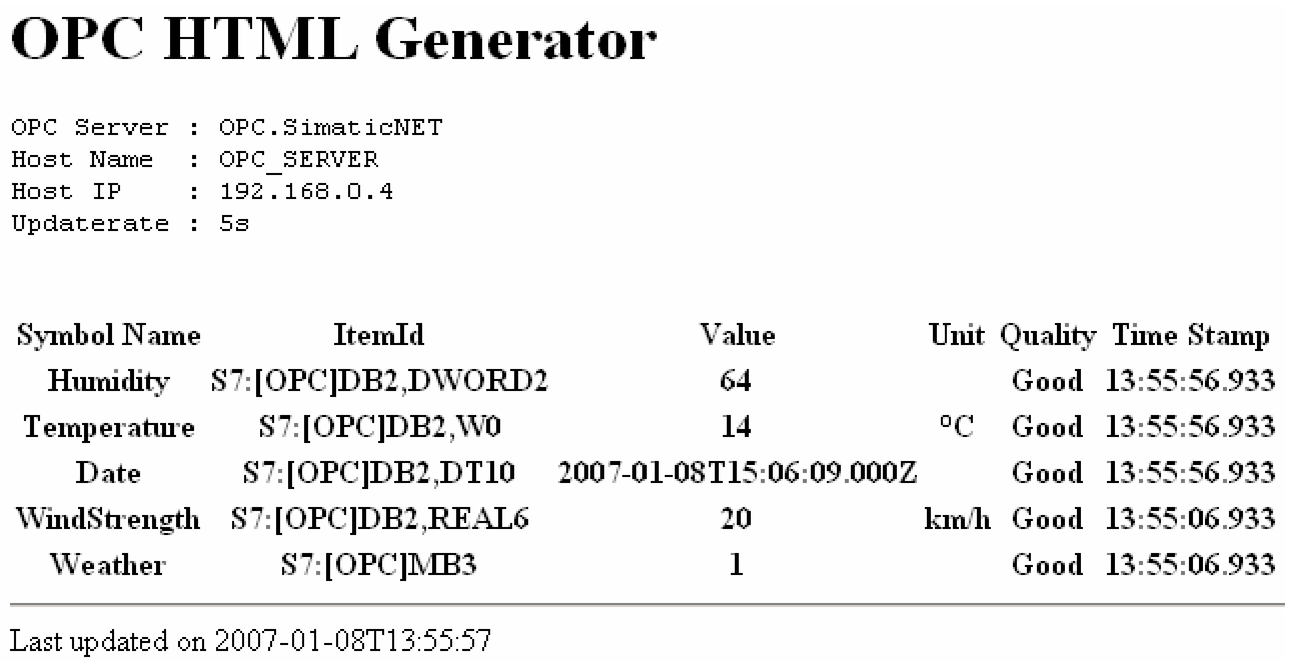
Automatic generation

Automatic generation of the HTML template on the basis of the variable configuration occurs simply via the user interface of the OPC HTML generators and is saved under the name given in the configuration.

The Look&Feel (colors, fonts, frames, etc.) of the HTML template can be filed in a separate CSS file and be adjusted at will.

The following extract gives an example of a generated HTML page. As a template serves the automatically generated HTML template.

Figure 5-3



OPC HTML Generator

```
OPC Server : OPC.SimaticNET
Host Name  : OPC_SERVER
Host IP    : 192.168.0.4
Updaterate : 5s
```

Symbol Name	ItemId	Value	Unit	Quality	Time Stamp
Humidity	S7:[OPC]DB2,DWORD2	64		Good	13:55:56.933
Temperature	S7:[OPC]DB2,W0	14	°C	Good	13:55:56.933
Date	S7:[OPC]DB2,DT10	2007-01-08T15:06:09.000Z		Good	13:55:56.933
WindStrength	S7:[OPC]DB2,REAL6	20	km/h	Good	13:55:06.933
Weather	S7:[OPC]MB3	1		Good	13:55:06.933

Last updated on 2007-01-08T13:55:57

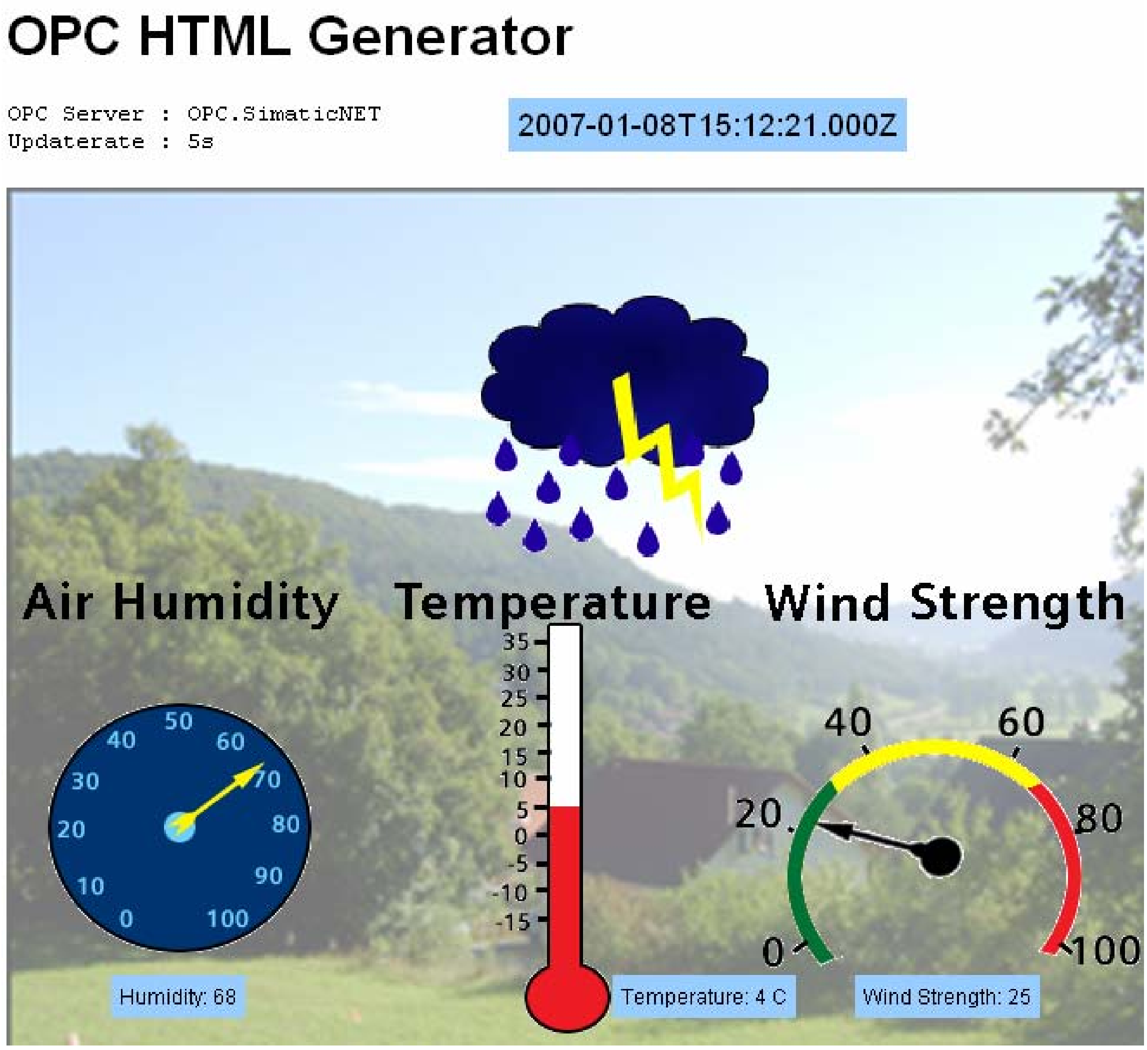
User defined settings

The HTML template can also be configured individually. It is not only restricted to text output, but can also be used for display of image sequences controlled by the variables.

On the user interface of the OPC HTML Generator only the path must be given, where the individual HTML template is located, as well as the path for the HTML file to be generated be specified.

Here the Look&Feel (colors, fonts, frames, etc.) of the HTML template can also be filed in a separate CSS file.

Figure 5-4



OPC HTML Generator

```
OPC Server : OPC.SimaticNET
Updaterate : 5s
```

2007-01-08T15:12:21.000Z

Humidity: 68

Temperature: 4 C

Wind Strength: 25

Note If you wish to generate an individual HTML template, it is helpful to build on an automatically generated template. Accept the HTML structure and file heading and remove the HTML code.

5.4 The dummy

Each OPC variable has its own defined dummy. This dummy is used in the HTML template and replaced with the current value of the variable during runtime, and copied into the corresponding HTML page.

The dummies are embraced by %-signs as in Windows, to recognize them. Non-existent dummies are simply ignored by the application and do not generate an error.

Types of dummies

There are two types of dummies.

- General dummies (e.g. Updaterate)
- Dummies consisting of the tag name and postfix

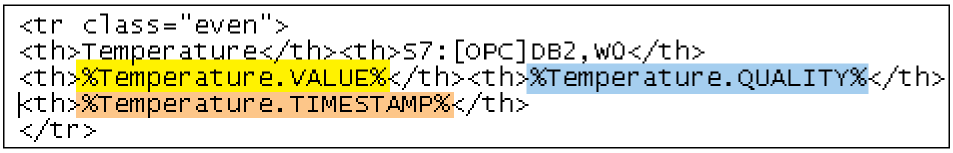
Table 5-6

Dummy	Description
UPDATERATE	Currently set update rate
LAST_UPDATE	Tim of last HTML update
<TAG>.VALUE	Value of <TAG>
<TAG>.QUALITY	Quality of <TAG>
<TAG>.TIMESTAMP	Time stamp of <TAG>

Example code with dummy

The following extract shows the HTML Code from the template with dummy:

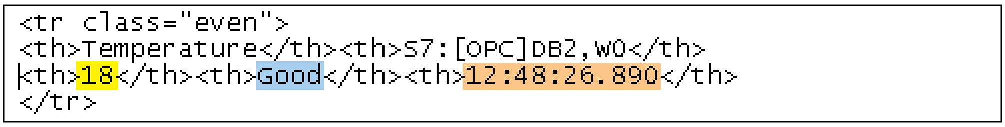
Figure 5-5



```
<tr class="even">
<th>Temperature</th><th>S7:[OPC]DB2,W0</th>
<th>%Temperature.VALUE%</th><th>%Temperature.QUALITY%</th>
<th>%Temperature.TIMESTAMP%</th>
</tr>
```

During runtime, the dummies are replaced by current values and copied into the HTML file:

Figure 5-6



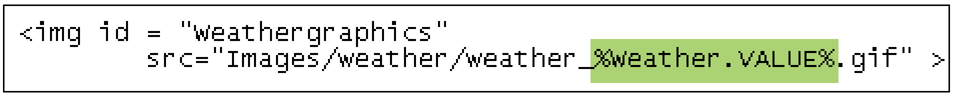
```
<tr class="even">
<th>Temperature</th><th>S7:[OPC]DB2,W0</th>
<th>18</th><th>Good</th><th>12:48:26.890</th>
</tr>
```

Visualization with dummy and graphics

Dummies also enable assigning possible variable values to image sequences. To do this, a dummy is installed in the filename, which is replaced by a value (as part of the image name) during runtime.

The template

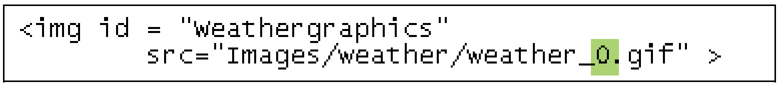
Figure 5-7



```
<img id = "weathergraphics"
        src="Images/weather/weather_%weather.VALUE%.gif" >
```

will during runtime be transformed into

Figure 5-8

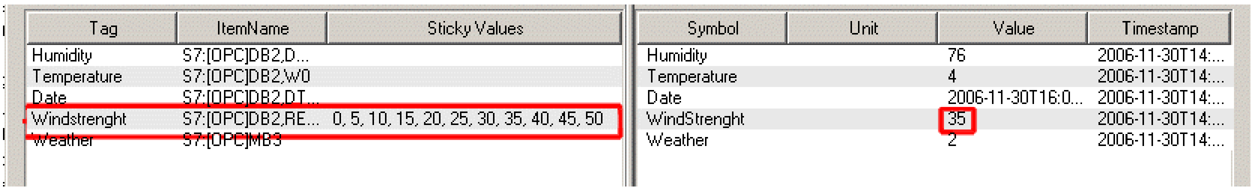


```
<img id = "weathergraphics"
        src="Images/weather/weather_0.gif" >
```

Sticky Values

Analog values such as temperature are generally imprecise and fluctuate in the decimal range. A graphical display of these imprecise values is very difficult, as this would require an undefined number of images. For mapping values to images for visualization purposes, it is an advantage if they are integers. This is where **Sticky Values** come in. On the user interface of the OPC HTML Generator **Sticky Values** can be generated for any variable. The OPC variable is connected to it. If the value of the OPC variable changes, the Sticky Values are searched for the next suitable configured value which is used instead.

Figure 5-9



Tag	ItemName	Sticky Values
Humidity	S7:[OPC]DB2,D...	
Temperature	S7:[OPC]DB2,W0	
Date	S7:[OPC]DB2,DT...	
Windstrenght	S7:[OPC]DB2,RE...	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Weather	S7:[OPC]MB3	

Symbol	Unit	Value	Timestamp
Humidity		76	2006-11-30T14:...
Temperature		4	2006-11-30T14:...
Date		2006-11-30T16:0...	2006-11-30T14:...
WindStrenght		35	2006-11-30T14:...
Weather		2	2006-11-30T14:...

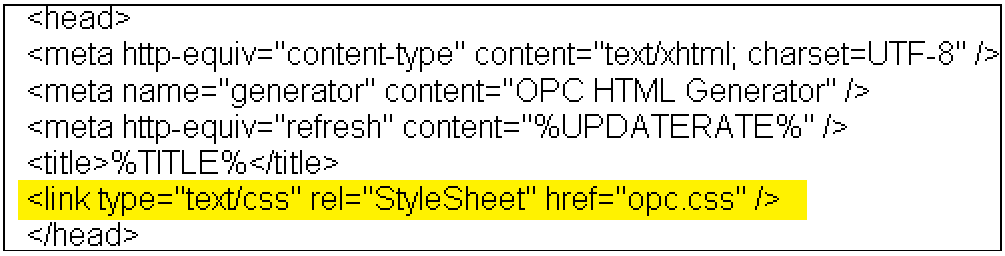
5.5 The CSS-file

The Cascading Style Sheet is a supplement for HTML programming. This technique enables storing the formatting of HTML elements and layout into an independent file, which separates content and design. This method has the advantage, that changes need only be made in this individual file and not in the HTML code itself. This makes the HTML code much clearer and smaller.

Integration into the HTML file

If format definitions are filed in a separate file, it must be referenced to in the data heading of an HTML file. The **<link>** tag is used for this.

Figure 5-10



```
<head>
<meta http-equiv="content-type" content="text/xhtml; charset=UTF-8" />
<meta name="generator" content="OPC HTML Generator" />
<meta http-equiv="refresh" content="%UPDATERATE%" />
<title>%TITLE%</title>
<link type="text/css" rel="StyleSheet" href="opc.css" />
</head>
```

The attribute **<href>** serves for addressing the desired CSS file.

File format

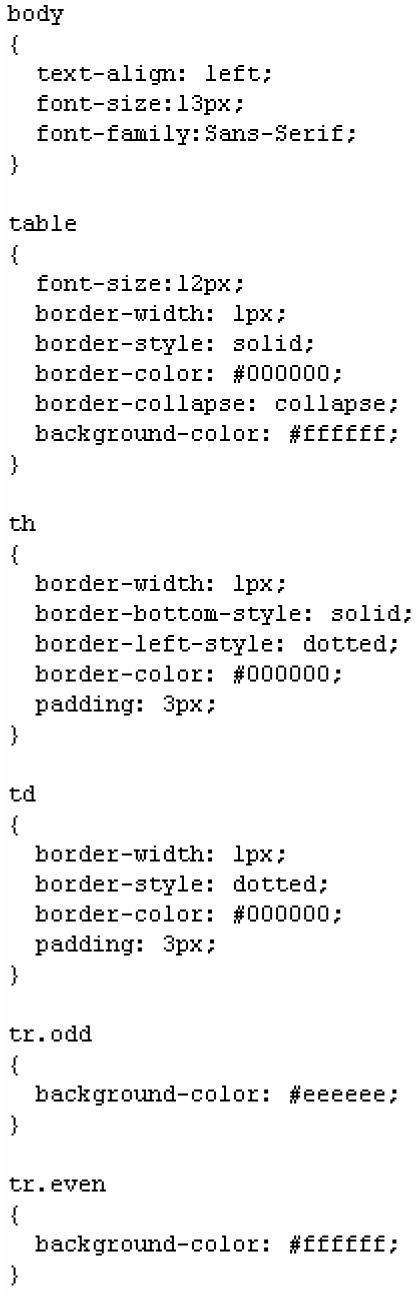
The referenced CSS file must be a mere text file with the ending **.css**. It must not contain anything but definitions of formats and comments.

Note A CSS file can be generated with a normal text editor.

Example CSS file for an automatically generated HTML page

The following graphic shows a possible CSS file for an automatically generated HTML template:

Figure 5-11

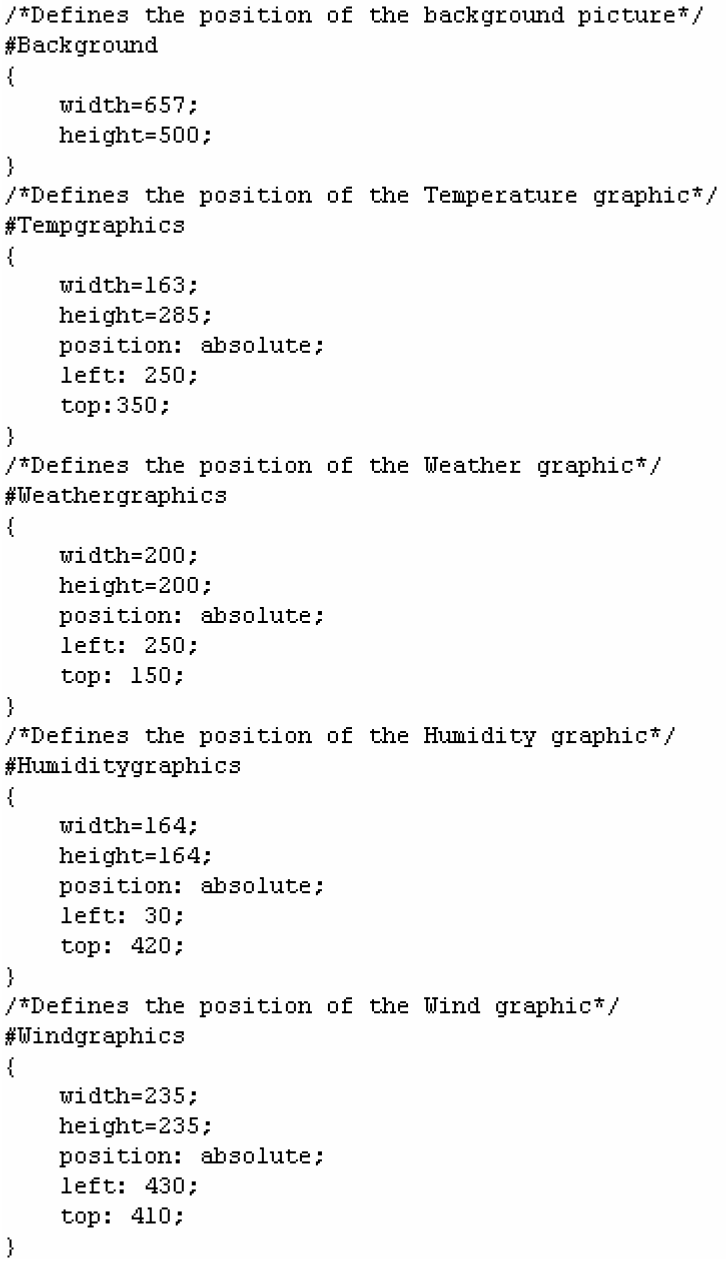


```
body
{
  text-align: left;
  font-size:13px;
  font-family:Sans-Serif;
}

table
{
  font-size:12px;
  border-width: 1px;
  border-style: solid;
  border-color: #000000;
  border-collapse: collapse;
  background-color: #ffffff;
}

th
{
  border-width: 1px;
  border-bottom-style: solid;
  border-left-style: dotted;
  border-color: #000000;
  padding: 3px;
}

td
{
  border-width: 1px;
  border-style: dotted;
  border-color: #000000;
  padding: 3px;
}

tr.odd
{
  background-color: #eeeeee;
}

tr.even
{
  background-color: #ffffff;
}
```

CSS file of the example project

The following graphic shows an extract of the CSS file from the example project.

Figure 5-12



```
/*Defines the position of the background picture*/
#Background
{
    width=657;
    height=500;
}
/*Defines the position of the Temperature graphic*/
#Tempgraphics
{
    width=163;
    height=285;
    position: absolute;
    left: 250;
    top:350;
}
/*Defines the position of the Weather graphic*/
#Weathergraphics
{
    width=200;
    height=200;
    position: absolute;
    left: 250;
    top: 150;
}
/*Defines the position of the Humidity graphic*/
#Humiditygraphics
{
    width=164;
    height=164;
    position: absolute;
    left: 30;
    top: 420;
}
/*Defines the position of the Wind graphic*/
#Windgraphics
{
    width=235;
    height=235;
    position: absolute;
    left: 430;
    top: 410;
}
```

6 Installation and Operation of the Sample/Tool

Preliminary remark

To test the functionality of the tool / function block, we offer you a finished software example with test code and test parameters as download. This software example supports you in the first steps and tests.

The examples are always assigned to the components used in this document and illustrate their interaction principles. However, they are not real applications in the sense of technological problem solving with definable properties.

6.1 Installation of Hardware and Software

Hardware setup

Set up the SIMATIC hardware components according to Figure 2-1 and the item list in chapter 3. Please observe the setup guidelines for the installation of the individual components.

Installation of the standard software

During installation of the standard software on the PG/PC proceed as follows.

Table 6-1

No.	Action	Comment
1.	Install STEP 7 V5.4 + SP 1	Only necessary when working with an S7-300 or S7-400 station.
2.	Install SIMATIC NET	

Addresses

For this example, the following IP addresses were used:

Table 6-2

Hardware	IP address	MPI address
PG/PC	192.168.0.4	0
CPU 315-2 PN/DP	192.168.0.145	2

6.2 Configuration of the PG/PC

How to set the different addresses

For the SIMATIC station and the PC/PG to be located in the same LAN network, you must assign a fixed IP address to your PC/PG. To do this you open the Properties of your network card and assign an address according to Table 6-2.

Setting PG/PC interface

It is an advantage to perform the first download of the S7 program into the CPU via the MPI cable, as it is likely that its CP is not yet configured with the IP address according to Table 6-2. Via `Start -> Control Panel -> Set PC/PG interface` you adjust the S7ONLINE access point to the MPI interface of your PC/PGs.

You can alternatively establish a connection to your S7-CPU via Ethernet and the CP. You first need to assign the IP address to your CP according to Table 6-2 (e.g. the PST tool, or directly via the SIMATIC Manager).

6.3 Installation of the user program

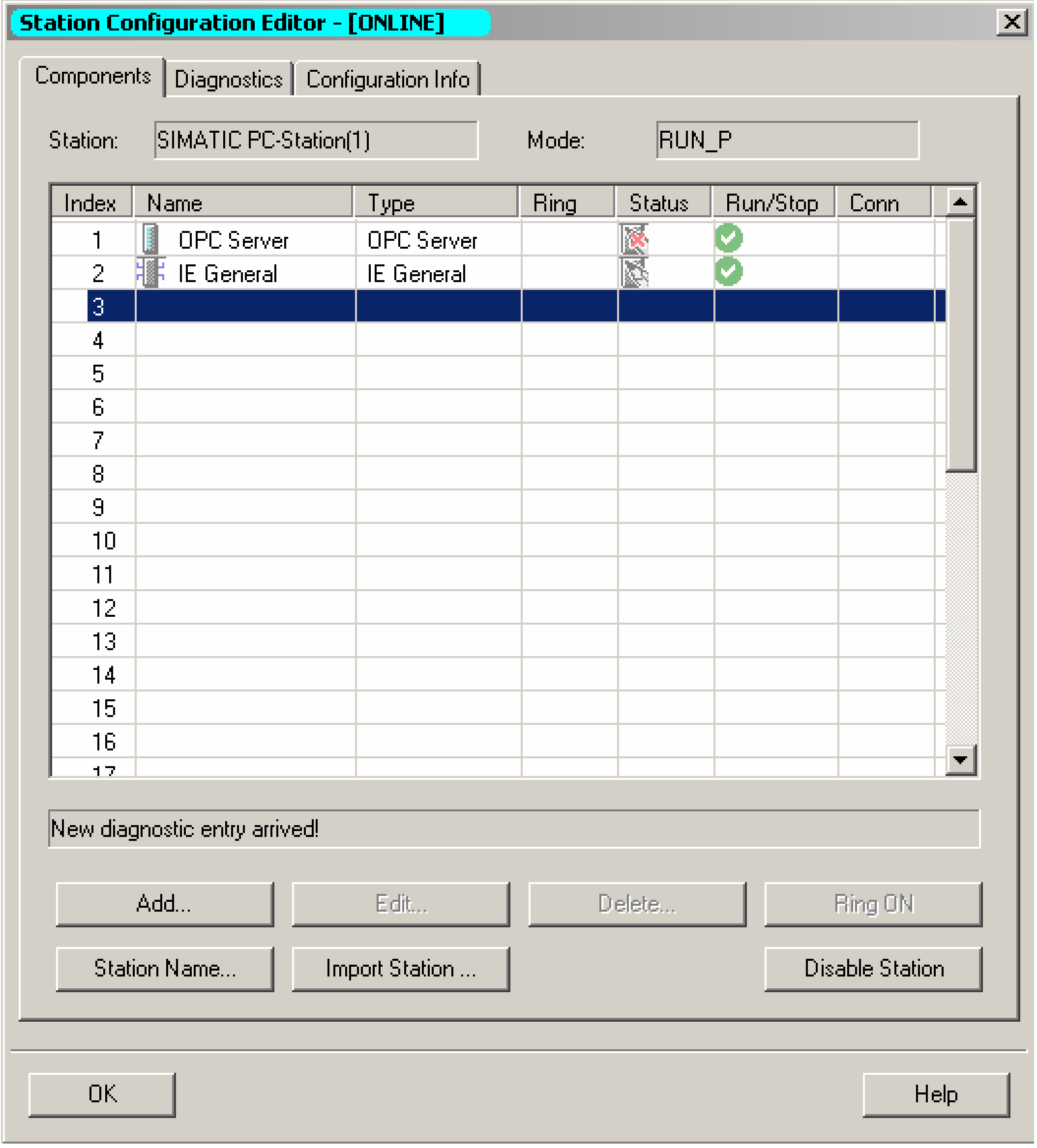
The code folder contains the following content:

- STEP 7 project **S7_html_OPC.zip**
- **htmlgen** folder with HTML Generator and example

Configuration of the Station Component Editor

Open the component editor via `Start -> Station Component Editor`. In the first line you enter an OPC server, and in the second line an IE General.

Figure 6-1



Station: SIMATIC PC-Station(1)

Mode: RUN_P

Index	Name	Type	Ring	Status	Run/Stop	Conn
1	OPC Server	OPC Server		☒	✓	
2	IE General	IE General		☒	✓	
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						

New diagnostic entry arrived!

Note

Please ensure that the station name is the same as in the STEP 7 program (**SIMATIC PC-Station(1)**). If necessary change it in the component editor.

Loading the S7 program

Open the SIMATIC MANAGER and extract the STEP 7 project. Then select the S7 station and load the entire project into your CPU. Your SIMATIC controller is now configured and contains the required user program. Set the PG/PC interface to **PC internal**.

Loading the PC station

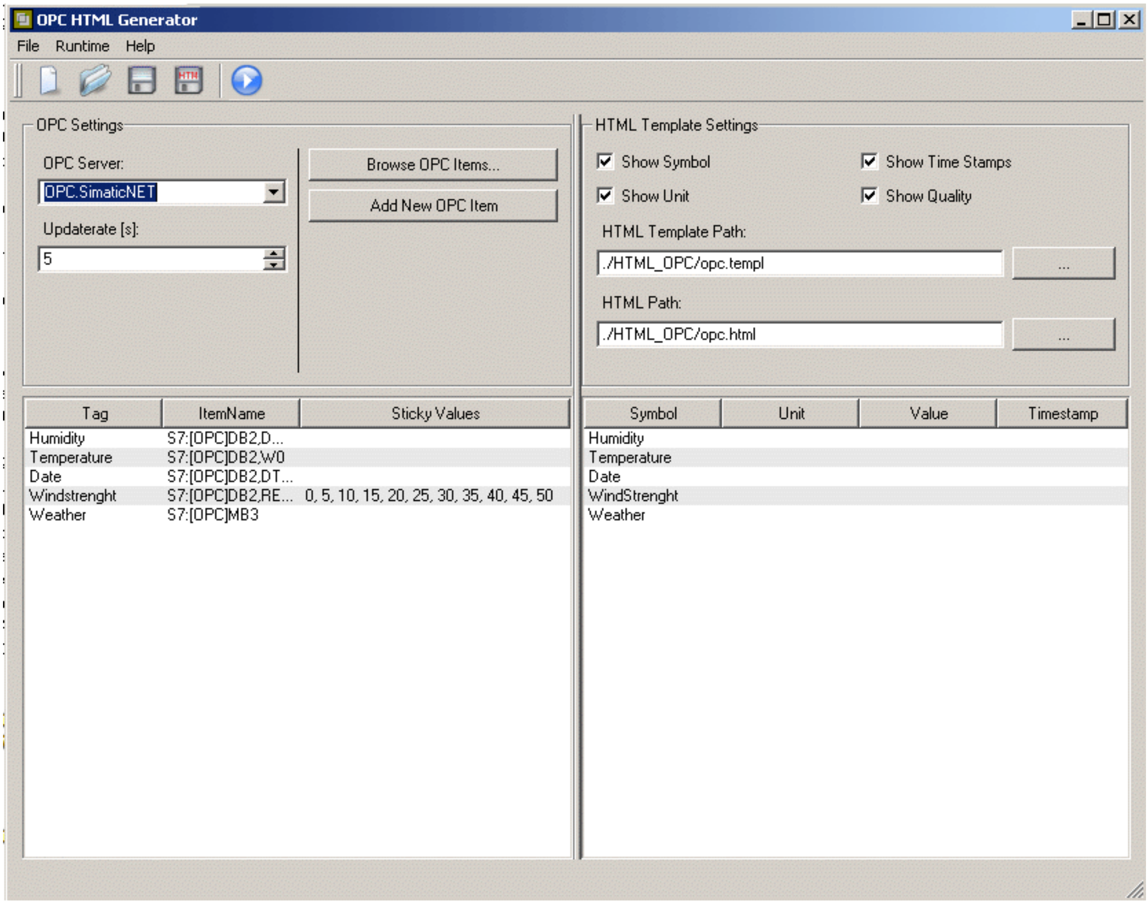
Select the PC station and load the configuration to your PC.

6.4 Operating the example

Open the **htmlgen** folder, which is contained in the Code folder of this tool.

Start the **htmlgen.exe** application. The user interface of the HTML generator appears. Via `File -> Open` you open the configuration file **Weather.ini**, which is located in the **HTML_OPC** folder in the same directory.

Figure 6-2



OPC Server: OPC.SimaticNET

Updaterate [s]: 5

☑ Show Symbol ☑ Show Time Stamps

☑ Show Unit ☑ Show Quality

HTML Template Path: ./HTML_OPC/opc.templ

HTML Path: ./HTML_OPC/opc.html

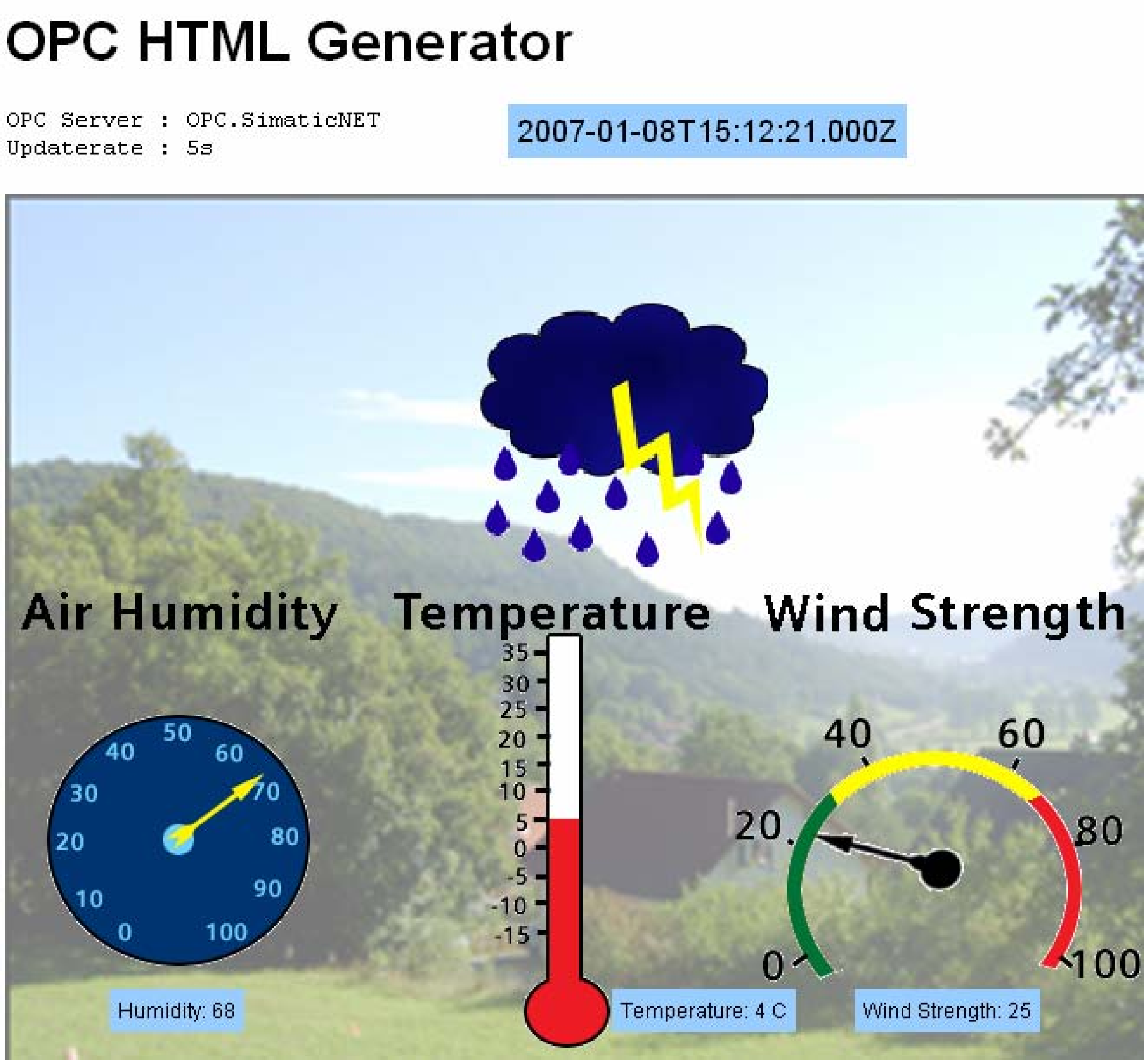
Tag	ItemName	Sticky Values
Humidity	S7:[OPC]DB2,D...	
Temperature	S7:[OPC]DB2,W0	
Date	S7:[OPC]DB2,DT...	
Windstrenght	S7:[OPC]DB2,RE...	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Weather	S7:[OPC]MB3	

Symbol	Unit	Value	Timestamp
Humidity			
Temperature			
Date			
WindStrenght			
Weather			

Operating the example program

Start the Runtime of the OPC HTML Generator and connect with the OPC server in this way. The HTML template, the CSS file, the images as well as the HTML page for this example are contained in the **HTML_OPC** folder in the same directory. The HTML page is updated in five second intervals.

Figure 6-3



OPC HTML Generator

```
OPC Server : OPC.SimaticNET
Updaterate : 5s
```

2007-01-08T15:12:21.000Z

Humidity: 68

Temperature: 4 C

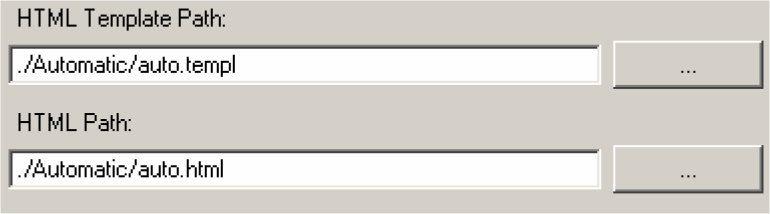
Wind Strength: 25

On this html page, five OPC variables are read and displayed. The graphics “Air Humidity“, “Temperature“, ”Wind Strength“ and “Actual Weather“ graphically depict the value for the corresponding variables.

Generating an automatically generated HTML template

Enter a storage path and name for the HTML template and the HTML page into the HTML configuration of the OPC HTML Generator.

Figure 6-4



HTML Template Path: ./Automatic/auto.templ

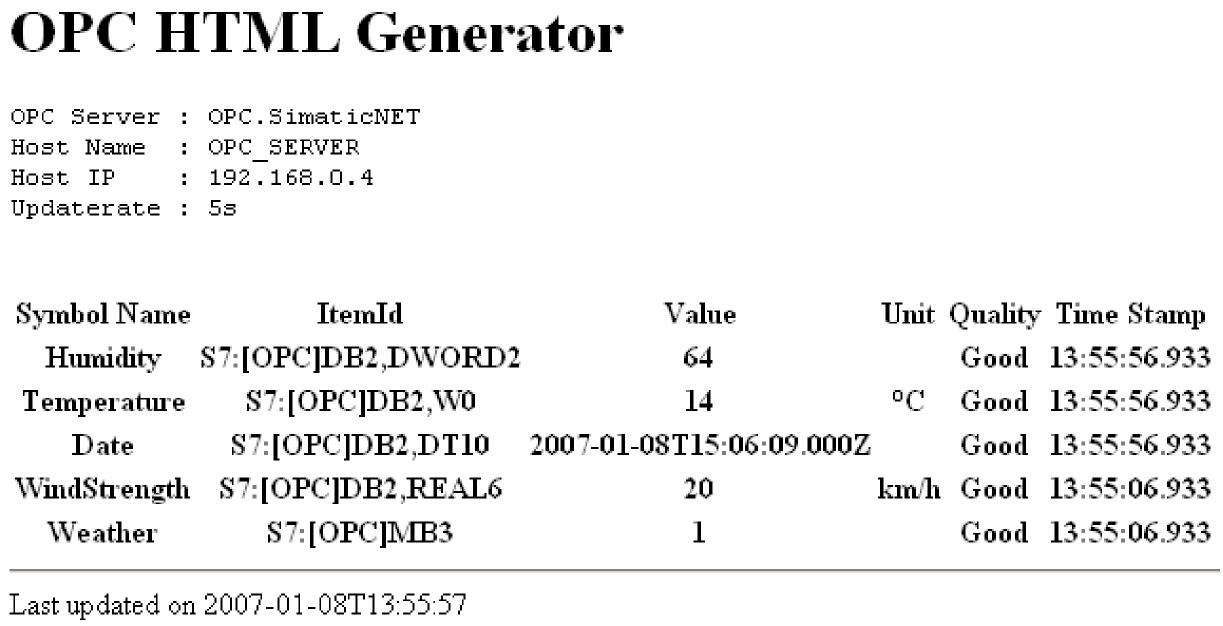
HTML Path: ./Automatic/auto.html

Note The storage path must exist or be recreated if necessary.

If you wish to monitor a different OPC variable, please add it to the list via the **Browse OPC Item…** or **Add New OPC Item** button. Edit the list as you wish (e.g. Tag-Name, Unit).

Generate a new HTML template and CSS file with the **Save HTML Template** button or at `File -> Save HTML Template`. Start the Runtime and connect with the OPC server. Change to the path specified in **HTML Path** and open the generated HTML page.

Figure 6-5



OPC HTML Generator

```
OPC Server : OPC.SimaticNET
Host Name  : OPC_SERVER
Host IP    : 192.168.0.4
Updaterate : 5s
```

Symbol Name	ItemId	Value	Unit	Quality	Time Stamp
Humidity	S7:[OPC]DB2,DWORD2	64		Good	13:55:56.933
Temperature	S7:[OPC]DB2,W0	14	°C	Good	13:55:56.933
Date	S7:[OPC]DB2,DT10	2007-01-08T15:06:09.000Z		Good	13:55:56.933
WindStrength	S7:[OPC]DB2,REAL6	20	km/h	Good	13:55:06.933
Weather	S7:[OPC]MB3	1		Good	13:55:06.933

Last updated on 2007-01-08T13:55:57

On this page, the OPC variables are depicted in form of a table. Depending on the configuration at the user interface, the columns ”Symbol Name“, ”Time Stamp“, ”Unit“ and ”Quality“ can be displayed or faded.

7 Basic Performance Data

OPC HTML Generator

Table 7-1

Criterion	Performance data	Additional note
Size of the user interface	828 x 655 pixels	
Update time	1s - 99s	Adjustable to the user interface
Number of OPC variables	100	
Sticky Values per variable	20	
Usable data types	- INT/ DINT - WORD/ DWORD/ REAL - BYTE/ BOOL - CHAR/ STRING - DATE_AND_TIME - TIME/DATE	

8 History

Table 8-1

Version	Data	Modification
V1.0	20.12.2006	First edition