FANselect API

Developer‘s Manual

Edition 0

ZIEHL-ABEGG

0. Table of Contents

1. Introduction

The FANselect API serves as an Application Programming Interface to FANselect. It requires a request string as input and outputs a response string.

Both request and response strings can be formatted as JSON or XML. It is up to the calling application to create the required input and parse the API's output.

This API can be:
A. Downloaded (as a Windows DLL) by clicking on the link below:
http://www.ziehl-abegg.com/fileadmin/de/de/05_Support/Software/FANselect/FANselect_DLL_daily.zip

B. Accessed through the web via: http://fanselect.net:8079/FSWebService

The downloadable FANselect DLL folder can be placed anywhere on your machine. It is important to keep the folder intact. Your application would need to access the fanselect.dll file inside this folder.

To update your version of the DLL:
1. Download the new DLL folder from the URL above
2. Delete your actual DLL folder
3. Place the new DLL folder in that location vacated by your previous DLL folder

In every DLL folder there is a test tool, called **ZADllTest.exe** or **ZADllTest64.exe**, with which you can test input and output strings.

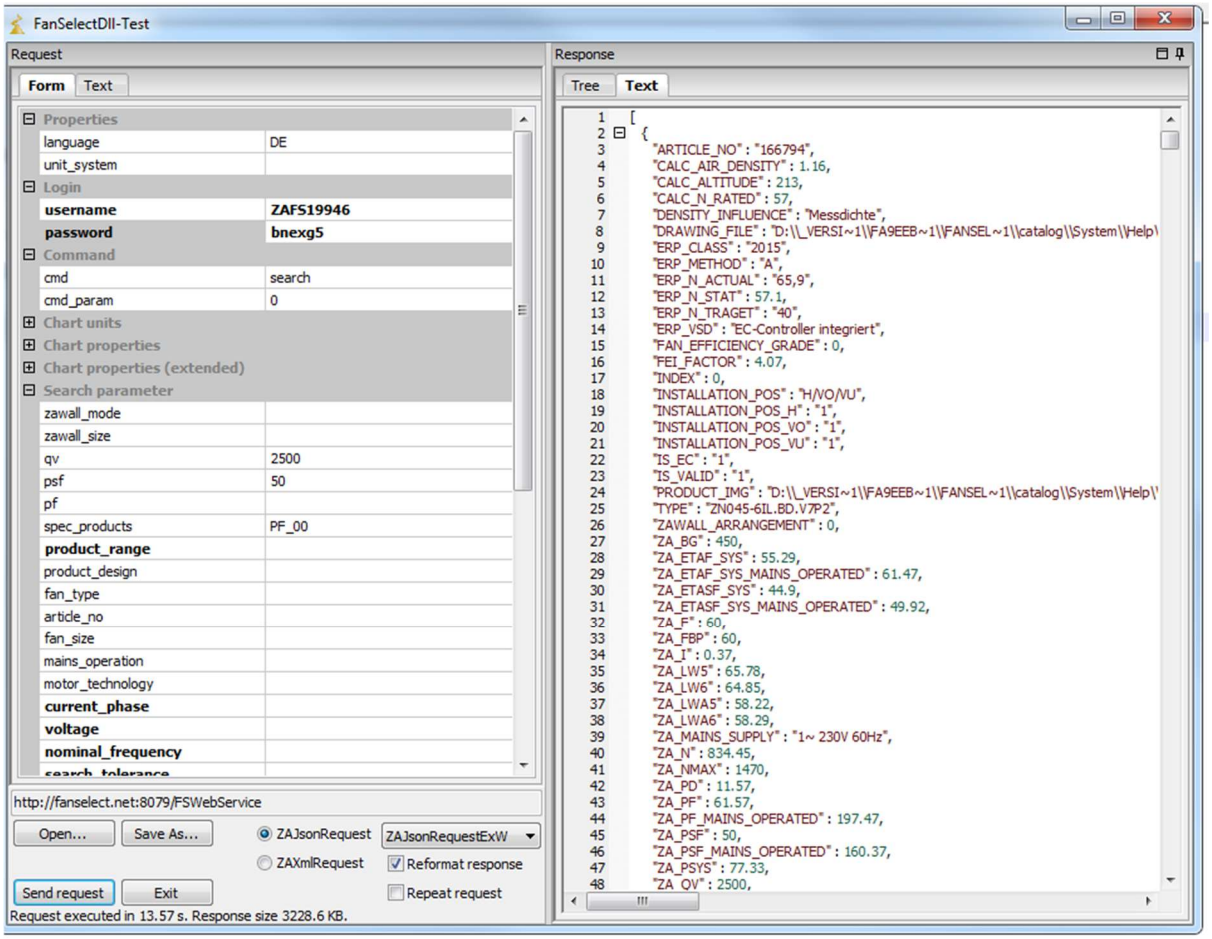

image 0: Left is the input area whereas the right side contains the outputs produced by the DLL. Click on the tab Text on the top left side to see the request string generated.

The FANselect web API is always update and hence does not require the user to update.

2. Connect to FANselect DLL

2.0 Minimal Required Inputs:

username: Your FANselect account's username
password: Your FANselect account's password
cmd: search (explained in section 2.1 below)
qv: Duty point's volumetric flow rate
psf: Duty point's static pressure
spec_products: Portfolio containing required fans (explained in section 3.0 below)
language: Choose language for outputs to appear in (explained in section 3.0 below)

With these minimum inputs, your request string should look like the samples below:

JSON Request String example

```
{
    "username" : "ZAFS19946"
    "password" : "bnexg5",
    "cmd" : "search",
    "qv" : "2500",
    "psf" : "50",
    "spec_products" : "PF_00",
    "language": "EN",
}
```

Identical Request String as XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<request>
        <username> ZAFS19946</username>
        <password> bnexg5</password>
        <cmd>search</cmd>
        <qv>2500</qv>
        <psf>50</psf>
        <spec_products>PF_00</spec_products>
        <language>EN</language>
</request>
```

2.1 Programming a DLL Reader

You can access the DLL via one of three functions.

ZAJsonRequestW: For Unicode Strings
ZAJsonRequestA: For UTF-8 Strings
ZAJsonRequestBSTR: For OLE objects

Your DLL reader has to pass the request string as an argument to one of the functions above, and then read the DLL's output.

DLL Reader function in Python

```
def za_dll_fan_selection(request_string, dll_path):
        import ctypes
        import json

        fanselect_dll = ctypes.WinDLL(dll_path)
        fanselect_dll_output = (ctypes.wstring_at(fanselect_dll.ZAJsonRequestW(request_string)))

        return fanselect_dll_output
```

request_string is identical format to Request String example above, albeit with more inputs
dll_path: is the path to the FANselect DLL, ie C.\FANselect_DLL\FANselect_DLL}fanselect.dll

DLL Reader function in VBA

```
Private Declare Function ZAJsonRequestBSTR Lib "C:\FANselect_DLL\FANselect_DLL\FANselect.dll"
(ByVal sRequest As String) As String

Public Function vba_reader(ByVal input_request_string As String) As String

    Dim request_string As String
    Dim response_string As String

    Dim request_string_unicode As Variant
    Dim response_string_unicode As Variant

    request_string = "{" + input_request_string + "}"

    request_string_unicode = StrConv(request_string, vbUnicode)
    response_string_unicode = ZAJsonRequestBSTR(request_string_unicode)

    response_string = StrConv(response_string_unicode, vbFromUnicode)

    vba_reader = response_string

End Function
```

Further Examples can be downloaded from the links below

C++	http://downloads.fanselect.net//fanselect/dll_examples/CPPConsoleApp.zip
C#	http://downloads.fanselect.net//fanselect/dll_examples/VCS10StandardApp.zip
Delphi	http://downloads.fanselect.net//fanselect/dll_examples/DelphiConsoleApp.zip
VB6	http://downloads.fanselect.net//fanselect/dll_examples/VB6StandardApp.zip
VB10	http://downloads.fanselect.net//fanselect/dll_examples/VB10StandardApp.zip

2.2 Connect to FANselect Web API

Accessing FANselect's web API is nearly identical to the process used to access the DLL.
The only difference is that you have to send two requests:

1st Request: To obtain a Session ID
2nd Request: Usual request, which includes the session ID obtained in the first request

The major advantage of the web API is that it is (as mentioned before) always up to date and does not require to be downloaded. Please examine Internet reliability in your location and your machine's firewall / security settings, as these could hamper the web API's performance-

As with the downloadable DLL, requests to and responses from the web API can be sent as JSON or XML strings.

Both DLL and web API produce identical outputs, as both use the same selection and calculation algorithms. Any discrepancies between DLL and web API, are probably due to an outdated DLL.

Web API Reader function in Python

```
import json
dll_path = "http://fanselect.net:8079/FSWebService"

def za_api_fan_selection_0(request_string, dll_path):
    import requests
    fanselect_api_output = requests.post(url=dll_path, data=request_string)
    return fanselect_api_output
```

```
# Get Session ID
request_string = "{'cmd' : 'create_session',  'username' : 'USERNAME', 'password' : 'PASSWORD' }"

request_string = str(request_string)
dll_path = str(dll_path)

response_string  = za_api_fan_selection_0(request_string, dll_path)
session_id = json.loads(response_string_raw.content)['SESSIONID']

# Usual Request
request_string = "{"
request_string = request_string + "'username' : 'USERNAME',"
request_string = request_string + "'password' : 'PASSWORD',"
request_string = request_string + "'language' : 'EN',"
request_string = request_string + "'unit_system' : 'm',"
request_string = request_string + "'cmd' : 'search',"
request_string = request_string + "'cmd_param' : '0',"
request_string = request_string + "'spec_products' : 'PF_00',"
request_string = request_string + "'product_range' : 'BR_01',"
request_string = request_string + "'qv' : '2500',"
request_string = request_string + "'psf' : '50',"
request_string = request_string + "'current_phase' : '3',"
request_string = request_string + "'voltage' : '400',"
request_string = request_string + "'nominal_frequency' : '50',"
request_string = request_string + "'sessionid' : '" + session_id + "',"
request_string = request_string + "'full_octave_band' : 'true',"
request_string = request_string + "}"

request_string = str(request_string)
response_string_initial = za_api_fan_selection_0(request_string, dll_path)
```

Further Examples can be downloaded from the links below

C# http://downloads.fanselect.net/fanselect/dll_examples/VCS10WebService.zip

VB10 http://downloads.fanselect.net//fanselect/dll_examples/VB10WebService.zip

3. Inputs & Outputs

3.0 All Inputs Explained

language

Set language of outputs

input options:

CS: Czech	**DA**: Danish	**DE**: German	**EN**: English
ES: Spanish	**FR**: French	**FI**: Finnish	**HU**: Hungarian
IT: Italian	**JA**: Japanese	**NL**: Dutch	**PL**: Polish
PT: Portuguese	**RU**: Russian	**SV**: Swedish	**TR**: Turkish
ZH: Chinese			

unit_system

unit system to be used in calculations.

Input options: **m**: metric **i**: imperial

username

Your FANselect account's username

password

Your FANselect account's password

Users who are only interested in a limited set of articles, can acquire one or more username / password combinations (logins). Each login would offer a specific set of articles - predefined by the user.

The user's application would then call the dll with one of these specific logins to select from a limited pool of articles. Advantages: Faster selection process and smaller number of articles among found set

cmd

cmd, short for command, is needed to instruct the DLL on the type of outputs required

Input Options:

search	selection by duty point + filters such as size, design etc
status	Delivers **username** and software version. Web API also outputs **SESSIONID**.
create_session	Obtain **SESSIONID**. This **cmd** is only relevant for the web API

The following **cmd**'s require an article number in **article_no**

select	Select by article number. Article's nominal data is output if duty point is not achieved
nominal_values	Obtain article's electric nominal values. This data also be obtained with your initial **search** request by setting **insert_nominal_values** to **true**.
motor_data	Article motor data. Can also be obtained with **search** and **insert_motor_data**: true
geo_data	Article dimensions. Get this data with **search** by setting **insert_geo_data** to **true**.
accessories	Depict accessories associated with article
get_chart	Create charts for selected article

cmd_param
You can set the index of the article you wish

zawall_mode
Pick whether you want to select multiple fans, with either one of two options
ZAWALL: Select using multiple fans only
ZAWALL_PLUS: Select using multiple and single fans

zawall_size
Set the number of fans you want to use in your multiple fans array. Maximum number of fans is set to 20. **zawall_size** can also be left empty. FANselect will automatically determine the number of fans required. Selections without a pre-set number of fans usually come with a longer response time.

qv
Volumetric rate either in m³/h for **unit_system** choice **m** or CFM for **unit_system** choice **i.**

psf
Static pressure either in Pa for **unit_system** choice **m** or in wg for **unit_system** choice **i.**

pf
Total pressure either in Pa for **unit_system** choice **m** or in wg for **unit_system** choice **i**
In your request string, you either specify **psf** or **pf**.

spec_products
Fans in FANselect are placed in discrete portfolios, denoted by the PF codes listed below. It is mandatory to input a specific portfolio. Currently it is not possible to select across multiple portfolios.

Input Options
PF_00: Standard Worldwide
PF_01: USA Standard Products
PF_02: Brasil Portfolio
PF_03: AMCA USA Products
PF_04: AMCA Thailand Products
PF_06: India Portfolio
PF_07: AMCA Germany Products

product_range
Fans are placed in clusters called product ranges and denoted by the BR codes listed below. It is not mandatory to input a **product_range**. It is also possible to use multiple BR codes per selection. These BR codes would have to be separated by |, i.e. BR_01 | BR_57 | BR_59

Input Options
BR_01 EC fans
BR_04 Vpro ECblue fans
BR_07 FEowlet AC fans
BR_08 Cpro AC fans
BR_09 C AC fans
BR_10 Vpro AC fans
BR_12 FE2owlet ECblue fans
BR_13 FE2owlet ECblue fans
BR_15 C ECblue fans
BR_17 Cpro fan series
BR_18 C fan series
BR_19 Vpro fan series
BR_21 Fans for air handling units
BR_28 Fans for transformer cooling
BR_40 Cpro ECblue fans
BR_41 Fans for agriculture applications
BR_44 MAXvent Owlet series
BR_56 Fans for clean room applications
BR_57 Cpro PMblue fans
BR_58 C PMblue fans
BR_59 Cpro AMblue fans
BR_60 C AMblue fans
BR_60 ZAvblue ECblue fans
BR_61 ZAvblue series
BR_82 ZAbluefin fans with PMblue motors
BR_84 ZApilot Series
BR_85 ZAbluefin fans with IEC motors
BR_87 ZAcube Series
BR_105 ZAventero Series
BR_106 ZAbluefin fans with ECblue motors

product_design
Every article can come in one of a multitude of designs. Leave empty if design is not known

Input Options

Axial flow fans with airflow direction A: Air is sucked over motor

A-A: Axial fan consisting only of impeller
A-D: Axial fan sucking through grille
A-F: Tube axial fan with longer tube, round housing
A-L: Tube axial fan with shorter tube, round housing
A-Q: Tube axial fan with shorter tube, rectangular housing
A-W: Axial fan sucking through grille

Axial flow fans with airflow direction V: Air is blown over motor

V-A: Axial fan consisting only of impeller
V-E: Tube axial fan with shorter tube and sucking through contact protection
V-F: Tube axial fan with very long tube
V-H: Tube axial fan with shorter tube, round housing
V-H: Wall-mounted Tube axial fan with shorter tube and guide-vanes
V-L: Wall-mounted Tube axial fan with shorter tube and guide-vanes
V-Q: Wall-mounted Tube axial fan with shorter tube and guide-vanes
V-I: Axial fan blowing through grille
V-K: Axial fan blowing through grille
V-L: Tube axial fan with shorter tube, round housing
V-Q: Tube axial fan with shorter tube, rectangular housing
V-S: Axial fan blowing through grille, surrounding whole backside of fan

Centrifugal Fans

ER: Centrifugal plug fan design
GR-H: Wall mounted centrifugal fan design, horizontally mounted
GR-Vo: Wall mounted centrifugal fan design, vertically mounted facing upward
GR-Vu: Wall mounted centrifugal fan design, vertically mounted facing downward
GR: Wall mounted centrifugal fan design
RH: Centrifugal fan consisting only of impeller
WR: Centrifugal fan placed in cube design

fan_type
Filter by defining part of fan's type key. Wild cards are: * for multiple characters and ? for 1 character.
i.e.: GR56C*1C to get all size 560 C impellers in GR design, ER??I-4* to get all ZAbluefin in ER design

article_no
Article number (if known) of the required fan.
Multiple article numbers can be input at once, each separated by a |, such as: 178125 | 178153 | 178113.

fan_size
fan size of required fans (if known)

mains_operation
Choose whether required fan should be connected to a controller or not.

Input Options:

NETZ: Fan directly connected to electric grid
FZ: Fan connected to frequency converter

motor_technology
Select the type of motor best suited for your application. Multiple choices can be input separated by a |
i.e.: ZAmotpremium IE2 | PMblue IE4 | ZAmotpremium IE3

Input Options;

AC ERM:	External rotor AC motor
AMblue IE3:	Internal rotor IE3 motor with controller
ECblue:	External rotor EC motors
ECQ:	External rotor EC motor
PMblue IE4:	Permanent magnet IE4 internal rotor motor
PMblue Standalone:	Permanent magnet IE4 internal rotor motor without controller
ZAmotbasic EX:	Low cost internal rotor ATEX motor
ZAmotbasic IE2:	Low cost internal rotor IE2 motor
ZAmotbasic IE3:	Low Cost iInternal rotor IE3 motor
ZAmotpremium IE2:	Premium internal rotor IE2 motor
ZAmotpremium IE3:	Premium internal rotor IE3 motor
ZAmotpremium PE:	Premium internal rotor Premium Efficiency (USA) motor

current_phase
Electric current phases. **Input Options**: **1** or **3.**

voltage
Electric voltage **Input Options: 230 400 460 690**

nominal_frequency
Electric nominal frequency. **Input Options: 50 60**

search_tolerance
Required selection tolerance

motor_safety_margin
Motor power reserve, if required
ie **motor_safety_margin** = 10 => 10 kW shaft power requires 11 kW motor

airflow_volume_reserve
Airflow volumetric reserve, if required
ie **airflow_volum_reseve** = 10 => 1000 m³/h required flow means fan must deliver 1100 m³/h

air_density
Fan operating air density. Fan selection and duty point calculations will adjust to the density.

ambient_temperature
Medium Temperature at which fan is operating

grill_influence
Only applicable to centrifugal fans
Input Options:
false: no grill taken into account
true: duty point calculations affecting fan performance and acoustics take grill into account

installation_height_mm
Height of enclosure in mm
Placing fans within enclosures requires the overall dimensions of these enclosures, as they negatively affect fan performance. The smaller the enclosure is relative to fan size the more detrimental it is to fan performance

installation_width_mm
Width of enclosure in mm.

installation_length_mm
Length of enclosure in mm.

installation_mode
Enclosure performance losses are calculated by specific algorithms. FANselect offers multiple loss calculating algorithms for single fans, yet only one (**RLT_2017**) for multiple fan layouts

Input Options:
ZA: Inhouse developed algorithm
RLT_2017: Most recent Algorithm developed by the AHU Manufacturer's Association

protection_class
Input required protection class (if known) as an IPxx number.

erp_class
Input ERP (Energy Related Products-Directive) class ie 2015.
The ErP class defines the minimum efficiency a fan can have to be sold in certain markets

sfp_class
Input SFP (Specific Fan Performance) Class as a digit, ie 3, 4. SFP is basically the input electric power in relation to the output airflow.

full_octave_band
To display the full octave band with **cmd: search**, set this parameter to **true**.

insert_nominal_values
Set this parameter to **true** to show all the electrical nominal values with **cmd: search**.

insert_motor_data
Set this parameter to **true** to show relevant moor data with **cmd: search**.

insert_geo_data
Set this parameter to **true** to depict article's dimensions

pricelist_name
By entering the name of the Excel sheet found in the DLL folder: **Product_Price_Reference..xls**, you can have the price appear among the DLL's outputs. Excel file has one spreadsheet with three columns

Column 1: Customer article number. Here any number system can be used.
Column 2: Ziehl-Abegg article number, which is used for the selection calculations
Column 3: Price of this article

3.1 All Outputs Explained

ARTICLE_NO	Article number
CALC_AIR_DENSITY	Air Density used in selection and calculation (kg/m³)
CALC_ALTITUDE	Altitude used in selection and calculation (m above sea level)
CALC_LW5_OKT	Suction side octave band, values separated by commas (dB)
CALC_LW6_OKT	Pressure side octave band, values separated by commas (dB)
CALC_LWA5_OKT	Suction side weighted octave band values (dBA)
CALC_LWA6_OKT	Pressure side weighted octave band values (dBA)
CALC_NOZZLE_PRESSURE	Pressure in nozzle, used to determine air flow (Pa)
CALC_N_RATED	Ratio of duty point fan rpm to maximum fan rpm (%)

CALC_P1_MAX	Maximum absorbed electrical power at duty point (W)
CALC_PL_MAX	Maximum absorbed shaft power at duty point (W)
CALC_PSYS_MAX	Maximum absorbed system power = motor + controller absorbed power (W)
CALC_TEMP_C	Medium temperature (°C)
CAPACITOR_CAPACITANCE	Capacitor capacitance (μF)
CAPACITOR_VOLTAGE	Capacitor voltage (V)
CHART_VIEWER_URL	URL to chart depicting fan curves
CIRCUIT	Type of electrical circuit
COSPHI	Fan motor Cosine Phi value
CURRENT_PHASE	Fan motor phases
DENSITY_INFLUENCE	Density used in determining duty point **measurement density** => Selection at fan's measured density **density** => Selection at density different from measured density
DRAWING_FILE	Path to fan drawing
EC_TYPE	Output is **1** if fan is powered by an EC motor and an empty string if fan motor is not an EC motor
EFFICIENCY_CLASS	Efficiency Class of IEC motor. Parameters only shows up alongside fans powered by IEC motors
EFFICIENCY_STAT	Static efficiency of fan = Volumetric Rate X Static Pressure / Power Absorbed by System (%)
EFFICIENCY_TOT	Total efficiency of fan = Volumetric Rate X Static Pressure / Power Absorbed by System (%)
ERP_CLASS	Fan ERP class
ERP_METHOD	Method used to measure ERP class
ERP_N_ACTUAL	ERP efficiency factor = Best efficiency possible
ERP_N_STAT	Actual ERP Efficiency at duty point (%)
ERP_N_TRAGET	ERP required efficiency factor
ERP_VSD	Returns **EC controller integrated** if fan is so equipped. and an empty string for fans without an integrated speed control system
FAN_EFFICIENCY_GRADE	This is a factor assigned to individual fans and is only relevant for AMCA fans
FEI_FACTOR	This factor is calculated based on the duty point and is only relevant for AMCA fans
GRILL_INFLUENCE	Returns **no** if grill influence is not factored into calculations, and **yes** if grill's influence is taken into account.
INCREASE_OF_CURRENT	Current increase (%)

INDEX	Sequence number of fan in found set. First fan in found set would have index **0**, second fan index **1** etc.
INSTALLATION_HEIGHT_MM	Height of Fan (mm)
INSTALLATION_LENGTH_MM	Length of fan (mm)
INSTALLATION_POS	Returns fan orientation(s): **H**: Horizontal **VO**: Vertical facing up **VU**: Vertical facing down
INSTALLATION_POS_H	Returns **1** for horizontally oriented fans (**INSTALLATION_POS** = **H)**, and an empty string for remaining fans.
INSTALLATION_POS_VO	Returns **1** for vertical upwards facing fans (**INSTALLATION_POS** = **VO)** and an empty string for remaining fans
INSTALLATION_POS_VU	Returns **1** for vertical downwards facing fans (**INSTALLATION_POS** = **VU)** and an empty string for remaining fans
INSTALLATION_WIDTH_MM	Width of fan (mm)
IS_EC	Returns **1** if fan has EC motor and empty string for non-EC motors
KFACTOR	Fan's nozzle pressure
MAX_CURRENT	Fan's maximum current (A)
MAX_FREQUENCY	Fan's maximum frequency (Hz)
MAX_TEMPERATURE_C	Fan's maximum temperature (°C)
MAX_VOLTAGE	Fan's maximum voltage (V)
MDRAWING	Name of drawing file
MIN_CURRENT	Fan's minimum current (A)
MIN_TEMPERATURE_C	Fan's minimum temperature (°C)
MIN_VOLTAGE	Fan's maximum voltage (V)
MOTOR_DESIGN	Type of motor design: (only for IEC motors) **IMB 3**: Foot mounted **IMB 5**: Flange mounted
MOTOR_POLES	Number of motor poles (for IEC powered fans)
MOTOR_SHAFT	IEC motor shaft description: number / diameter X length
MOTOR_SIZE	IEC motor size
NOMINAL_CURRENT	Fan motor nominal current (A)
NOMINAL_FREQUENCY	Fan motor nominal frequency (Hz)
NOMINAL_IECMOTOR_EFFICIENCY	IEC Motor nominal efficiency as a decimal number
NOMINAL_SPEED	Fan's nominal speed (1/min)
NOMINAL_VOLTAGE	Fan motor nominal voltage

NOZZLE_GUARD	Information on how fan was measured. Predominantly for axial fans
NUMBER_OF_POLES	IEC motor number of poles
PHASE_DIFFERENCE	Phase difference
POWER_INPUT_KW	Power required by motor (kW)
POWER_INPUT_KW	Power Output by motor (kW)
PRODUCT_IMG	Path to product image
PROTECTION_CLASS_IP	Protection class as IP number
PROTECTION_CLASS_THCL	Temperature protection class as THCL number
RUBBER_MOT_DIAMETER	Motor rubber damper diameter
RUBBER_MOT_HEIGHT	Motor rubber damper height
SPRING_MOT_DIAMETER	Motor spring damper diameter
SPRING_MOT_HEIGHT	Motor spring damper height
TYPE	Type key of fan
VOLTAGE_TOLERANCE	Voltage tolerance (%)
ZAWALL_ARRANGEMENT	Multiple fan layout. Returns 0 if no multiple fans are selected
ZA_BG	Fan nominal size
ZA_COSPHI	Fan motor Cos Phi
ZA_ETAF	Total efficiency of fan = Volumetric Rate X Total Pressure / Power Absorbed by System (%)
ZA_ETAF_L	Fan impeller total efficiency (%)
ZA_ETAF_SYS	System total efficiency (%)
ZA_ETAM	Motor efficiency (%)
ZA_ETASF	Static efficiency of fan = Volumetric Rate X Static Pressure / Power Absorbed by System (%)
ZA_ETASF_L	Fan impeller static efficiency (%)
ZA_ETASF_SYS	System static efficiency (%)
ZA_F	Fan nominal electrical frequency (Hz)
ZA_FBP	Fan electrical frequency at duty point (Hz)
ZA_I	Fan current at duty point (A)
ZA_IN	Fan nominal current (A)
ZA_LW5	Duty point acoustic power level suction side (dB)
ZA_LW6	Duty point acoustic power level pressure side (dB)
ZA_LWA5	Duty point weighted acoustic power level suction side (dBA)

ZA_LWA6	Duty point weighted acoustic power level pressure side (dBA)
ZA_MAINS_SUPPLY	Mains supply: phases, voltage and electric frequency
ZA_N	RPM at duty point (1/min)
ZA_NMAX	Maximum RPM of fan (1/min)
ZA_PD	Dynamic pressure at duty point (Pa)
ZA_PF	Total pressure of fan. ZA_PF = ZA_PSF + ZA_PD (Pa)
ZA_PF_MAINS_OPERATED	Total Pressure of fan in mains operation (Pa)
ZA_PSF	Static pressure of fan (Pa)
ZA_PSF_MAINS_OPERATED	Static pressure of fan in mains operation (Pa)
ZA_P1	Electrical power required at duty point (W)
ZA_PD	Duty point dynamic pressure (Pa)
ZA_PF	Duty point total pressure (Pa)
ZA_PL	Calculated shaft power at duty point (W)
ZA_PSF	Duty point static pressure (Pa)
ZA_PSYS	Absorbed power by system (W)
ZA_QV	Duty point volumetric flow rate (m³/h)
ZA_QV_MAINS_OPERATED	Duty point volumetric flow rate in mains operation (m³/h)
ZA_SFP	SFP number of fan
ZA_SFP_CLASS	Fan's SFP class
ZA_U	Fan voltage at duty point (V)
ZA_UN	Fan nominal voltage (V)
ZA_WEIGHT	Mass of fan

3.2 Outputs of each cmd

3.2.0 cmd: search Outputs

ARTICLE_NO	CALC_AIR_DENSITY	CALC_ALTITUDE
CALC_NOZZLE_PRESSURE	CALC_N_RATED	DENSITY_INFLUENCE
DRAWING_FILE	ERP_CLASS	ERP_METHOD
ERP_N_ACTUAL	ERP_N_STAT	ERP_N_TRAGET
ERP_VSD	FAN_EFFICIENCY_GRADE	FEI_FACTOR
GRILL_INFLUENCE	INDEX	INSTALLATION_HEIGHT_MM
INSTALLATION_LENGTH_MM	INSTALLATION_POS	INSTALLATION_POS_H

INSTALLATION_POS_VO	INSTALLATION_POS_VU	INSTALLATION_WIDTH_MM
IS_EC	IS_VALID	KFACTOR
NOZZLE_GUARD	PRODUCT_IMG	TYPE
ZAWALL_ARRANGEMENT	ZA_BG	ZA_COSPHI
ZA_ETAF_SYS	ZA_ETAF_SYS_ MAINS_OPERATED	ZA_F
ZA_FBP	ZA_I	ZA_LW5
ZA_LW6	ZA_LWA5	ZA_LWA6
ZA_MAINS_SUPPLY	ZA_N	ZA_NMAX
ZA_PD	ZA_PF	ZA_PF_MAINS_OPERATED
ZA_PSF	ZA_PSF_MAINS_OPERATED	ZA_PSYS
ZA_QV	ZA_QV_MAINS_OPERATED	ZA_SFP
ZA_SFP_CLASS	ZA_U	ZA_UN
ZA_WEIGHT		

3.2.1 cmd: select Outputs

This **cmd** requires that you input an article number in **article_no**.

ARTICLE_NO	CALC_AIR_DENSITY	CALC_ALTITUDE
CALC_LW5_OKT	CALC_LW6_OKT	CALC_LWA5_OKT
CALC_LWA6_OKT	CALC_NOZZLE_PRESSURE	CALC_N_RATED
CAPACITOR_CAPACITANCE	CAPACITOR_VOLTAGE	CHART_VIEWER_URL
CIRCUIT	COSPHI	CURRENT_PHASE
DENSITY_INFLUENCE	DRAWING_FILE	EC_TYPE
EFFICIENCY_STAT	EFFICIENCY_TOT	ERP_CLASS
ERP_METHOD	ERP_N_ACTUAL	ERP_N_STAT
ERP_N_TRAGET	ERP_VSD	FAN_EFFICIENCY_GRADE
FEI_FACTOR	GRILL_INFLUENCE	INCREASE_OF_CURRENT
INSTALLATION_HEIGHT_MM	INSTALLATION_LENGTH_MM	INSTALLATION_POS
INSTALLATION_POS_H	INSTALLATION_POS_VO	INSTALLATION_POS_VU
INSTALLATION_WIDTH_MM	IS_EC	IS_VALID
KFACTOR	MAX_CURRENT	MAX_TEMPERATURE_C
MAX_VOLTAGE	MIN_CURRENT	MIN_TEMPERATURE_C

MIN_VOLTAGE	NOMINAL_FREQUENCY	NOMINAL_SPEED
NOMINAL_VOLTAGE	NOZZLE_GUARD	PHASE_DIFFERENCE
POWER_INPUT_KW	PRODUCT_IMG	PROTECTION_CLASS_IP
PROTECTION_CLASS_THCL	TYPE	VOLTAGE_TOLERANCE
ZAWALL_ARRANGEMENT	ZA_BG	ZA_COSPHI
ZA_ETAF_SYS	ZA_ETAF_SYS_ MAINS_OPERATED	ZA_ETASF_SYS
ZA_ETASF_SYS_ MAINS_OPERATED	ZA_F	ZA_FBP
ZA_I	ZA_LW5	ZA_LW6
ZA_LWA5	ZA_LWA6	ZA_MAINS_SUPPLY
ZA_N	ZA_NMAX	ZA_PD
ZA_PF	ZA_PF_MAINS_OPERATED	ZA_PSF
ZA_PSF_MAINS_OPERATED	ZA_PSYS	ZA_QV
ZA_QV_MAINS_OPERATED	ZA_SFP	ZA_SFP_CLASS
ZA_U	ZA_UN	ZA_WEIGHT

3.2.2 cmd: nominal_values Outputs

This **cmd** requires that you input an article number in **article_no**.
The outputs below can also be output using **cmd search** by setting **insert_nominal_values** to **true**

ARTICLE_NO	CAPACITOR_CAPACITANCE	CAPACITOR_VOLTAGE
CIRCUIT	COSPHI	CURRENT_PHASE
EC_TYPE	EFFICIENCY_STAT	EFFICIENCY_TOT
INCREASE_OF_CURRENT	MAX_CURRENT	MAX_FREQUENCY
MAX_SPEED	MAX_TEMPERATURE_C	MAX_VOLTAGE
MIN_CURRENT	MIN_PSF	MIN_TEMPERATURE_C
MIN_VOLTAGE	NOMINAL_CURRENT	NOMINAL_FREQUENCY
NOMINAL_SPEED	NOMINAL_VOLTAGE	PHASE_DIFFERENCE
POWER_INPUT_HP	POWER_INPUT_KW	POWER_OUTPUT_HP
POWER_OUTPUT_KW	PROTECTION_CLASS_IP	PROTECTION_CLASS_THCL
VOLTAGE_TOLERANCE		

3.2.3 cmd: get_chart Outputs

This **cmd** requires that you input an article number in **article_no**.
get_chart produces the outputs below plus the selected fan's curves

BOTTOM_MARGIN	CHART_FILE	CHART_MAX_X
CHART_MAX_Y	CHART_MIN_X	CHART_MIN_Y
LEFT_MARGIN	MEASUREMENT_ID	RIGHT_MARGIN
TOP_MARGIN		

3.2.4 cmd: motor_data Outputs

For EC Motors

CIRCUIT	NOMINAL_VOLTAGE	PROTECTION_CLASS_IP

FOR IEC Motors

CIRCUIT	EFFICIENCY_CLASS	MOTOR_DESIGN
MOTOR_SHAFT	MOTOR_SIZE	NOMINAL_CURRENT
NOMINAL_VOLTAGE	NUMBER_OF_POLES	POWER_OUTPUT_KW
PROTECTION_CLASS_IP	RUBBER_MOT_DIAMETER	RUBBER_MOT_HEIGHT
SPRING_MOT_DIAMETER	SPRING_MOT_HEIGHT	

3.2.5 cmd: status Outputs

This **cmd** is useful to obtain the DLL's version and the user's username

USERNAME	VERSION	

3.2.6 cmd: create_session

This **cmd** is used to create a session, before calling the web DLL
.

USERNAME	VERSION	

4. Help and Support

4.0 Contact Information

Please feel free to contact us, should you need further help or council on how to integrate the FANselect API into your application.

Contact Person:
Sargon Baki
Email: sargon.baki@ziehl-abegg.de
Tel: 0049 7940 16 257

4.1 Links

FANselect DLL Download
http://www.ziehl-abegg.com/fileadmin/de/de/05_Support/Software/FANselect/FANselect_DLL_daily.zip

FANselect Web API
http://fanselect.net:8079/FSWebService

Article Images and Drawings
https://www.ziehl-abegg.com/fileadmin/de/de/05_Support/Software/FANselect/catalog.zip

5. About this Manual

Author	Sargon Baki
Department	V-SPT – Product Management / Tools & Training
Edition	0
Publication Date	11.04.2019