HTTP Download Interface V2.0

Version 2.05, Andreas Müllauer 03.Dec 2020

1 Goal:

The download interface allows a machine to machine communication between a recordum data logging device and a data collecting computer. With the info and the stationinfo command the available parameters and information on the station can be checked, with the download command the data can be polled to the host computer.

2 Commands:

2.1 /info

[airpointer IP|Name]/cgi-bin/info.cgi
Example:
http://192.168.10.185/cgi-bin/info.cgi?loginstring=user&user_pw=user

2.2 /stationinfo

[airpointer IP|Name]/cgi-bin/stationinfo.cgi
Example:
http://192.168.10.185/cgi-bin/stationinfo.cgi?loginstring=user&user_pw=user

2.3 /gasinfo

[airpointer IP|Name]/cgi-bin/gasinfo.cgi
Example:
http://192.168.10.185/cgi-bin/gasinfo.cgi?loginstring=user&user_pw=user

2.4 /zsinfo

[airpointer IP|Name]/cgi-bin/zsinfo.cgi
Example:
http://192.168.10.185/cgi-bin/zsinfo.cgi?loginstring=user&user_pw=user

2.5 /download

[airpointer IP|Name]/cgi-bin/download.cgi
Example:
http://192.168.10.185/cgi-bin/download.cgi?loginstring=user&user_pw=user&tstart=2014-12-12,10:00:00&tend=2014-12-13,10:00:00&avg1=1,2,12705

2.6 /aircanister

[airpointer IP|Name]/cgi-bin/aircanister.cgi
Example:
http://192.168.6.127/cgi-bin/aircanister.cgi?loginstring=user&user_pw=user&tstart=2020-12-01,00:00:00&tend=2020-12-03,10:00:00&aircan=2

2.7 /zs_control

[airpointer IP|Name]/cgi-bin/zs_control.cgi
Example:
http://192.168.10.185/cgi-bin/zs_control.cgi?loginstring=user&user_pw=user?zs_cycle=all
see details here: **9 Zero / Span check control**

3 Limitations:

A maximum of 100 parameters can be polled on one request.
A maximum of 100000 datasets are answered in one request.
Up to 3 connections at the time are handled as maximum.

4 Authentication - Parameters for info and download …

GET-Parameter	Value	Description
loginstring	String	Login name of existing (recommended: low-privileged) user
user_pw	String	Password for login

Note:
If you are logged in with your browser you can work without this loginstring/user_pw for easier testing. In that case the session_id is checked. For automatic systems you always need to use the loginstring/user_pw.

5 GET - Parameters for info, stationinfo and zsinfo

GET-Parameter	Value	Description
full	none	Full set of information
quotes [opt.]	none	If set, fields are surrounded by double quotes
nohtml [opt.]	none	If set, only csv data is sent back to client, no html code
type	[return_type]	Set, how returned data should be structured: html, xml, csv, json (default:html) (case sensitive!)
del [opt.]	[Delimiter]	Field delimiter, possible values (default is: SEMI): SEMI;COMMA;TAB;SPACE (case sensitive!)
dec [opt.]	[DecimalSeparator]	Decimal separator, possible values (default is: COMMA): COMMA, POINT (case sensitive!)
Conf	None	Only on info, adds Installed, Visible and Poll to each parameter to make configuration of central software according to station easier

6 Parameters for aircanister

GET-parameter	Value	Description
tstart	YYYY-MM-DD,hh:mm:ss	Start time, Example value: 2014-09-28,10:00:00
tend	YYYY-MM-DD,hh:mm:ss	End time, Example value: 2012-09-28,12:00:00
aircan	1..8	Number of air can valve allowed 1..8
last	none	Last Sample Periode for the selected air canister
null [opt.]	String	Fill null fields with *String* (default is: NULL)
del [opt.]	[Delimiter]	Field delimiter, possible values (default is: SEMI): SEMI;COMMA;TAB;SPACE (case sensitive!)
dec [opt.]	[DecimalSeparator]	Decimal separator, possible values (default is: COMMA): COMMA, POINT(case sensitive!)
quotes [opt.]	none	If set, fields are surrounded by double quotes
nohtml [opt.]	none	If set, only csv data is sent back to client, no html code
noheader [opt.]	none	If set, only a small header is send in xml, no header is send on csv
type [opt.]	[return_type]	Set, how returned data should be structured: xml, csv, json (default: xml)
~~compr [opt.]~~	~~[compression_type]~~	~~Only valid on async 1 or 2. Possible values: zip; tgz; none (default: zip) (case sensitive!)~~

7 GET - Parameters for download

GET-Parameter	Value	Description
tstart	YYYY-MM-DD,hh:mm:ss	Start time, Example value: 2014-09-28,10:00:00
tend	YYYY-MM-DD,hh:mm:ss	End time, Example value: 2012-09-28,12:00:00
avg1 [opt.]	[P_id],[P_id],...	Parameter ids to download from average 1 source
avg2 [opt.]	[P_id],[P_id],...	Parameter ids to download from average 2 source
avg3 [opt.]	[P_id],[P_id],...	Parameter ids to download from average 3 source
null [opt.]	String	Fill null fields with *String* (default is: NULL)
del [opt.]	[Delimiter]	Field delimiter, possible values (default is: SEMI): SEMI;COMMA;TAB;SPACE (case sensitive!)
dec [opt.]	[DecimalSeparator]	Decimal separator, possible values (default is: COMMA): COMMA, POINT(case sensitive!)
interpolate [opt.]	none	If set, missing time values are interpolated
quotes [opt.]	none	If set, fields are surrounded by double quotes
nohtml [opt.]	none	If set, only csv data is sent back to client, no html code
noheader [opt.]	none	If set, only a small header is send in xml, no header is send on csv
status [opt.]	none	If set, status bytes are added to each query value
minmax [opt.]	none	If set, min/max values are added to each query value
stddev [opt.]	none	If set, standard deviations are added to each query value
sort_num [opt.]	none	If set, response is sorted according to the ID's in a numeric order If more than one average are requested the shortest average will be first.
header_name [opt.]	none	If set, header contains parameter name otherwise the ID_avg is displayed.
resume [opt.]	none	If set, a resume showing number of rows and so on is added
verbose [opt.]	none	If set, xml format is fully structured
type [opt.]	[return_type]	Set, how returned data should be structured: xml, csv, json (default: xml)
async [opt.]	[async_type]	Pseudo-asynchronous/asynchronous and compressed download: File that can then be downloaded Values: 1 or 2 (see examples below)
readystate	[filename]	Query the state of the download file. Used in asynchronous mode 2. Returns a status of OK or WAITING
remove	[filename]	Cleans up the file
compr [opt.]	[compression_type]	Only valid on async 1 or 2. Possible values: zip; tgz; none (default: zip) (case sensitive!)

8 Asynchronous Mode

8.1 Pseudo Asynchronous Mode (type 1):

After invoking the special download url, the call lasts until the file with requested measurement data has been generated. An xml document is returned that includes a filename where the composed data can be downloaded.

8.2 Fully Asynchronous Mode (type 2)

The url with download parameter selection is invoked but unlike with type 1 the request returns immediately and sends back an xml document with the filename of the data file. Afterwards consecutive calls to download should follow with the GET parameter „readystate=[filename]“. Those calls will return the status of generation (either OK or WAITING). A status OK indicates, the file is ready for download.

9 Zero / Span check control

[airpointer IP|Name]/cgi-bin/zs_control.cgi
The zero / span check can be controlled by the zs_control.cgi followed by a command and the indicator for the module/instrument involved, separated by a comma. With the keyword “all” all modules/instruments are switched. You can find all available indicators using the zsinfo.cgi command.

9.1 zero

Example:
http://192.168.10.185/cgi-bin/zs_control.cgi?loginstring=user&user_pw=user?zero=LS01,LS03

If the module/instrument stays in zero the same time (default 720sec) or longer as configured for a zero check, zero values are calculated and stored in the database.
Response give an error number + a error message.
Example no error: `Error 0: OK`
Example error: `Error 126: TAG not found LS04`

9.2 span

Example:
http://192.168.10.185/cgi-bin/zs_control.cgi?loginstring=user&user_pw=user?span=LS01,LS03

If the module/instrument stays in span the same time (default 720sec) or longer as configured for a span check, span values are calculated and stored in the database.
Response give an error number + a error message.
Example no error: `Error 0: OK`
Example error: `Error 126: TAG not found LS04`

9.3 sample

Example:
http://192.168.10.185/cgi-bin/zs_control.cgi?loginstring=user&user_pw=user?sample=LS01,LS03

The module/instrument is switched back to sample, the data is not stored until the configured purge out time is over (default: 180 seconds)
Response give an error number + a error message.
Example no error: `Error 0: OK`
Example error: `Error 126: TAG not found LS04`

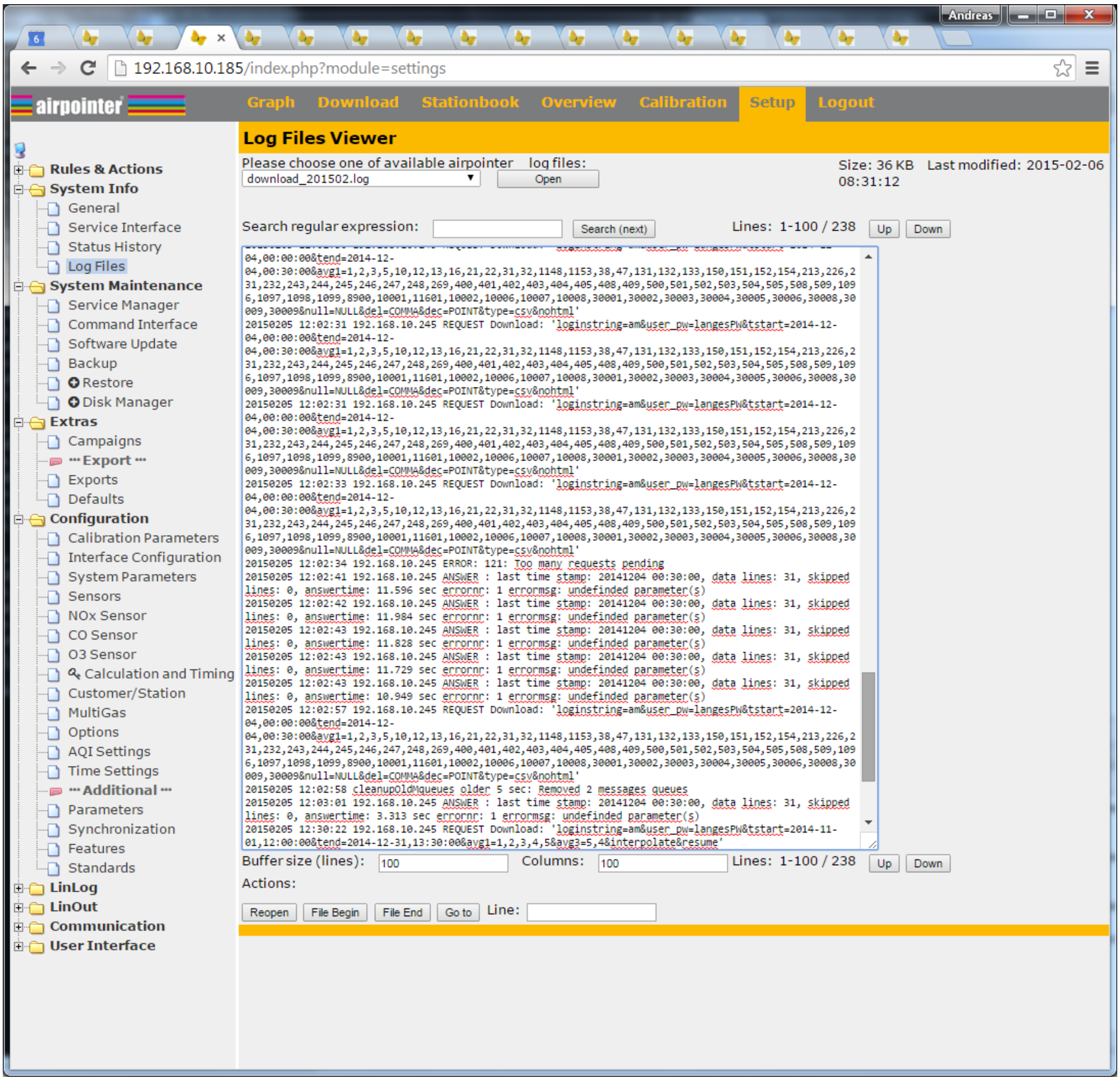
9.4 zs_cycle

Example:
http://192.168.10.185/cgi-bin/zs_control.cgi?loginstring=user&user_pw=user?zs_cycle=LS01,LS03

The module/instrument performs a zero/span cycle as configured and stores the calibration check data.
Response give an error number + a error message.
Example no error: `Error 0: OK`
Example error: `Error 126: TAG not found LS04`

10 Log File

For easier debugging each request is documented in a monthly log file. The file name is download_YYMM.log, the file can be checked using the Log File Viewer in Setup-> System Info-Log Files.



Log Files Viewer

Please choose one of available airpointer log files: download_201502.log

Size: 36 KB Last modified: 2015-02-06 08:31:12

Search regular expression:

Lines: 1-100 / 238

```
04,00:00:00&tend=2014-12-
04,00:30:00&avg1=1,2,3,5,10,12,13,16,21,22,31,32,1148,1153,38,47,131,132,133,150,151,152,154,213,226,2
31,232,243,244,245,246,247,248,269,400,401,402,403,404,405,408,409,500,501,502,503,504,505,508,509,109
6,1097,1098,1099,8900,10001,11601,10002,10006,10007,10008,30001,30002,30003,30004,30005,30006,30008,30
009,30009&null=NULL&del=COMMA&dec=POINT&type=csv&nohtml'
20150205 12:02:31 192.168.10.245 REQUEST Download: 'loginstring=am&user_pw=langesPW&tstart=2014-12-
04,00:00:00&tend=2014-12-
04,00:30:00&avg1=1,2,3,5,10,12,13,16,21,22,31,32,1148,1153,38,47,131,132,133,150,151,152,154,213,226,2
31,232,243,244,245,246,247,248,269,400,401,402,403,404,405,408,409,500,501,502,503,504,505,508,509,109
6,1097,1098,1099,8900,10001,11601,10002,10006,10007,10008,30001,30002,30003,30004,30005,30006,30008,30
009,30009&null=NULL&del=COMMA&dec=POINT&type=csv&nohtml'
20150205 12:02:31 192.168.10.245 REQUEST Download: 'loginstring=am&user_pw=langesPW&tstart=2014-12-
04,00:00:00&tend=2014-12-
04,00:30:00&avg1=1,2,3,5,10,12,13,16,21,22,31,32,1148,1153,38,47,131,132,133,150,151,152,154,213,226,2
31,232,243,244,245,246,247,248,269,400,401,402,403,404,405,408,409,500,501,502,503,504,505,508,509,109
6,1097,1098,1099,8900,10001,11601,10002,10006,10007,10008,30001,30002,30003,30004,30005,30006,30008,30
009,30009&null=NULL&del=COMMA&dec=POINT&type=csv&nohtml'
20150205 12:02:33 192.168.10.245 REQUEST Download: 'loginstring=am&user_pw=langesPW&tstart=2014-12-
04,00:00:00&tend=2014-12-
04,00:30:00&avg1=1,2,3,5,10,12,13,16,21,22,31,32,1148,1153,38,47,131,132,133,150,151,152,154,213,226,2
31,232,243,244,245,246,247,248,269,400,401,402,403,404,405,408,409,500,501,502,503,504,505,508,509,109
6,1097,1098,1099,8900,10001,11601,10002,10006,10007,10008,30001,30002,30003,30004,30005,30006,30008,30
009,30009&null=NULL&del=COMMA&dec=POINT&type=csv&nohtml'
20150205 12:02:34 192.168.10.245 ERROR: 121: Too many requests pending
20150205 12:02:41 192.168.10.245 ANSWER : last time stamp: 20141204 00:30:00, data lines: 31, skipped
lines: 0, answertime: 11.596 sec errornr: 1 errormsg: undefinded parameter(s)
20150205 12:02:42 192.168.10.245 ANSWER : last time stamp: 20141204 00:30:00, data lines: 31, skipped
lines: 0, answertime: 11.984 sec errornr: 1 errormsg: undefinded parameter(s)
20150205 12:02:43 192.168.10.245 ANSWER : last time stamp: 20141204 00:30:00, data lines: 31, skipped
lines: 0, answertime: 11.828 sec errornr: 1 errormsg: undefinded parameter(s)
20150205 12:02:43 192.168.10.245 ANSWER : last time stamp: 20141204 00:30:00, data lines: 31, skipped
lines: 0, answertime: 11.729 sec errornr: 1 errormsg: undefinded parameter(s)
20150205 12:02:43 192.168.10.245 ANSWER : last time stamp: 20141204 00:30:00, data lines: 31, skipped
lines: 0, answertime: 10.949 sec errornr: 1 errormsg: undefinded parameter(s)
20150205 12:02:57 192.168.10.245 REQUEST Download: 'loginstring=am&user_pw=langesPW&tstart=2014-12-
04,00:00:00&tend=2014-12-
04,00:30:00&avg1=1,2,3,5,10,12,13,16,21,22,31,32,1148,1153,38,47,131,132,133,150,151,152,154,213,226,2
31,232,243,244,245,246,247,248,269,400,401,402,403,404,405,408,409,500,501,502,503,504,505,508,509,109
6,1097,1098,1099,8900,10001,11601,10002,10006,10007,10008,30001,30002,30003,30004,30005,30006,30008,30
009,30009&null=NULL&del=COMMA&dec=POINT&type=csv&nohtml'
20150205 12:02:58 cleanupOldMqueues older 5 sec: Removed 2 messages queues
20150205 12:03:01 192.168.10.245 ANSWER : last time stamp: 20141204 00:30:00, data lines: 31, skipped
lines: 0, answertime: 3.313 sec errornr: 1 errormsg: undefinded parameter(s)
20150205 12:30:22 192.168.10.245 REQUEST Download: 'loginstring=am&user_pw=langesPW&tstart=2014-11-
01,12:00:00&tend=2014-12-31,13:30:00&avg1=1,2,3,4,5&avg3=5,4&interpolate&resume'
```

Buffer size (lines): 100 Columns: 100 Lines: 1-100 / 238

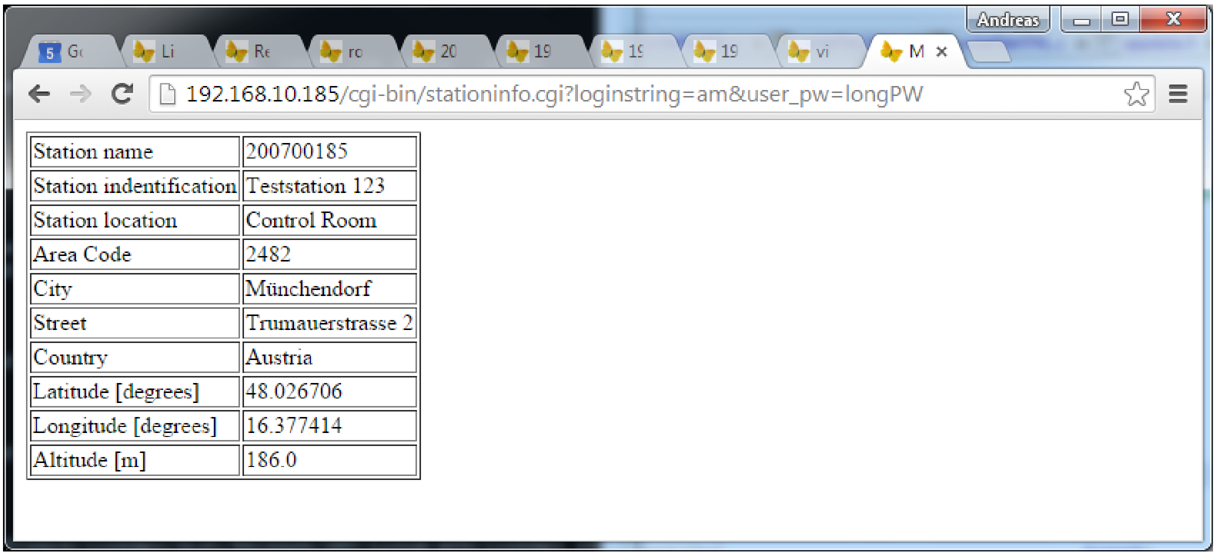
Actions:

Line:

11 Examples

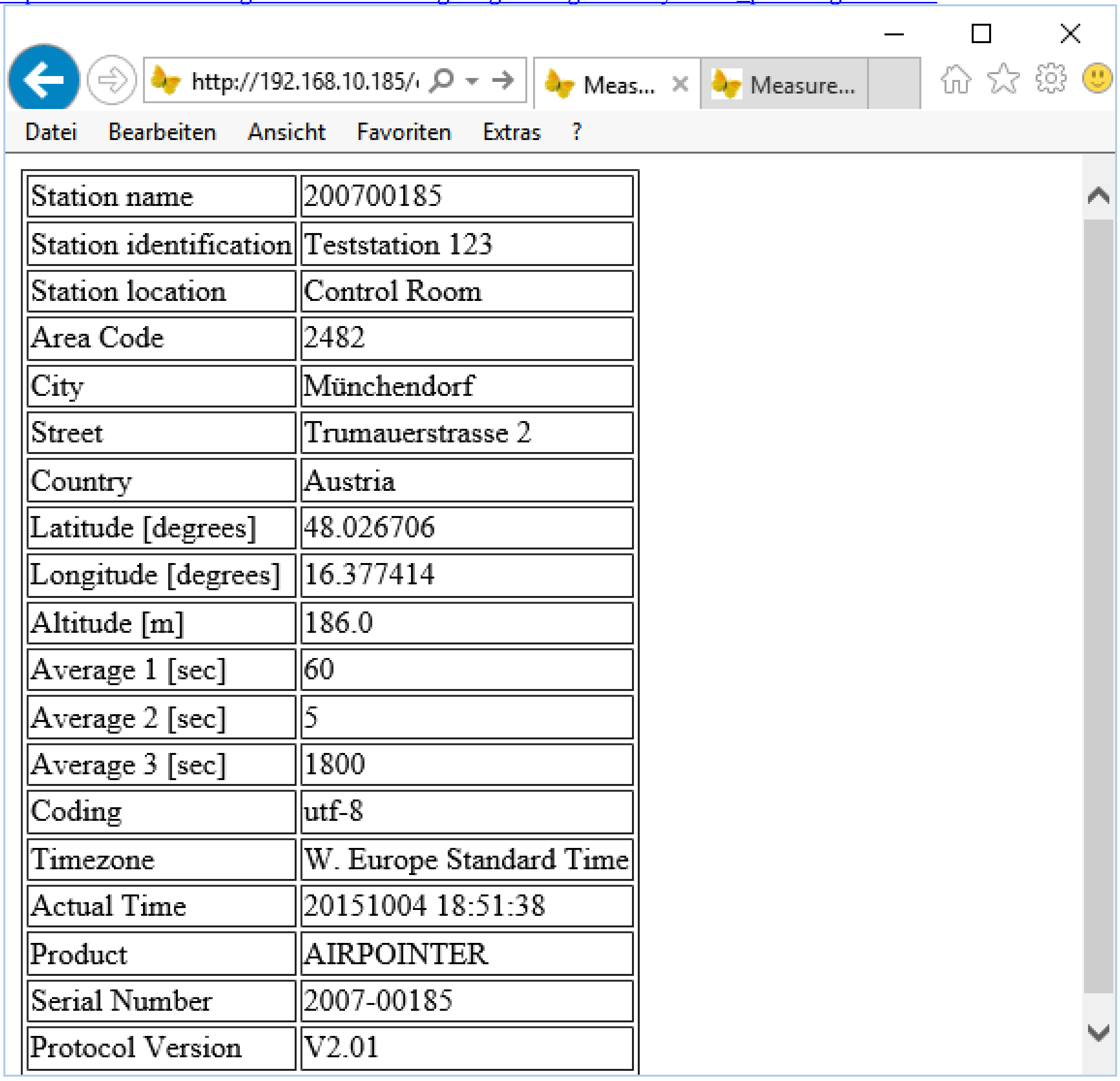
11.1 Stationinfo

With the station info command you request information about the measuring station:
http://192.168.10.185/cgi-bin/stationinfo.cgi?loginstring=dummy&user_pw=longPW
Answer:


Andreas
Gc
Li
Re
rc
20
19
19
19
vi
M
192.168.10.185/cgi-bin/stationinfo.cgi?loginstring=am&user_pw=longPW

Station name	200700185
Station indentification	Teststation 123
Station location	Control Room
Area Code	2482
City	Münchendorf
Street	Trumauerstrasse 2
Country	Austria
Latitude [degrees]	48.026706
Longitude [degrees]	16.377414
Altitude [m]	186.0

http://192.168.10.185/cgi-bin/stationinfo.cgi?loginstring=dummy&user_pw=longPW&full



Station name	200700185
Station identification	Teststation 123
Station location	Control Room
Area Code	2482
City	Münchendorf
Street	Trumauerstrasse 2
Country	Austria
Latitude [degrees]	48.026706
Longitude [degrees]	16.377414
Altitude [m]	186.0
Average 1 [sec]	60
Average 2 [sec]	5
Average 3 [sec]	1800
Coding	utf-8
Timezone	W. Europe Standard Time
Actual Time	20151004 18:51:38
Product	AIRPOINTER
Serial Number	2007-00185
Protocol Version	V2.01

You also can add ‘&type=csv’ or ‘&type=xml’ to get the response in that format, but default is html.

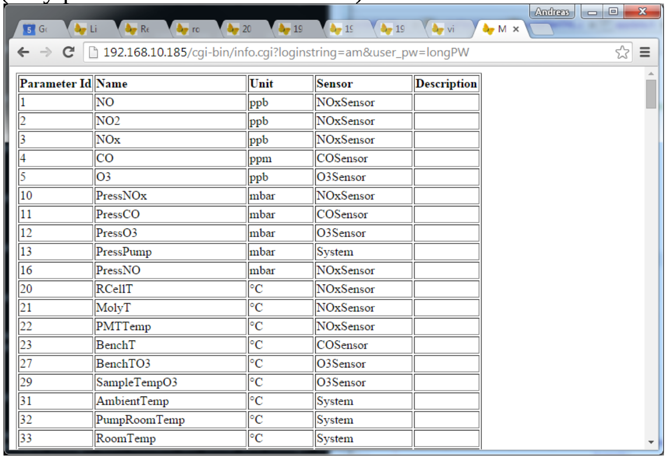
http://192.168.10.185/cgi-bin/stationinfo.cgi?loginstring=dummy&user_pw=longPW&full&type=xml

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<AirpointerStationInfoData>
        <Station_name>200700185</Station_name>
        <Station_identification>Teststation 123</Station_identification>
        <Station_location>Control Room</Station_location>
        <Area_Code>2482</Area_Code>
        <City>Münchendorf</City>
        <Street>Trumauerstrasse 2</Street>
        <Country>Austria</Country>
        <Latitude>48.026706</Latitude>
        <Longitude>16.377414</Longitude>
        <Altitude>186.0</Altitude>
        <Average_1>60</Average_1>
        <Average_2>5</Average_2>
        <Average_3>1800</Average_3>
        <Coding>utf-8</Coding>
        <Timezone>W. Europe Standard Time</Timezone>
        <Actual_Time>20151004 14:40:53</Actual_Time>
        <Product>AIRPOINTER</Product>
        <Serial_Number>2007-00185</Serial_Number>
        <Protocol_Version>V2.01</Protocol_Version>
</AirpointerStationInfoData>
```

11.2 Info

With the info command you can request all parameters available in the station.

http://192.168.10.185/cgi-bin/info.cgi?loginstring=dummy&user_pw=longPW
(only part of the answer is shown)

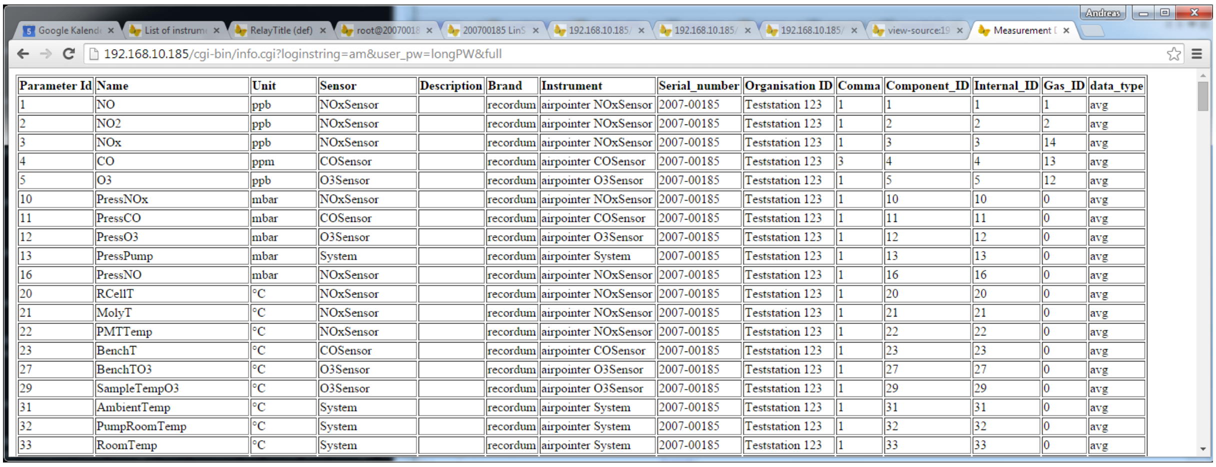


192.168.10.185/cgi-bin/info.cgi?loginstring=am&user_pw=longPW

Parameter Id	Name	Unit	Sensor	Description
1	NO	ppb	NOxSensor	
2	NO2	ppb	NOxSensor	
3	NOx	ppb	NOxSensor	
4	CO	ppm	COSensor	
5	O3	ppb	O3Sensor	
10	PressNOx	mbar	NOxSensor	
11	PressCO	mbar	COSensor	
12	PressO3	mbar	O3Sensor	
13	PressPump	mbar	System	
16	PressNO	mbar	NOxSensor	
20	RCellT	°C	NOxSensor	
21	MolyT	°C	NOxSensor	
22	PMTTemp	°C	NOxSensor	
23	BenchT	°C	COSensor	
27	BenchTO3	°C	O3Sensor	
29	SampleTempO3	°C	O3Sensor	
31	AmbientTemp	°C	System	
32	PumpRoomTemp	°C	System	
33	RoomTemp	°C	System	

You find the Parameter Id the system has given or you have defined in Setup->Configuration->Parameter. With this Id you can request data.

Adding '&full' gives more information on each parameter
http://192.168.10.185/cgi-bin/info.cgi?loginstring=dummy&user_pw=longPW&full
(only part of the answer is shown)



192.168.10.185/cgi-bin/info.cgi?loginstring=am&user_pw=longPW&full

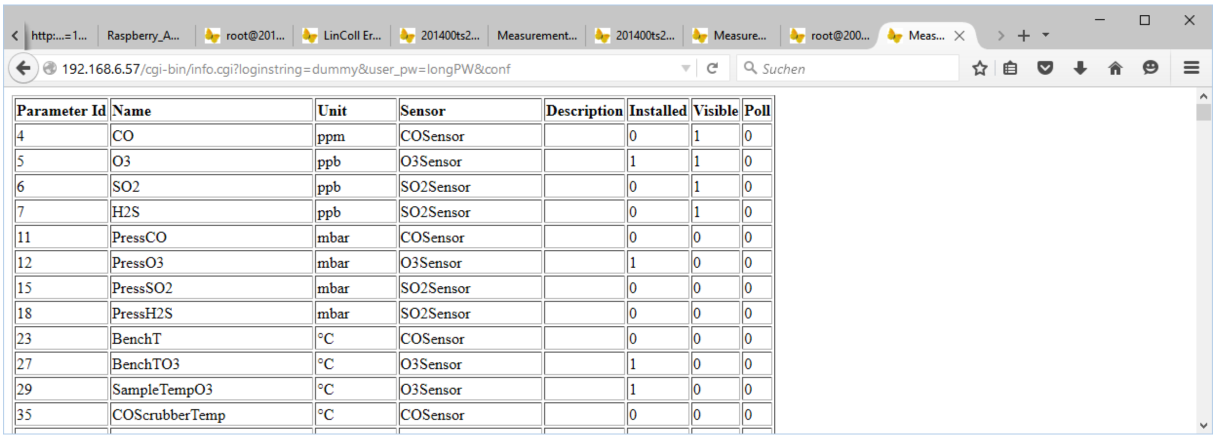
Parameter Id	Name	Unit	Sensor	Description	Brand	Instrument	Serial_number	Organisation ID	Comma	Component_ID	Internal_ID	Gas_ID	data_type
1	NO	ppb	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	1	1	1	avg
2	NO2	ppb	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	2	2	2	avg
3	NOx	ppb	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	3	3	14	avg
4	CO	ppm	COSensor		recordum	airpointer COSensor	2007-00185	Teststation 123	3	4	4	13	avg
5	O3	ppb	O3Sensor		recordum	airpointer O3Sensor	2007-00185	Teststation 123	1	5	5	12	avg
10	PressNOx	mbar	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	10	10	0	avg
11	PressCO	mbar	COSensor		recordum	airpointer COSensor	2007-00185	Teststation 123	1	11	11	0	avg
12	PressO3	mbar	O3Sensor		recordum	airpointer O3Sensor	2007-00185	Teststation 123	1	12	12	0	avg
13	PressPump	mbar	System		recordum	airpointer System	2007-00185	Teststation 123	1	13	13	0	avg
16	PressNO	mbar	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	16	16	0	avg
20	RCellT	°C	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	20	20	0	avg
21	MolyT	°C	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	21	21	0	avg
22	PMTTemp	°C	NOxSensor		recordum	airpointer NOxSensor	2007-00185	Teststation 123	1	22	22	0	avg
23	BenchT	°C	COSensor		recordum	airpointer COSensor	2007-00185	Teststation 123	1	23	23	0	avg
27	BenchTO3	°C	O3Sensor		recordum	airpointer O3Sensor	2007-00185	Teststation 123	1	27	27	0	avg
29	SampleTempO3	°C	O3Sensor		recordum	airpointer O3Sensor	2007-00185	Teststation 123	1	29	29	0	avg
31	AmbientTemp	°C	System		recordum	airpointer System	2007-00185	Teststation 123	1	31	31	0	avg
32	PumpRoomTemp	°C	System		recordum	airpointer System	2007-00185	Teststation 123	1	32	32	0	avg
33	RoomTemp	°C	System		recordum	airpointer System	2007-00185	Teststation 123	1	33	33	0	avg

Besides the self explaining columns, some needs to be explained: The Parameter Id is explained above, the Internal_Id is the original Id given by the system. On parameters belonging to another parameter like NO_Zero and NO_Span have the Component ID directing to the Internal ID of that parameter. The data_type shows if the parameter is an avg (average), zero or span value. The gas_ID shows what kind of gas is measured here according to the recordum gas table.

You also can add '&type=csv' or '&type=xml' to get the response in that format.
http://192.168.10.185/cgi-bin/info.cgi?loginstring=dummy&user_pw=longPW&full&type=csv
(only part of the answer is shown)

```
Parameter_Id;Name;Unit;Sensor;Description;Brand;Instrument;Serial_number;Organisation_ID;Comma;Component_ID;Int
ernal_ID;Gas_ID;data_type
1;NO;ppb;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;1;1;1;avg
2;NO2;ppb;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;2;2;2;avg
3;NOx;ppb;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;3;3;14;avg
4;CO;ppm;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;3;4;4;13;avg
5;O3;ppb;O3Sensor; ;recordum;airpointer O3Sensor;2007-00185;Teststation 123;1;5;5;12;avg
10;PressNOx;mbar;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;10;10;0;avg
11;PressCO;mbar;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;1;11;11;0;avg
12;PressO3;mbar;O3Sensor; ;recordum;airpointer O3Sensor;2007-00185;Teststation 123;1;12;12;0;avg
13;PressPump;mbar;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;13;13;0;avg
16;PressNO;mbar;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;16;16;0;avg
20;RCellT;Â°C;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;20;20;0;avg
21;MolyT;Â°C;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;21;21;0;avg
22;PMTTemp;Â°C;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;22;22;0;avg
23;BenchT;Â°C;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;1;23;23;0;avg
27;BenchTO3;Â°C;O3Sensor; ;recordum;airpointer O3Sensor;2007-00185;Teststation 123;1;27;27;0;avg
29;SampleTempO3;Â°C;O3Sensor; ;recordum;airpointer O3Sensor;2007-00185;Teststation 123;1;29;29;0;avg
31;AmbientTemp;Â°C;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;31;31;0;avg
32;PumpRoomTemp;Â°C;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;32;32;0;avg
33;RoomTemp;Â°C;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;33;33;0;avg
34;CoolerOutTemp;Â°C;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;34;34;0;avg
35;COScrubberTemp;Â°C;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;1;35;35;0;avg
38;Temp_PC;Â°C;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;38;38;0;avg
47;TempChipWatchdog;Â°C;System; ;recordum;airpointer System;2007-00185;Teststation 123;1;47;47;0;avg
60;PMTSigNO;Hz;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;60;60;0;avg
61;PMTSigNOx;Hz;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;61;61;0;avg
62;PMTSigAuto0;Hz;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;62;62;0;avg
63;COMeas;mV;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;1;63;63;0;avg
64;CORef;mV;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;1;64;64;0;avg
65;CORatio;-;COSensor; ;recordum;airpointer COSensor;2007-00185;Teststation 123;4;65;65;0;avg
69;ClimaActMode;%;System; ;recordum;airpointer System;2007-00185;Teststation 123;0;69;69;0;avg
70;FanSampleRPM;rpm;System; ;recordum;airpointer System;2007-00185;Teststation 123;0;70;70;0;avg
71;FanPumpRoomRPM;rpm;System; ;recordum;airpointer System;2007-00185;Teststation 123;0;71;71;0;avg
76;HVPS_NOx;V;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;0;76;76;0;avg
80;PowerToRCell;%;NOxSensor; ;recordum;airpointer NOxSensor;2007-00185;Teststation 123;1;80;80;0;avg
```

Another option is &conf this make an automatic configuration of a central software easier.



192.168.6.57/cgi-bin/info.cgi?loginstring=dummy&user_pw=longPW&conf

Parameter Id	Name	Unit	Sensor	Description	Installed	Visible	Poll
4	CO	ppm	COSensor		0	1	0
5	O3	ppb	O3Sensor		1	1	0
6	SO2	ppb	SO2Sensor		0	1	0
7	H2S	ppb	SO2Sensor		0	1	0
11	PressCO	mbar	COSensor		0	0	0
12	PressO3	mbar	O3Sensor		1	0	0
15	PressSO2	mbar	SO2Sensor		0	0	0
18	PressH2S	mbar	SO2Sensor		0	0	0
23	BenchT	°C	COSensor		0	0	0
27	BenchTO3	°C	O3Sensor		1	0	0
29	SampleTempO3	°C	O3Sensor		1	0	0
35	COScrubberTemp	°C	COSensor		0	0	0

Three additional parameter are transmitted. Installed show if the sensor is actually build in. Visible is a value taken from the public column, this shows if a parameter is a result of a measurement or a less important value like a temperature inside the sensor. Poll a parameter for future (when written in Oct. 2015) to allow the user to define the data to be transmitted.

11.3 Gasinfo

Because in most cases customer can freely choose the name of a parameter it is not clear what gas is shown behind a name at least for a machine. To make this possible parameters can be marked with a so called Gas Id. This unique id links to the gas_definitions table in the database. In this table the typically used gases, but also parameters like PM10 has an entry. Customers can add his own gas definitions to this table (at the moment only on airQrate calibrators). These entries are shown in the “User defined gas” column. By our definition gas_ids below 10000 are defined by us and used ids are fixed, but we can add new ones. This means all recordum instruments will use the same gas_id, only user defined gas can be different.
With the gasinfo command you can request the data of the gas_defintions table available in the station.

http://192.168.10.185/cgi-bin/gasinfo.cgi?loginstring=dummy&user_pw=longPW

Gas Id	Short	Name
1	NO	Nitrogen monoxide
2	NO2	Nitrogen dioxide
3	CO2	Carbon dioxide
4	NH3	Ammoniak
5	H2S	Hydrogen sulfide
6	BEN	Benzene

(Only part of the answer is shown)

Adding ‘&full’ gives more information on each gas:
http://192.168.10.185/cgi-bin/gasinfo.cgi?loginstring=dummy&user_pw=longPW&full

Gas Id	Short	Name	g/mol	Description	User defined gas
1	NO	Nitrogen monoxide	30.01	Nitrogen monoxide	0
2	NO2	Nitrogen dioxide	46.01	Nitrogen dioxide	0
3	CO2	Carbon dioxide	44.01	Carbon dioxide	0
4	NH3	Ammoniak	17.04	Ammoniak	0
5	H2S	Hydrogen sulfide	34.08	Hydrogen sulfide	0
6	BEN	Benzene	78.00	Benzene	0
7	TOL	Toluene	91.92	Toluene	0
8	XYL	Xylene	105.84	Xylene	0
9	CH4	Methane	16.04	Methane	0
10	C4H10	Butane	58.13	Butane	0

11	SO2	Sulfur dioxide	64.06	Sulfur dioxide	0
12	O3	Ozone	48.00	Ozone	0
13	CO	Carbon monoxide	28.01	Carbon monoxide	0
14	NOX		-		0
15	TN		-		0
16	TRS		-		0
17	TNX		-		0
18	THC		-		0
19	ETHYLBEN	Ethylbenzene	106.17	C8H10	0
20	MP-XYL	mpXylene	106.16	C8H10	0
21	O-XYL	oXylene	106.16	C8H10	0
22	non-CH4		-		0
9000	PM-TSP	total suspended particles	-	total suspended particles	0
9001	PM10		-		0
9002	PM4		-		0
9003	PM2.5		-		0
9004	PM1		-		0

You also can add ‘&type=csv’ or ‘&type=xml’ to get the response in that format, but default is html.

11.4 Download

We are working only with a small number of data here in this example to show the different possibilities. We have chosen ID 1(NO), ID 2(NO2) and ID 5(O3) to be downloaded in different ways from our testing unit.

http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2

```
Time;5_3;1_3;2_3
2015-01-31 12:00:00;-0.0;-0.3;0.1
2015-01-31 12:30:00;-0.0;-0.1;-0.0
2015-01-31 13:00:00;-2.0;-0.0;4.0
2015-01-31 13:30:00;-0.1;-0.2;1.4
2015-01-31 14:00:00;-0.1;-0.1;0.1
```

The ID’s are answered in the same order they are requested. If an ID is not present in a system the column is filled up with NULL/Missing values.
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2,32000

```
Time;5_3;1_3;2_3;32000_3
2015-01-31 12:00:00;-0.0;-0.3;0.1;-9999
2015-01-31 12:30:00;-0.0;-0.1;-0.0;-9999
2015-01-31 13:00:00;-2.0;-0.0;4.0;-9999
2015-01-31 13:30:00;-0.1;-0.2;1.4;-9999
2015-01-31 14:00:00;-0.1;-0.1;0.1;-9999
```

Also different averages can be polled in one request:
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:30:00&tend=2015-01-31,13:00:00&avg1=5&avg3=5,1,2,32000

```
Time;5_1;5_3;1_3;2_3;32000_3
2015-01-31 12:30:00;0.1;-0.0;-0.1;-0.0;-9999
2015-01-31 12:46:00;-4.6;-9999;-9999;-9999;-9999
2015-01-31 12:47:00;-4.0;-9999;-9999;-9999;-9999
2015-01-31 12:48:00;-3.5;-9999;-9999;-9999;-9999
2015-01-31 12:49:00;-3.1;-9999;-9999;-9999;-9999
2015-01-31 12:50:00;-2.7;-9999;-9999;-9999;-9999
2015-01-31 12:51:00;-2.5;-9999;-9999;-9999;-9999
2015-01-31 12:52:00;-2.3;-9999;-9999;-9999;-9999
2015-01-31 12:53:00;-2.1;-9999;-9999;-9999;-9999
```

```
2015-01-31 12:54:00;-1.4;-9999;-9999;-9999;-9999
2015-01-31 12:55:00;-0.9;-9999;-9999;-9999;-9999
2015-01-31 12:56:00;-0.7;-9999;-9999;-9999;-9999
2015-01-31 12:57:00;-0.5;-9999;-9999;-9999;-9999
2015-01-31 12:58:00;-0.5;-9999;-9999;-9999;-9999
2015-01-31 12:59:00;-0.4;-9999;-9999;-9999;-9999
2015-01-31 13:00:00;-0.4;-2.0;-0.0;4.0;-9999
```

11.4.1 Interpolate

We see here in the example above that there are some minute values missing (see highlighted yellow), with the option '&interpolate' we can fill up the table with missing values (-9999).
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:30:00&tend=2015-01-31,13:00:00&avg1=5&avg3=5,1,2,32000&interpolate

```
Time;5_1;5_3;1_3;2_3;32000_3
2015-01-31 12:30:00;0.1;-0.0;-0.1;-0.0;-9999
2015-01-31 12:31:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:32:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:33:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:34:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:35:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:36:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:37:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:38:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:39:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:40:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:41:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:42:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:43:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:44:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:45:00;-9999.0;-9999;-9999;-9999;-9999
2015-01-31 12:46:00;-4.6;-9999;-9999;-9999;-9999
2015-01-31 12:47:00;-4.0;-9999;-9999;-9999;-9999
2015-01-31 12:48:00;-3.5;-9999;-9999;-9999;-9999
2015-01-31 12:49:00;-3.1;-9999;-9999;-9999;-9999
2015-01-31 12:50:00;-2.7;-9999;-9999;-9999;-9999
2015-01-31 12:51:00;-2.5;-9999;-9999;-9999;-9999
2015-01-31 12:52:00;-2.3;-9999;-9999;-9999;-9999
2015-01-31 12:53:00;-2.1;-9999;-9999;-9999;-9999
2015-01-31 12:54:00;-1.4;-9999;-9999;-9999;-9999
2015-01-31 12:55:00;-0.9;-9999;-9999;-9999;-9999
2015-01-31 12:56:00;-0.7;-9999;-9999;-9999;-9999
2015-01-31 12:57:00;-0.5;-9999;-9999;-9999;-9999
2015-01-31 12:58:00;-0.5;-9999;-9999;-9999;-9999
2015-01-31 12:59:00;-0.4;-9999;-9999;-9999;-9999
2015-01-31 13:00:00;-0.4;-2.0;-0.0;4.0;-9999
```

11.4.2 Sorting

Maybe you want to have always a numeric order for our ID's you can do this with the option '&sort_num'
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg1=5&avg3=5,1,2,32000&interpolate&sort_num

(only part of answer is shown)

```
Time;1_3;2_3;5_1;5_3;32000_3
2015-01-31 12:00:00;-0.3;0.1;-0.1;-0.0;-9999
2015-01-31 12:01:00;-9999;-9999;-0.1;-9999;-9999
2015-01-31 12:02:00;-9999;-9999;-0.1;-9999;-9999
2015-01-31 12:03:00;-9999;-9999;-0.1;-9999;-9999
2015-01-31 12:04:00;-9999;-9999;-0.1;-9999;-9999
2015-01-31 12:05:00;-9999;-9999;-0.1;-9999;-9999
```

Be aware first sorting priority is ID, second priority is average number.

11.4.3 Status

Let's go back to the first example and add status information with '&status' to it:
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&status

```
Time;5_3;5_3_ss;5_3_bs;5_3_fs;5_3_nval;1_3;1_3_ss;1_3_bs;1_3_fs;1_3_nval;2_3;2_3_ss;2_3_bs;2_3_fs;2_3_nval
2015-01-31 12:00:00;-0.0;0;0;0;1800;-0.3;0;0;0;1800;0.1;0;0;0;1800
2015-01-31 12:30:00;-0.0;0;0;0;1800;-0.1;0;0;0;1800;-0.0;0;0;0;1800
2015-01-31 13:00:00;-2.0;0;10;0;899;-0.0;0;14;0;178;4.0;0;14;0;178
2015-01-31 13:30:00;-0.1;0;0;0;1800;-0.2;0;0;0;1800;1.4;0;0;0;1800
2015-01-31 14:00:00;-0.1;0;0;0;1800;-0.1;0;0;0;1800;0.1;0;0;0;1800
```

We now have additional columns:
_ss stands for system status
_bs stands for “Betriebsstatus” a German word meaning operational status
_fs stands for failure status
_nval means number of 1 second values averaged to that value

Because we have a 30 minute average here nval should be 1800, depending on the regulation a value can be taken as valid or not. Very common is that 75% of data needs to be present for a valid average. In this example we see at 13:30 that we are missing values and also that the operational status was active.

11.4.4 MinMax / Standard Deviation

To see what was going on during an average minimum, maximum and standard deviation values can be help full. They are turned on by the options ‘&minmax’ and ‘&stddev’.
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&minmax&stddev

```
Time;5_3;5_3_min;5_3_max;5_3_sdev;1_3;1_3_min;1_3_max;1_3_sdev;2_3;2_3_min;2_3_max;2_3_sdev
2015-01-31 12:00:00;-0.0;-0.1;0.1;0.05;-0.3;-0.4;-0.1;0.10;0.1;-0.0;0.2;0.07
2015-01-31 12:30:00;-0.0;-0.2;0.1;0.06;-0.1;-0.3;0.1;0.10;-0.0;-0.2;0.1;0.06
2015-01-31 13:00:00;-2.0;-5.0;-0.4;1.36;-0.0;-0.1;0.1;0.04;4.0;3.4;4.7;0.34
2015-01-31 13:30:00;-0.1;-0.4;0.1;0.13;-0.2;-0.5;-0.0;0.16;1.4;0.5;3.4;0.88
2015-01-31 14:00:00;-0.1;-0.1;0.1;0.05;-0.1;-0.3;0.1;0.12;0.1;-0.1;0.6;0.18
```

11.4.5 Names in Header (CSV Format)

If you prefer names instead of id’s as column header you just add the option ‘&header_name’:
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&header_name

```
Time;O3;NO;NO2
2015-01-31 12:00:00;-0.0;-0.3;0.1
2015-01-31 12:30:00;-0.0;-0.1;-0.0
2015-01-31 13:00:00;-2.0;-0.0;4.0
2015-01-31 13:30:00;-0.1;-0.2;1.4
2015-01-31 14:00:00;-0.1;-0.1;0.1
```

11.4.6 Resume

In automatic operation it can be very helpful to get a resume on the last transmission. To get it add option ‘&resume’:
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&resume

```
Time;5_3;1_3;2_3
2015-01-31 12:00:00;-0.0;-0.3;0.1
2015-01-31 12:30:00;-0.0;-0.1;-0.0
2015-01-31 13:00:00;-2.0;-0.0;4.0
2015-01-31 13:30:00;-0.1;-0.2;1.4
2015-01-31 14:00:00;-0.1;-0.1;0.1
RESUME
last_timestamp;20150131 14:00:00
datalines;5
skippedlines;0
answertime_sec;0.176
errornr;0
errormsg;OK
```

In csv the resume starts with the keyword RESUME.

On too big queries the data are only transmitted until the maximum number of datasets (Rows are always completed). For this it is very nice to have the ‘last_timestamp’ information. If interpolation is not active you find the number of rows missing in ‘skippedlines’. The other values are self explaining.

This is a resume example in xml:

```
<resume>
  <last_timestamp>20150131 14:00:00</last_timestamp>
  <datalines>5</datalines>
  <skippedlines>0</skippedlines>
  <answertime_sec>0.170</answertime_sec>
  <errornr>0</errornr>
  <errormsg>OK</errormsg>
</resume>
```

11.4.7 XML/CSV

The default value for the data type is csv format but you can change with option ‘&type=xml’.
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&type=xml

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<monitoringdata>
<SiteIdentification>
  <Location>
    <City>Münchendorf</City>
    <AreaCode>2482</AreaCode>
    <Street>Trumauerstrasse 2</Street>
    <State>Austria</State>
  </Location>
  <NameOfStation>200700185</NameOfStation>
  <StationId>Teststation 123</StationId>
  <SerialNumber>2007-00185</SerialNumber>
</SiteIdentification>
<FileSettings>
  <DateStart>2015-01-31</DateStart>
  <TimeStart>12:00:00</TimeStart>
  <DateEnd>2015-01-31</DateEnd>
  <TimeEnd>14:00:00</TimeEnd>
  <DecimalSeparator>POINT</DecimalSeparator>
  <FieldSeparator>;</FieldSeparator>
  <OptionsStatus>0</OptionsStatus>
  <NULLField>-9999</NULLField>
  <MissingIndicator>-9999</MissingIndicator>
</FileSettings>
<AvgSettings>
                    <Avg1Description>average periode 60 sec</Avg1Description>
                    <Avg2Description>average periode 5 sec</Avg2Description>
                    <Avg3Description>average periode 1800 sec</Avg3Description>
                    <Avg10Description>span check values</Avg10Description>
                    <Avg11Description>zero check values</Avg11Description>
</AvgSettings>
<ParameterDetails Position="1">
  <Sensor>O3Sensor</Sensor>
  <Name>O3</Name>
  <Id>5</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</ParameterDetails>
<ParameterDetails Position="2">
  <Sensor>NOxSensor</Sensor>
  <Name>NO</Name>
  <Id>1</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</ParameterDetails>
<ParameterDetails Position="3">
  <Sensor>NOxSensor</Sensor>
  <Name>NO2</Name>
  <Id>2</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</ParameterDetails>
<SData d="2015-01-31" t="12:00:00">-0.0;-0.3;0.1</SData>
<SData d="2015-01-31" t="12:30:00">-0.0;-0.1;-0.0</SData>
<SData d="2015-01-31" t="13:00:00">-2.0;-0.0;4.0</SData>
<SData d="2015-01-31" t="13:30:00">-0.1;-0.2;1.4</SData>
<SData d="2015-01-31" t="14:00:00">-0.1;-0.1;0.1</SData>
</monitoringdata>
```

For a smaller header in xml use option ‘&noheader’
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&type=xml&noheader

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<monitoringdata>
<ParameterDetails Position="1">
  <Sensor>O3Sensor</Sensor>
  <Name>O3</Name>
  <Id>5</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</ParameterDetails>
<ParameterDetails Position="2">
  <Sensor>NOxSensor</Sensor>
  <Name>NO</Name>
  <Id>1</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</ParameterDetails>
<ParameterDetails Position="3">
  <Sensor>NOxSensor</Sensor>
  <Name>NO2</Name>
  <Id>2</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</ParameterDetails>
<SData d="2015-01-31" t="12:00:00">-0.0;-0.3;0.1</SData>
<SData d="2015-01-31" t="12:30:00">-0.0;-0.1;-0.0</SData>
<SData d="2015-01-31" t="13:00:00">-2.0;-0.0;4.0</SData>
<SData d="2015-01-31" t="13:30:00">-0.1;-0.2;1.4</SData>
<SData d="2015-01-31" t="14:00:00">-0.1;-0.1;0.1</SData>
</monitoringdata>
```

With the option ‘&verbose’ you receive a more strict xml response:
http://192.168.10.185/cgi-bin/download.cgi?loginstring=dummy&user_pw=longPW&tstart=2015-01-31,12:00:00&tend=2015-01-31,14:00:00&avg3=5,1,2&type=xml&noheader&verbose

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<monitoringdata>
<responseformat>
<column Pos="1">
  <Sensor>O3Sensor</Sensor>
  <Name>O3</Name>
  <Id>5</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</column>
<column Pos="2">
  <Sensor>NOxSensor</Sensor>
  <Name>NO</Name>
  <Id>1</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</column>
<column Pos="3">
  <Sensor>NOxSensor</Sensor>
  <Name>NO2</Name>
  <Id>2</Id>
  <Avg>3</Avg>
  <Unit>ppb</Unit>
</column>
</responseformat>
```

```
<body>
<row d="2015-01-31" t="12:00:00"><column Pos="1">-0.0</column><column Pos="2">-0.3</column><column
Pos="3">0.1</column></row>
<row d="2015-01-31" t="12:30:00"><column Pos="1">-0.0</column><column Pos="2">-0.1</column><column Pos="3">-
0.0</column></row>
<row d="2015-01-31" t="13:00:00"><column Pos="1">-2.0</column><column Pos="2">-0.0</column><column
Pos="3">4.0</column></row>
<row d="2015-01-31" t="13:30:00"><column Pos="1">-0.1</column><column Pos="2">-0.2</column><column
Pos="3">1.4</column></row>
<row d="2015-01-31" t="14:00:00"><column Pos="1">-0.1</column><column Pos="2">-0.1</column><column
Pos="3">0.1</column></row>
</body>
</monitoringdata>
```

11.5 Synchronous Mode Example:

Invoke download for NO2 (ParamId: 2) and CO (4) of all average values (i.e. 1,2 and 3) for the time period between 1st September 2014, 3p.m. and 5th September 2014 3a.m., using „NULL“ to fill NULL-fields. The domain name of the airpointer is like „airpointer.domain.at“, a registered user's login is „max“, the user's password is „secret“ and this user has at least „Create downloadable data files “ privileges.
Remember: Don't forget to encode the URL string approriately!
-> *Request from application:*
http://airpointer.domain.at/cgi-bin/download.cgi?loginstring=max&user_pw=secret&tstart=2014-09-01,15:00:00&tend=2014-09-05,03:00:00&avg1=2,4&avg2=2,4&avg3=2,4&null=NULL

11.6 Fully Asynchronous Mode Example:

Invoke download for NO2 (ParamId: 2) and CO (4) of average 1 values for the time period between 1st August 2014, 3p.m. and 2nd August 2014 3p.m.
a) Invoke data request:
-> *Request from application:*
http://airpointer.domain.at/cgi-bin/download.cgi?loginstring=max&user_pw=secret&tstart=2014-08-01,15:00:00&tend=2014-08-02,15:00:00&avg1=2,4&type=xml&async=2
-> *Response from instrument:*

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<AirpointerMonitorData>
        <REQUEST>
                <STATUS>WAITING</STATUS>
                <FILE>/download/tmpdata/20150205_135811_2007-00185_200700185_147929.xml.zip</FILE>
        </REQUEST>
</AirpointerMonitorData>
```

b) Check if file is ready for download by consecutively invoking readystate request:
-> *Request from application:*
http://airpointer.domain.at/cgi-bin/download.cgi?loginstring=max&user_pw=secret &readystate=/ download/tmpdata/20150205_135811_2007-00185_200700185_147929.xml.zip
-> *Response from instrument (data file not ready yet):*

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<AirpointerMonitorData>
        <REQUEST>
                <STATUS>WAITING</STATUS>
                <FILE>/download/tmpdata/20150205_135811_2007-00185_200700185_147929.xml.zip</FILE>
        </REQUEST>
</AirpointerMonitorData>
```

-> *Response from instrument (data file ready):*

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<AirpointerMonitorData>
        <REQUEST>
                <STATUS>OK</STATUS>
                <FILE>/download/tmpdata/20150205_135811_2007-00185_200700185_147929.xml.zip</FILE>
        </REQUEST>
</AirpointerMonitorData>
```

c) After receiving `<STATUS>OK</STATUS>` download the created data file (use the file name from the xml response):

-> *Request from application:*
*http://airpointer.domain.at/*download/tmpdata/20150205_135811_2007-00185_200700185_147929.xml.zip

d)After the file has been downloaded it should be removed to keep the unit clean:
-> *Request from application:*
http://airpointer.domain.at/cgi-bin/download.cgi?loginstring=max&user_pw=secret &remove=/ download/tmpdata /20150205_135811_2007-00185_200700185_147929.xml.zip

12 Error Codes:

0 is OK, 1-99 Warnings, >= 100 Failures

0	ERR_OK:	OK
1	ERR_NO_DATA_FOR_REQUEST:	no data for that request
2	ERR_TOO_MANY_DATA:	Too many datasets defined
3	ERR_PARAM_UNDEF:	undefined parameter(s)
4	ERR_Z/S_CHECK_PENDING:	a calibration is already in progress
5	ERR_NO_ZERO_VALVE_PRESENT:	there is no zero valve on the grp
6	ERR_NO_SPAN_VALVE_PRESENT:	there is no span valve on the grp
100	ERR_FAIL:	failure
101	ERR_UNDEF:	undefined error
102	ERR_NAME_MISS:	name missing
103	ERR_WRONG_NAME:	wrong name");
104	ERR_UNKNOWN_ID:	cannot find definition for id:
105	ERR_NO_TABLE:	no table
106	ERR_READ_TABLE:	error reading table
107	ERR_UPDATE_FAIL:	update failed
108	ERR_QUERY_FAIL:	query failed
109	ERR_CANNOT_OPEN_FILE:	cannot open file
110	ERR_WRONG_VERSION:	wrong version
111	ERR_WRONG_TIME_DEFINITION:	Cannot find correct time definition
112	ERR_NO_PARAM_DEF:	No parameter defined
113	ERR_TOO_MANY_PARAM_DEF:	Too many parameters defined!
114	ERR_DURING_COMPRESSION:	Error during file compression!
115	ERR_WRONG_FORMAT:	wrong format
116	ERR_SQL_PROBLEM:	SQL problem
117	ERR_AUTHENT_FAILED:	Authentication failure
118	ERR_NO_COMMAND:	No command found
119	ERR_WRONG_SEPARATOR:	wrong separator
120	ERR_WRONG_DECSEP:	wrong decimal separator
121	ERR_TOO_MANY_REQUESTS:	too many requests pending
122	ERR_WRONG_PATH:	wrong path in filename
123	ERR_FILENAME_MISS:	missing file name
124	ERR_WRONG_COMPRESSION:	wrong compression parameter

ChangeLog:

2.01 : correcting misspelled station identification -> station identification
Corrected string length of time zone string
Added protocol version to full station info
2.02 : Option &conf in info makes an automatic configuration of a central software easier.
2.03 : new command Gasinfo
2.04 : Function check controlled by http interface