Testo Data Interface Flue Gas Analyzers

1 Table of Contents

Document Revision History

Version	Date	Description
1.0	2015-02-27	Initial version
1.1	2019-07-05	adding testo 300

2 Introduction

This documentation describes an interface for the transfer of analyzer, measure site and measurement data between your own PC Software Application and Testo flue gas measure analyzers.

To transfer data between such business program and the analyzer the Data Interface Flue Gas Analyzers was defined. The interface definition consists of a XML schema file (.xsd) and this text document. If there are any doubts or ambiguities, the XML schema file is the preferred document.

3 Technical description

3.1 File based data transfer

The Testo Data Interface Flue Gas Analyzers is realized on a specified file. This file could be read from your own PC Software Application. If you want to transfer data to a Testo analyzer you can create such a file with your own software.

In this scenarios, your own PC Software Application calls the Data Interface Flue Gas Analyzer Transfer Program. The transfer program reads the measurements from the instrument and creates the specified file. Or in case you want to transfer the data to the analyzer, the specified file is transferred to the instrument.

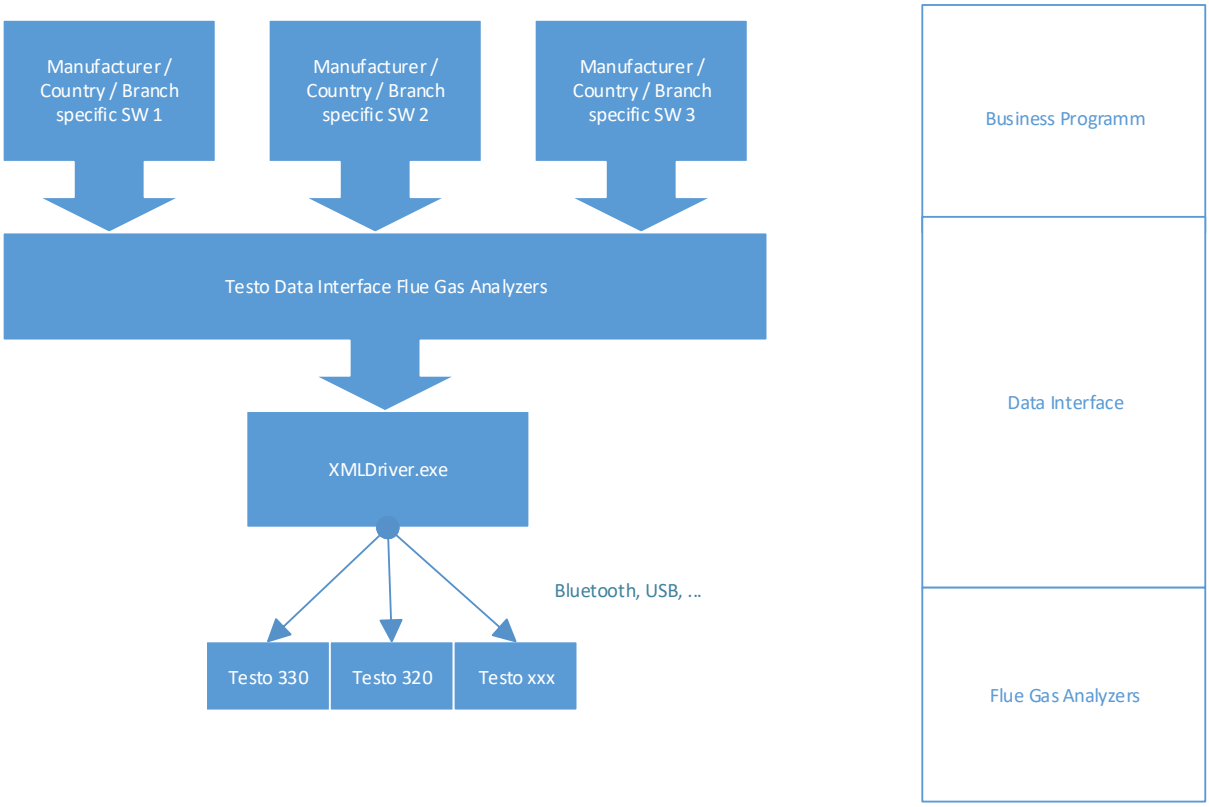

Figure 1, Overview

3.2 File format

For the data transfer we use a text file with a hierarchical structure. The structuring is done by the XML (Extensible Markup Language) format and the file gets the XML extension (XML-File). XML is a markup language that defines a set of rules for encoding documents in a format that could be read from humans or machines.

Additional to this documentation there is always the XSD file (XML schema definition) which describes the elements of the specified XML file. With this XSD file you can check the compatibility of any XML file transmitted from or to a Testo analyzer. If you transmit an invalid XML, the data interface always answers with an error message. Many modern development environment can assist you to import and export XML Files according to a XSD scheme definition. Often there is the possibility to create code with the help of the XSD file.

```
<xs:element name="PcToInstrument" type="PcToInstrumentType" />
<xs:element name="TransmissionResult" type="TransmissionResultType" />
<!-- Call: ProgamName.exe -S PcToInstrument.xml TransmissionResult.xml-->

<xs:element name="TransmissionInfo" type="TransmissionInfoType" />
<xs:element name="InstrumentToPc" type="InstrumentToPcType" />
<!-- Call: ProgamName.exe -R TransmissionInfo.xml InstrumentToPc.xml-->

<xs:element name="SingleStart" type="SingleStartType" />
<xs:element name="SingleResult" type="SingleResultType" />
<!-- Call: ProgramName.exe -O SingleStart.xml SingleResult.xml-->

<xs:element name="InstrumentInfo" type="InstrumentInfoType" />
<!-- Call: ProgramName.exe -I InstrumentInfo.xml-->

<xs:complexType name="PcToInstrumentType">
  <xs:sequence>
    <xs:annotation>
      <xs:documentation>
        Data transmission from PC to instrument.
      </xs:documentation>
    </xs:annotation>
    <xs:element name="Version" type="VersionType" minOccurs="1" maxOccurs="1" />
    <xs:element name="Control" type="ControlType" minOccurs="1" maxOccurs="1" />
    <xs:element name="Customer" type="CustomerType" minOccurs="0" maxOccurs="unbounded" />
    <xs:element name="Tester" type="TesterType" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="TransmissionId" type="xs:string" use ="required"/>
</xs:complexType>
```

Figure 2, Excerpt of an XSD File

For the compatibility to different measurement devices, it's generally acceptable that just a part of the fields listed here are in use. The missing of fields should not lead to an error or exit condition of the data interface.

! Some fields are mandatory: Missing one of these fields results in an error message from the Data Interface Flue GAS analyzers. (see Figure 2, Sample: use = "required")

Sample:

```
<xs:attribute name="TransmissionId" type="xs:string" use ="required"/>
```

Figure 3, mandatory attribute

The following descriptions of the transmitting types serves as a clarification of data combinations and the generally structure of the specified XML file. Simple data types are not described in detail in this documentation. The exact using of data types and data structures could be extracted from the XSD file and the "XSD Recommendation of the World Wide Web Consortium".

3.3 Software Components

The Data Interface Flue Gas Analyzer Software consists of three parts:

- The Data Interface Flue Gas Analyzer Transfer Program (XMLDriver.exe)
- The Data Interface Flue Gas Analyzer Configuration Program
- The Data Interface Flue Gas Analyzer Installation which installs the Transfer Program, the Configuration Program and additional help files.

The transfer program is responsible for the transmission of data between the business program and the analyzer. The transfer program is a driver without user interaction. The transfer program has only a small status window. (see also 7. Operating modes)

The configuration program is a separate program to set some configuration variables (e.g. which analyzer, which communication type,...) (see also 4. Data Interface Flue Gas Analyzers Configuration)

4 Data Interface Flue Gas Analyzers Configuration

4.1 Usage

The software Testo Data Interface Flue Gas Analyzers Configuration is used to configure the interface and the computer.
The software configuration supports the following Testo analyzers:
• testo 300 (Version 2010)
• testo 320
• testo 330 (Version 2010 with color display)
• testo 330 (Version 2010 with color display / Testo 380 (Fine particle measurement)
Analyzers can be connected via Bluetooth (optional analyzer function) or USB.
The USB driver can be installed together with the software testo Data Interface Flue Gas Analyzers configuration, see 4.3 Installing the software and configuration software
In order to use the Bluetooth interface, a connection must be set up between analyzer and PC, see 4.3 Bluetooth pairing.
Knowledge of Windows® operating systems is required when working with the software.

4.2 System requirements

Administrator rights are required for installation.
Operation system
The software requires Microsoft Windows 8.1 or higher
32 bit and 64 bit operating systems are supported. Microsoft Windows 8 RT for ARM processors (tablets and smartphones) is not supported.
Computer
The computer must meet the requirements of the corresponding operating system. The following requirements must additionally be fulfilled:
• Interface USB 2.0 or higher
• Optional: Bluetooth module

4.3 First steps

4.3.1 Installing the transfer program and the configuration software

If Microsoft .NET 4.7 Framework is not present on the computer, it will be installed automatically. The Data Interface Flue Gas Analyzers cannot be run without installed Microsoft .NET 4.7 Framework.
The USB drivers for analyzers testo 300, testo 320 and testo 330 can be installed automatically (default setting). Without installed USB driver it is not possible to set up a connection to the analyzer via USB interface.
1. Download the program (**www.testo.com/download-center**) and unpack the zip file using a suitable compression program.
2. Start the file **TestoSetup.exe**.
3. Follow the instructions of the installation wizard.
When installing under Vista, please pay attention to the following steps during the installation process:
• The window **User account control** is opened:
> Click on **[Next]**.
• The window **Windows Security** is opened:
> Choose **Still install this driver software**.
4. Click on **[Complete]** to complete the software installation.

4.3.2 Bluetooth pairing

In order to use the Bluetooth interface (optional analyzer function), a connection must be set up between analyzer and PC (pairing).

The handling steps required to do this vary depending on which PC Bluetooth module and operating system are used. Typically, the input template / installation assistant are called up via:
• The Bluetooth symbol in the task bar: Bluetooth symbol
• Windows 7: Start symbols > Devices and Printers > Add Device
The password or the pass key for the instrument identification for all testo instruments is: 1234
If necessary, also refer to the documentation of your Bluetooth module or operating system.

testo 300 doesn't require pairing.

4.3.3 Starting the software

If Bluetooth interface is used, first start bluetooth connection. Afterwards Data Interface Flue Gas Analyzer Configuration software can be started.

Start Data Interface Flue Gas Analyzers configuration software
The software user interface is opened in the language of the operating system, if this is supported. If the operating system language is not supported, the user interface is in English.
> Click on **[Start]** | **All Programs** | **Testo** | **Data Interface Flue Gas Analyzers Configuration**.
On Windows Vista, the window **User account control** is opened when the software is started the first time.
> Click on **Accept**.

4.4 Using the product

Help button
If pressed on **[Help]** user manual is shown in pdf-format.

4.4.1 Analyzer

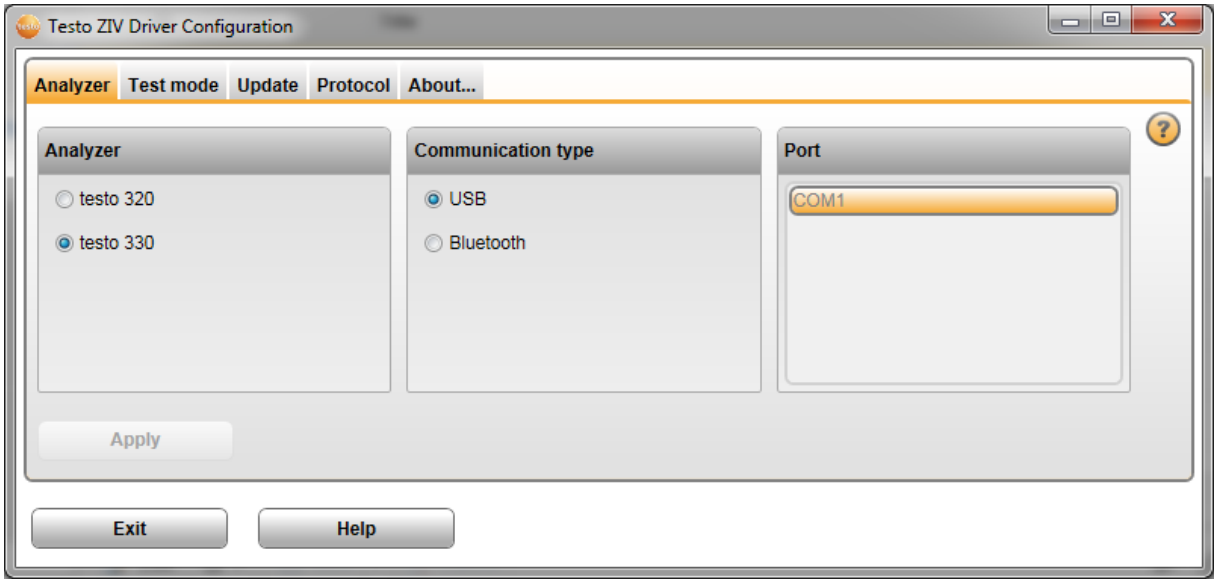

Figure 4, Configuration Analyzer

Data exchange between Analyzer and PC is only possible if analyzer and communication type is configured correctly.
In this menu the used analyzer and the communication type can be set. With the selection of communication type Bluetooth the port must additionally be set.
> Choose **Analyzer**.
> Choose **Communication type**.
> With the selection of communication type Bluetooth: Choose **Port**.
> Save settings with **[Apply]**.

4.4.2 Test mode

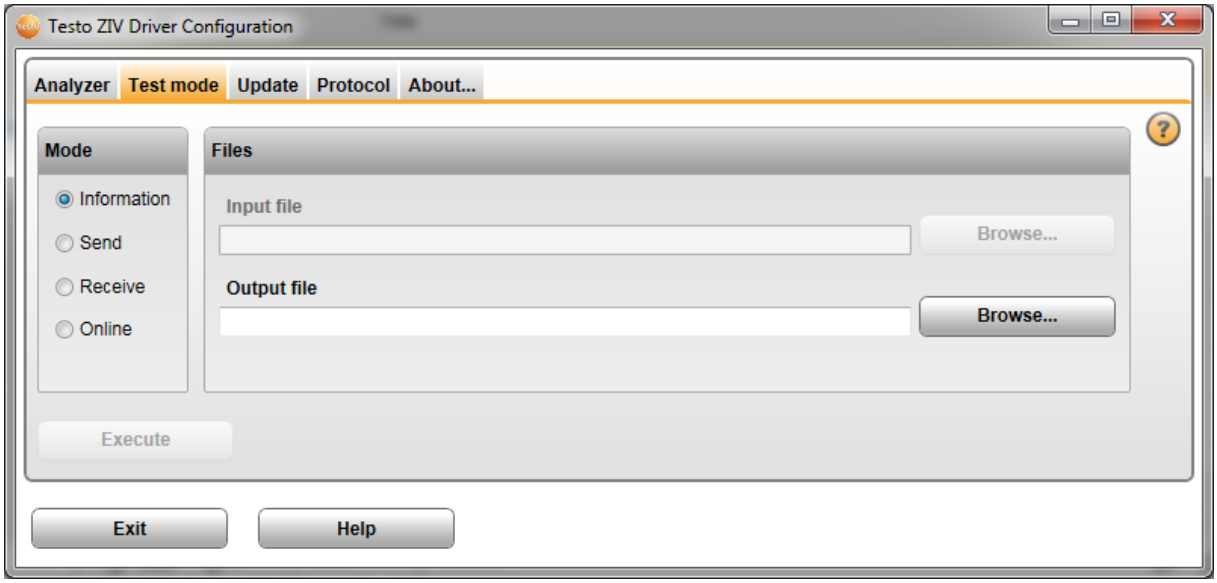

Figure 5, Configuration Test Mode

This menu is used for support purposes only. If necessary,
a support representative will assist you.

4.4.3 Update

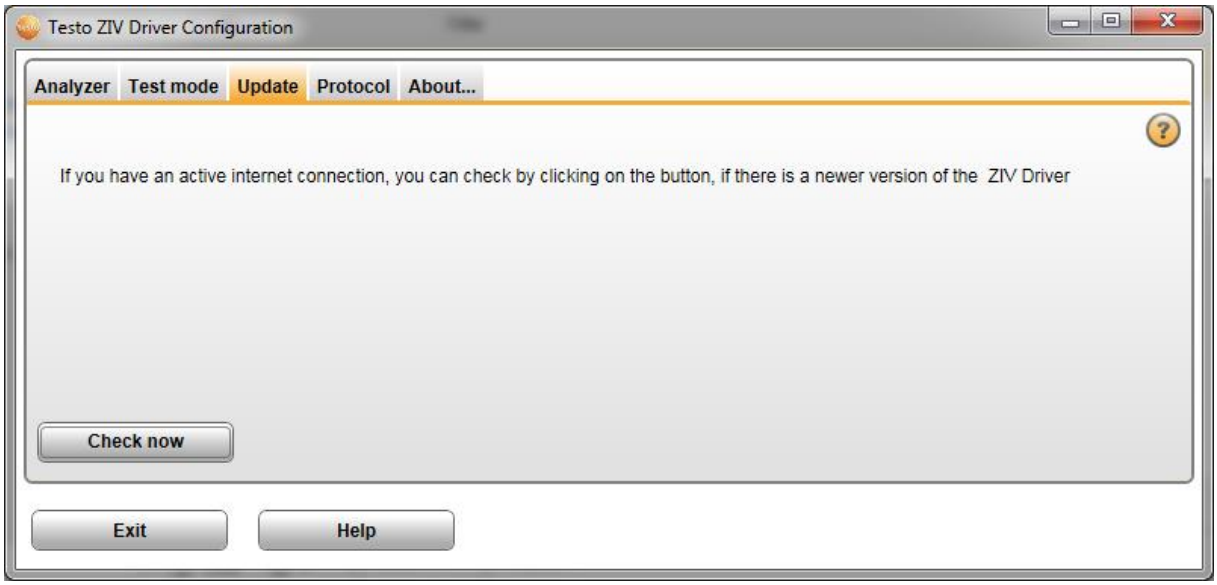

Figure 6, Configuration Update

In this menu it can be checked if the installed testo Data Interface Flue Gas Analyzers is upto-date.
> Click on **[Check now]**.

4.4.4 Protocol

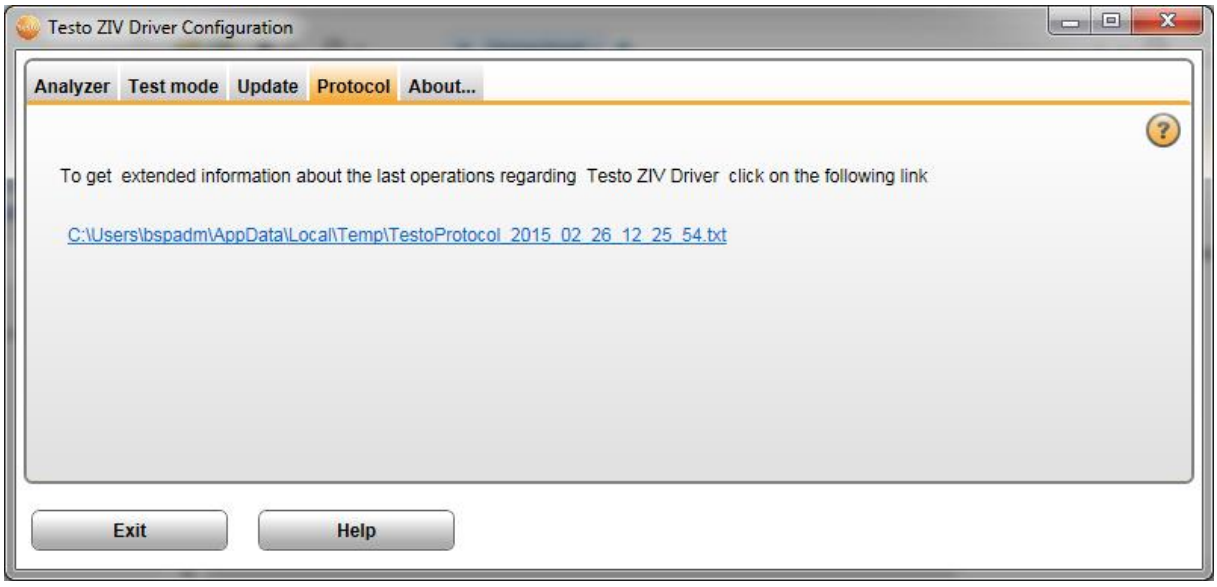

Figure 7, Configuration Protocol

This menu is used for support purposes only. If necessary, a support representative will assist you.

4.4.5 About …

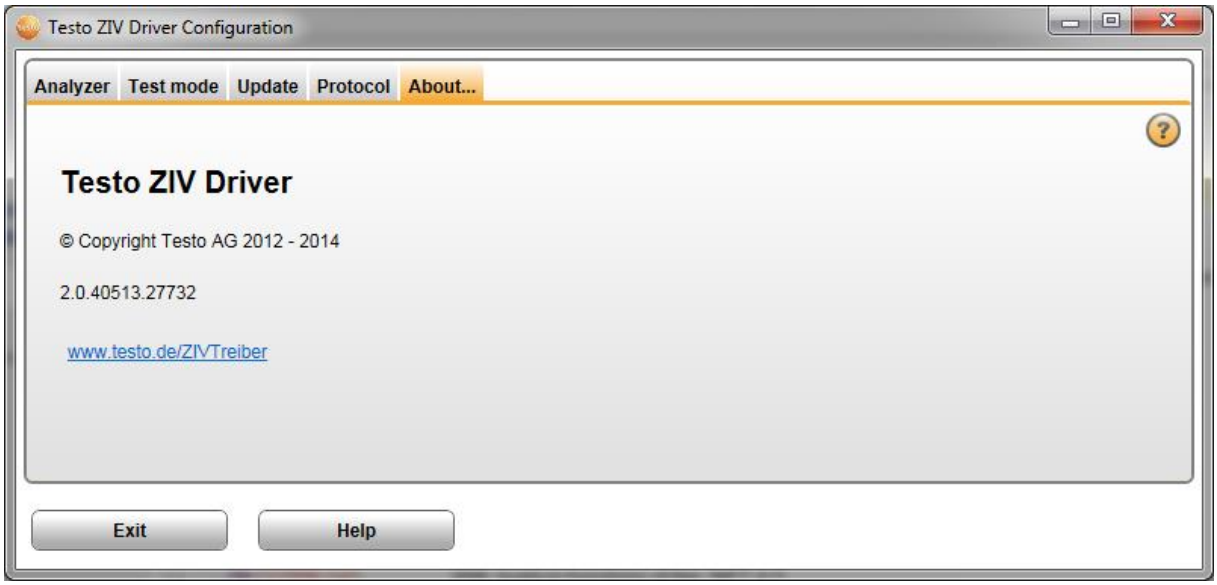

Figure 8, Configuration About…

This menu contains information about the installed software and status.

5 Country specific documentation

After installation of the Data Interface Flue Gas Anylyzer software there is a subfolder in your installation path with detailed country specific information.

In the country specific documentation you can find a mapping between the XML tags delivered by the data interface and the display names in the instrument. Tables are available e.g. for:

- Channel naming
- Unit naming
- Fuel numbers
- System types
- …

The path to this folder is generally: %InstallDir%/Documentation/Detailed Tables

6 Windows Registry settings

The data transfer program (XMLDriver.exe) which is called from an extern business software is responsible to create the XML response files from the flue gas analyzers, or in case data should transfered to the analyzer it transfers the files to the analyzer.

The transfer program can be detected with help of it's registry settings.
The information of detection can be found under the following registry key:

HKEY_CURRENT_USER\Software\FluegasAnalyzerInterface\Testo\XMLDriver

The „Filename " registry value specifies the path to the transfer program.
The „Types" registry value specifies a list of suitable device types.

Sample:

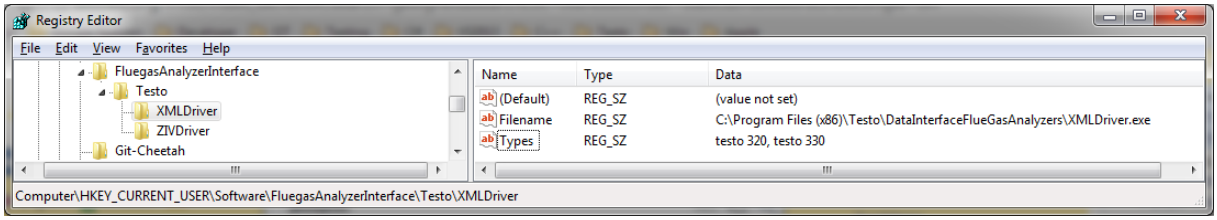

Figure 9, Registry transfer program detection

The configuration of the transfer program is normally done with help of the Data Interfce Flue Gas Analyzer Configuration Program (see also 4. Data Flue Gas Analyzer Configuration Software), but it's also possible to do this configuration directly in the Windows Registry.

The configuration information can be found under the following registry key:

HKEY_CURRENT_USER\Software\Testo\XMLDriver

The „Analyzer“ registry value represents the configured analyzer type. Possible entries are „330“ or „320“.
The „Connection“ registry value represents the connection type. Possible entries are „USB“ or „BT“)
The „Port“ registry value represents the port, which has to be configured for for a bluetooth connection.
The „UnpluggedPrefix“ registry value allows to limit the Bluetooth connection to a given testo 300 by adding „testo 300 <SERIALNUMBER>“ (replace <SERIALNUMBER> with the actual serial number). If you do not specify a value connection will be established to the first testo 300 found.

Sample:

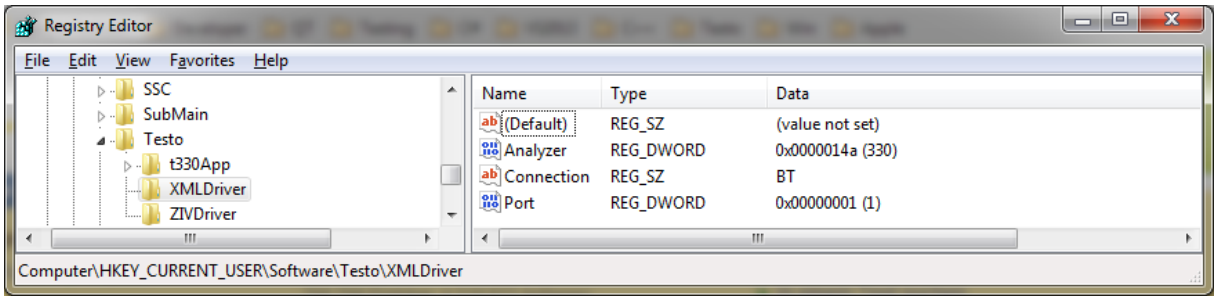

Figure 10, Registry configuration

7 Operating Modes

The Data Interface Flue Gas Analyzers Transfer Program has four basic operating modes

- **Information:** Get information about the data interface and connected analyzer
- **Send:** Send information to the analyzer
- **Read:** Read measure data from the analyzer
- **Read with delete:** Read with deletion of the measure data on the analyzer
- **Online:** Read actual measure data of analyzer

7.1 Information

This mode returns information about the driver itself (e.g. version) and the connected measure analyzer (model, serial number, ...)

Calling Sample

```
XmlDriver -I InstrumentInfo.xml
```

The file InstrumentInfo.xml is written by the Data Interface Flue Gas Analyzer as output. It contains a structure of the type **InstrumentInfo** and contains information about the data interface driver and the measure analyzer.This mode has no input file.

7.1.1 InstrumentInfo

Tag		Description	Type	Multiplicity
Instrument Info		Request result of an information data set	InstrumentInfo Type	
	Version	Version of the interface	string	1 x
	Instrument	Information of the intrument's properties	InstrumentType	1 x
	TransmissionState	Information if transmission is Ok or in an error state	TransmissionState Type	1 x
	ErrorText	Information if transmission is Ok or in an error state	string	1 x

7.2 Send

This mode is used to send location information to the measure instrument.

Calling Sample

```
XMLDriver -S PcToInstrument.xml Result.xml
```

The file PcToInstrument.xml is used as input file and should contain the location information. It is a structure of the type **PcToInstrument**. There is a sample of this file in your intallation directory:

„TestInputSend.xml“.

The result of this operation is returned as output file in Result.xml which is a structure of the type **TransmissionResult**.

If you have requested this device information before, the data can be adopted to the abilities of the connected analyzer. The data type “Fireplace” can include information of the subordinated measurements. The correlation of individual customers, fire place sites and fuels is done by unique attributes. (see 4. country specific documentation)

7.2.1 PcToInstrument

Tag	Description	Type	Multiplicity
PcToInstrument	Transfer of custumer tables to the instrument	PCToIntrumentType	
Version	Version of the interface	string	
Control	Controlling data for the transferring action	ControlType	1x
Customer	Table with customer data	CustomerType	0..9999
CustomerID	Identifier of the customer	CustomerIDType	1
Address	Address data of the customer	AddressType	
FirePlace	Fire place of the customer	FirePlaceType	1 x
FirePlaceNumber	Number of the fireplace	Integer	1
FirePlaceID	Identifier of the fire place	FirePlaceIDType	0..1
RequiredMeasurements	List of the required measurements	Required MeasurementType	0..1
Address	Address of the fire place	AddressType	0..1
InstallPlace	Put up place	string	0..1
Fuel	Fuel type data	TransmissionState Type	0..1
Boiler	Boiler type data	BoilerType	0..1
Tester	Table with tester data	TesterType	0..9999
TesterNumber	Identifier of the customer	Integer	1
Address	Address of the tester	AdressType	1
TransmissionID	Number of the transmission The transmission id ensures, that the result file could be assigned correctly.	string	1

7.2.2 TransmissionResult

After a transmission to the instrument, the Testo XML Interface supplies a result file. With this information a PC Software Application could detect a successful transmission and if necessary output an error message.

Tag		Description	Type	Multiplicity
TransmissionResult		Result of a transmission to the instrument	TransmissionResultType	
	Version	Version of the interface	string	1
	Instrument	Information of the intrument's properties	InstrumentType	1
	TransmissionState	Information if transmission is Ok or in an error state	TransmissionStateType	
	ErrorText	Error text if necessary	string	0..1
	TransmissionID	Number of the transmission, the transmission id ensures, that the result file could be assigned correctly	string	1

7.3 Read

This mode is used to read the saved measure data on the instrument to the PC.

Calling Sample

```
XMLDriver -R TransmissionInfo.xml InstrumentToPc.xml
```

The file TransmissionInfo.xml is used as input file and should contain information to verify the transmission. It is a structure of the type **TransmissionInfo**. There is a sample of this file in your intallation directory: „TestInputRead.xml".

The saved measure data on the instrument are read to the PC. These data are written in the file InstrumentToPC.xml as output file which is a structure of the type **InstrumentToPc**.

7.3.1 InstrumentToPc

Measurement data inclusive customer data is transferred to the PC. The data structure renders a hierarchical assignment of Customer – Fire Place – Measurement Data. Besides the instruments data the transmission contains customer data, each customer can contain several fire places and each fireplace contains contain several measurements.

Tag		Description	Type	Multiplicity
	InstrumenToPc	Transfer of measure and custumer data to the PC	IntrumentToPcType	
	Version	Version of the interface	string	
	Instrument	Information data set to the instrument	InstrumentType	1
	Customer	Table with customer data, consider the max numbers of customers	CustomerType	0..9999

	CustomerID	Identifier of the customer	CustomerIDType	1
	Address	Address data of the customer	AddressType	
	FirePlace	Fire place of the customer	FirePlaceType	1 x
	FirePlaceNumber	Number of the fireplace	Integer	1
	FirePlaceID	Identifier of the fire place	FirePlaceIDType	0..1
	RequiredMeasurements	List of the required measurements	Required MeasurementType	0..1
	Address	Address of the fire place	AddressType	0..1
	InstallPlace	Place of the installation	string	0..1
	Fuel	Fuel type data	TransmissionState Type	0..1
	Boiler	Boiler type data	BoilerType	0..1
	Tester	Table with tester data	TesterType	0..1
	GeneralMeasurement	Currently all measurements are transferred as a GeneralMeasurementType	GeneralMeasurementType	0..9999

7.4 Read with delete

This mode is used to read the saved measure data on the instrument to the PC and deletes all the read measure data on the instrument. The Custumers and Fireplaces are not deleted with this command. Because this mode is critical to some users this mode is not accessable in the „Test mode-Tab“ of the Data Interface Flue Gas Analyzer Configuration Program (see also 4. Data Flue Gas Analyzer Configuration Software).

Calling Sample

```
XMLDriver -RD TransmissionInfo.xml InstrumentToPc.xml
```

The file TransmissionInfo.xml is used as input file and should contain information to verify the transmission. It is a structure of the type **TransmissionInfo**. There is a sample of this file in your intallation directory: „TestInputRead.xml“.

The saved measure data on the instrument are read to the PC. These data are written in the file InstrumentToPC.xml as output file which is a structure of the type **InstrumentToPc** (see also 7.3.1. InstrumentToPc).

7.5 Online

This mode is used to read the actual measure values from the instrument (cached measurement values).

Calling Sample

```
XMLDriver -O SingleStart.xml SingleResult.xml
```

The file SingleStart.xml is used as input file and should contain information to verify the transmission. It is a structure of the type **SingleStart**. There is a sample of this file in your intallation directory: „TestInputOnline.xml“.

The actual (online) measure values of the instrument are transferred to the PC and written to the output file SingleResult.xml as output file which is a structure of the type **SingleResult**.

7.5.1 SingleStart

This transmission prepares a contemporary single measurement. The fuel of the instrument can be set. Afterwards the measurement result is transferred as a SingleResult.

! There could be stored more than one measurement in the cache of the analyzer. In the result file (SingleResult) these measurements are merged to one GeneralMeasurement.

Tag	Description	Type	Multiplicity
SingleStart	Preparing of a single measurement, PC to instrument	SingelStartType	
Version	Version of the interface	string	1
TransmissionID	Number of the transmission, the transmission id ensures, that the result file could be assigned correctly	string	1
FuelID	Label of fuel	integer	0..1
RequiredMeasurements	Required measurements, this is for future use, currently this element is not used	Required MeasurementType	0..1

7.5.2 SingleResult

Measurement data of single measurement which has been prepared through SingelStart are transmitted to the PC.

Besides the instrument information the fire place data set is transmitted

Tag	Description	Type	Multiplicity
SingleResult	Result of a single measurement	SingleResultType	
Version	Version of the interface	string	1
Instrument	Information of the intrument’s properties	InstrumentType	1
FirePlace	Fire place of the customer	FirePlaceType	0..1
FirePlaceNumber	Number of the fireplace	Integer	1 aus 2
FirePlaceID	Identifier of the fire place	FirePlaceIdType	1 aus 2
Address	Address of the fire place	AddressType	0..1
InstallPlace	Place of the installation	string	0..1
Fuel	Put up place	FuelType	0..1
System	System type	SystemType	0..1

	PressureBarometric	Barometric pressure	MeasValueType	0..1
	Altitude	Altitude	MeasValueType	0..1
	PressureAbsolute	Absolute pressure	BoilerType	0..1
	Tester	Table with tester data	TesterType	0..1
	GeneralMeasurement	Currently all measurements are transferred as a GeneralMeasurementType	GeneralMeasurementType	0..9999

7.6 Complex data structures

In the following complex data structures for the transmission are described. These structures can itself contain other structures. Simple structures are not described here. The according information could be looked up in the XSD file.

7.6.1 Instrument information

The instrument information data set contains the information of the device, data management, etc…

Tag		Description	Type
Instrument		Information about the properties fo the instrument and it's data sets	InstrumentType
	Manufacturer	Manufacturer of the instrument	string
	Type	Instrument type	string
	Serial Number	Serial number of the instrument	string
	Certifications	Information about the instrument's certification	string
	Properties	Properties of the instrument	InstrumentPropertiesType
	LastCheckdate	Last check date	date
	TestCenter	Test center	string
	MaxNumCustomers	Maximal number of customer data sets	Integer
	MaxNumFirePlaces	Maximal number of fire place data sets	Integer
	MaxNumTesters	Maximal number of checkers	Integer
	MaxNumMeasurements	Maximal number of measurements	Integer
	CustumerIdLength	Maximal lenght of the customer id	Integer
	FirePlaceIdLength	Maximal lenth of fireplace id	Integer
	NumCustomers	Number of currently saved customers	Integer
	NumFireplaces	Number of currently saved fire places	Integer
	NumTesters	Number of the saved testers	Integer
	NumMeasurements	Number of the saved measurements	Integer
	SelectedTester	Number of the currently selected checker	Integer
	TimeValid	Is the clock setting OK	Boolean
	FWVersion	Firmware Version of the connected analyzer	string
	LastServiceData	Date of the last performance of a service	date format YYYY-MM-DD
	LastCOSensorCalibrationDate	Date of the last CO sensor recalibration	date format YYYY-MM-DD

Characteristics of properties

PROP_CUST_NUMBER	Instrument uses customer numbers
PROP_CUST_NAME	Instrument uses customer names
PROP_INST_NUMBER	Instrument uses fire place numbers
PROP_INST_TYPE	Instrument uses fire place types
PROP_FIREPLACE_ID	Instrument uses fire place IDs
PROP_TESTER	Instrument uses checkers
PROP_EDITABLE	Instrument allows the input and change of customer data sets
PROP_ERASE	Instrument requires to delete the current data sets before uploading of new customer data
PROP_SINGLE	Instrument allows single measurements
PROP_CUST_DB	Instrument contains customer tables

7.6.2 Address

The address data set used in customers, fire places and checkers.

Address		AddressType
	Name	string
	Company	string
	Street	string
	Postcode	string
	City	string
	Country	string
	Telephone	string
	Fax	string
	Email	string

7.6.3 Fuels

The fuels which can be transferred from and to an instrument are enlisted in the country specific tables (see 5. Country specific documentation).

! Consider that the Testo flue gas analyzers have got a fuel number grouping according to the country version of the instrument. This means, that e.g.: Fuel “Light oil” on an instrument with “German” country version is represented as “100”, but on an instrument with country version “Great Britain” the fuel “Light oil” is represented as “2001”.

! Consider that the fuels for a Testo 330 in combination with Testo 380 (Fine particle measurement) can be transfered from the instrument to the PC but could not be transfered to the instrument.

7.6.4 GeneralMeasurement Type

For representation of the measurements there is a GeneralMeastype introduced. Currently all measurements are mapped to a GeneralMeasurement. To differentiate the different types a GeneralMeasurement contains the specific Type - represented in the tag <MeasurementType> - for each measurement. In the country specific tables all these types are matched to the type representation displayed in the flue gas .

Sample:

```
<GeneralMeasurement>
  <Date>2014-01-27</Date>
  <Time>16:45:26</Time>
  <MeasurementType>MEAS_FLUEGAS</MeasurementType>
  <SmokePumpNumber>39402426</SmokePumpNumber>
  <ValuesNumber>1</ValuesNumber>
  <NOxSurcharge>0.0</NOxSurcharge>
  <T_Flue Unit="UNIT_DEG_C" Average="AVG_NONE" Error="ERR_NONE">34.1</T_Flue>
  <T_Air Unit="UNIT_DEG_C" Average="AVG_NONE" Error="ERR_NONE">32.1</T_Air>
  <GT Unit="UNIT_DEG_C" Average="AVG_NONE" Error="ERR_NONE">33.5</GT>
  <O2 Unit="UNIT_VOL_PERCENT" Average="AVG_NONE" Error="ERR_NONE">17.9</O2>
  <CO2 Unit="UNIT_VOL_PERCENT" Average="AVG_NONE" Error="ERR_NONE">2.27</CO2>
  <Q_Flue Unit="UNIT_PERCENT" Average="AVG_NONE" Error="ERR_NONE">0.5</Q_Flue>
  <Efficiency Unit="UNIT_PERCENT" Average="AVG_NONE" Error="ERR_NONE">99.5</Efficiency>
  <QAgr Unit="UNIT_PERCENT" Average="AVG_NONE" Error="ERR_NONE">0.5</QAgr>
  <Effg Unit="UNIT_PERCENT" Average="AVG_NONE" Error="ERR_NONE">99.5</Effg>
  <PDiff Unit="UNIT_ANGLO_PRESSURE_INHG" Average="AVG_NONE" Error="ERR_NOT_VALID">0</PDiff>
  <CO_Dil Unit="UNIT_ANGLO_GAS_GRAM_PER_HORSEPOWER_HOUR" Average="AVG_NONE" Error="ERR_NONE">0</CO_Dil>
  <CO_Norm_Exh Unit="UNIT_PPM" Average="AVG_NONE" Error="ERR_NONE">0</CO_Norm_Exh>
  <NO_Dil Unit="UNIT_ANGLO_GAS_GRAM_PER_HORSEPOWER_HOUR" Average="AVG_NONE" Error="ERR_NONE">0</NO_Dil>
  <NOx_Dil Unit="UNIT_ANGLO_GAS_GRAM_PER_HORSEPOWER_HOUR" Average="AVG_NONE" Error="ERR_NONE">0</NOx_Dil>
  <Lambda Unit="UNIT_NONE" Average="AVG_NONE" Error="ERR_NONE">6.77</Lambda>
  <Co_Amb Unit="UNIT_PPM" Average="AVG_NONE" Error="ERR_NOT_VALID">0</Co_Amb>
  <CO2Amb Unit="UNIT_PPM" Average="AVG_NONE" Error="ERR_NOT_VALID">0</CO2Amb>
  <O2Sup Unit="UNIT_PERCENT" Average="AVG_NONE" Error="ERR_NONE">21</O2Sup>
  <ExtDraught Unit="UNIT_PA" Average="AVG_NONE" Error="ERR_NOT_VALID">0</ExtDraught>
  <ExtDeltaP Unit="UNIT_PA" Average="AVG_NONE" Error="ERR_NOT_VALID">0</ExtDeltaP>
  <T_Dew Unit="UNIT_DEG_C" Average="AVG_NONE" Error="ERR_NONE">28.2</T_Dew>
  <Soot_1 Unit="UNIT_NONE" Average="AVG_NONE" Error="ERR_NOT_VALID">0</Soot_1>
  <Soot_2 Unit="UNIT_NONE" Average="AVG_NONE" Error="ERR_NOT_VALID">0</Soot_2>
  <Soot_3 Unit="UNIT_NONE" Average="AVG_NONE" Error="ERR_NOT_VALID">0</Soot_3>
  <SootAvg Unit="UNIT_NONE" Average="AVG_NONE" Error="ERR_NOT_VALID">0</SootAvg>
</GeneralMeasurement>
```

Figure 1, GeneralMeasurement type

The GeneralMeasurement type contains always header data and measured values of the measuerment.

In many measurement types, the header data contains a lot of values which doesn't represent measurement channels of the instrument. Often these data are input values from the end user of the instrument. This data is available in the result file as text representations, so the consumer software has to intruduce his own interpeter for this data, e.g. for boolean values. It's also possible that this data contains localized information. Detailed information which types are represented as strings can be found in the XSD schema definition and the country specific tables. (See also 5. Country specific documentation).

Tag		Description	Type
GeneralMeasurement		Measurement data set	OilGasMeasType
	Date	Date of the measurement	date

	Time	Timepoint o the measurement	time
	MeasurementType	MeasurementType	string
	O2Ref	O2 reference	MeasValueType
	O2	O2 concentration	MeasValueType
	CO_Dil	Measured CO concentration	MeasValueType
	CO_Norm	CO concentration normed to O2 base	MeasValueType
	CO_Norm_MgkWh	CO concentration correponding to energy	MeasValueType
	NO_Dil	Measured NO concentration	MeasValueType
	NO_Norm	NO concentration normed to O2 base	MeasValueType
	NO2_Dil	Measured NO2 concentration	MeasValueType
	NO2_Norm	NO2 concentration normed on O2 base	MeasValueType
	NOx_Dil	Measured NOx concentration	MeasValueType
	NOx_Norm	NOx concentrationnormed pn O2 base	MeasValueType
	SO2_Dil	Measured SO2 concentration	MeasValueType
	SO2_Norm	SO2 concentration normed on O2 base	MeasValueType
	T_Air	Ambient air temperature	MeasValueType
	T_Flue	Flue gas temperature	MeasValueType
	Q_Flue	Exhaust gas loss	MeasValueType
	Efficiency	Degree of effectiveness	MeasValueType
	T_Dew	Dew point	MeasValueType
	P_Draft	Draft pressure	MeasValueType
	O2_Exh	O2 Abgaswegeprüfung	MeasValueType
	Co_Dil_Exh	CO Abgaswegeprüfung gemessen	MeasValueType
	CO_Norm_Exh	CO Abgaswegeprüfung Norm auf 0% O2	MeasValueType
	O2_Diff_Ring	O2 Verlust im Ringspalt	MeasValueType
	P_Ring	Druck im Ringspalt	MeasValueType
	T_Ring	Temperatur im Ringspalt	MeasValueType
	Soot_1	Smoke number 1	MeasValueType
	Soot_2	Smoke number 2	MeasValueType
	Soot_3	Smoke number 3	MeasValueType
	SootAvg	Smoke number mean value	MeasValueType
	OilDeriv	Oil derivate	boolean
	T_Boiler	Boiler / HCT	MeasValueType
	Ph_cond	PH-value of condensate	MeasValueType
	...	...	...

Further information see 5. Country specific documentation

7.6.5 Dust particle measurements

If there are dust concentration values available (measured with a Testo 380), these values are included in the measurement. In dust particle measurements (MEAS_PROG_PARTICLE, MEAS_PROG_PARTICLE_TEST) the International Data Interface calculates the average value of several measurments from the instrument.

AVG_NONE	No averaging
AVG_30	30 s averaging (gas/oil)
AVG_900	900 s averaging (solid fuels)
AVG_1800	1800 s averaging (TA Luft)
AVG_3M	Average value of 3 single measurements

Values which are invalid were not considered for the average. This means it is like to have one value less. If there is no valid value, the result (average) is also invalid.

7.6.6 4 PA Measurement

By this measurement a lot of equal pressure values are transmitted as a list.

GeneralMeasType		Measure data set	GeneralMeasType
	Date	Date of the measurement	date
	Time	Time point of the measurement	time
	MeasurementType	MeasurementType	string
	Duration	Duration of the measurement	Integer
	Values	List of the pressure values (1 / second)	FloatListType
	Result	Result OK/not OK	ResultType

8 Glossary

XML- Extensible Markup Language

XSD - XML Schema Definition