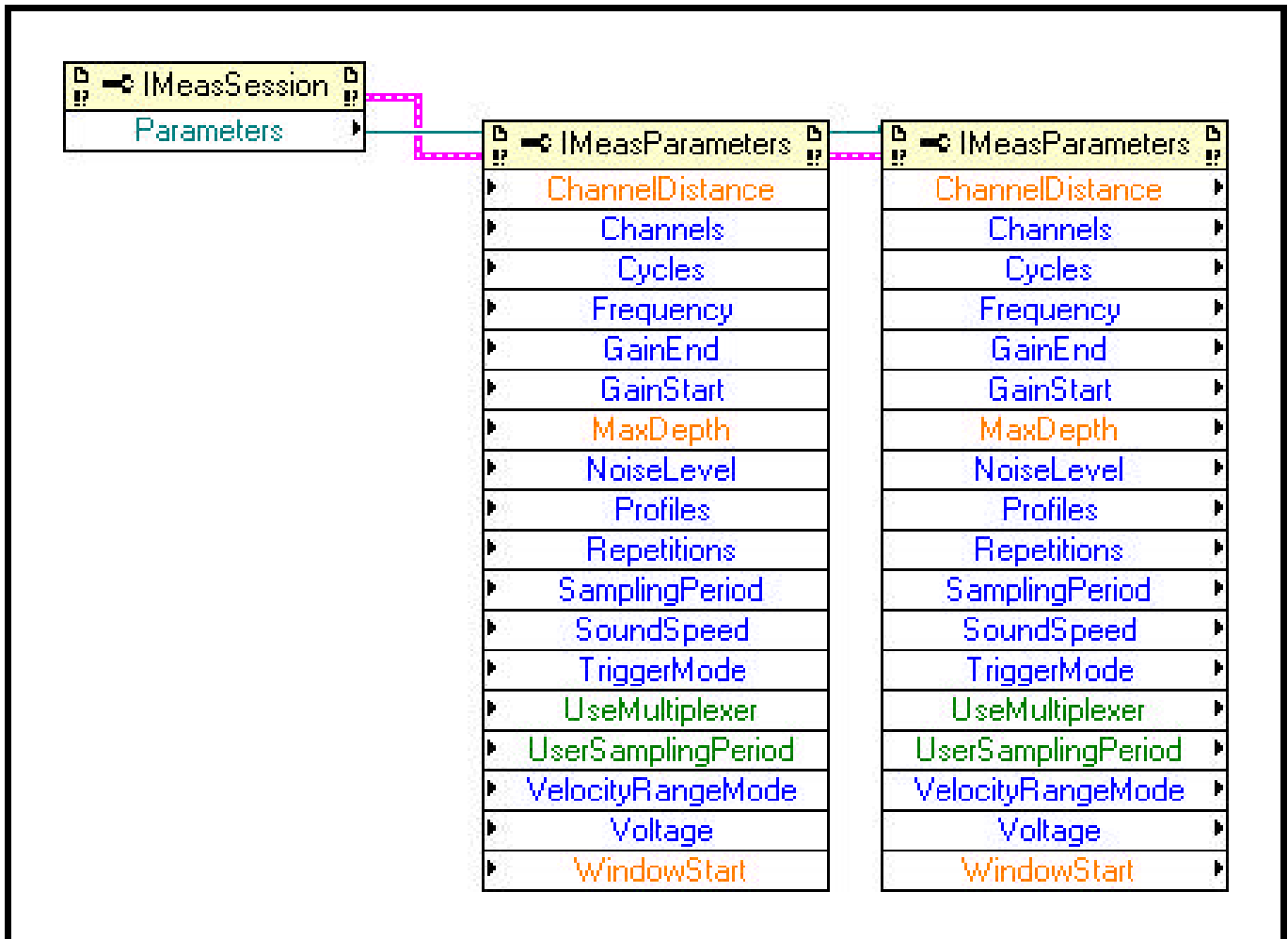


# *ActiveX Objects for UVP-DUO Monitor*



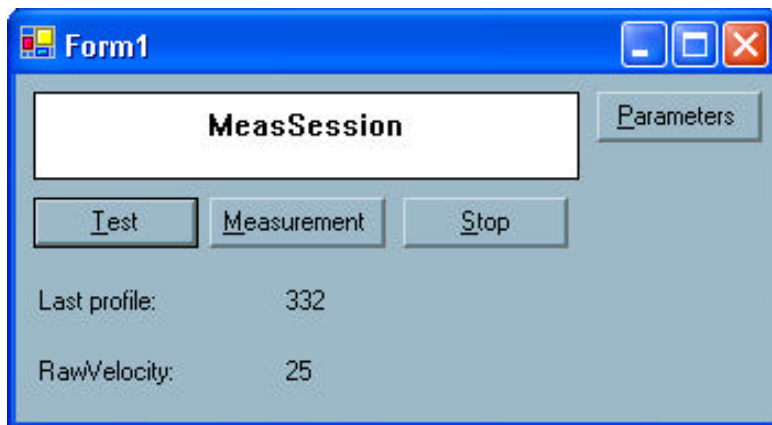
## ***Programming with ActiveX Objects***

- All UVP-DUO functionalities accessible to programmers
- Full compatibility with C++, LabVIEW, Visual Basic 7
- Easy integration of UVP-DUO into measurement systems
- Programming of custom-made features and data analysis
- Trouble-free installation, sample programs provided
- Multiplexer functions fully supported
- ActiveX Objects : a must for industrial users

## What ActiveX Objects do for you

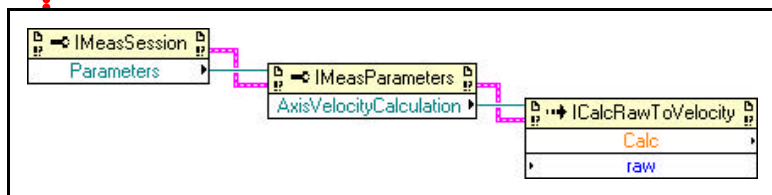
The UVP Software Version 3 provides the total control of UVP-DUO data acquisition and allows for measured data analysis. Nevertheless, it is a stand-alone software, which does not offer the possibility to integrate UVP Monitor into a measuring chain together with other instruments and functions.

For this reason Met-Flow offers ActiveX objects.



Application main window example

ActiveX objects control all UVP-DUO functions and can be used in any programming environment able to work with ActiveX objects, especially C++, Visual Basic and LabVIEW programs. In this way, customers are able to design their own UVP-DUO programs and also to integrate UVP-DUO acquisition into their laboratory measurement chains.



Example of velocity calculation in LabVIEW

## Who can use Activex Objects

Using ActiveX control functions is not trivial, as it requires considerable experience in programming of the respective language used.

Since resulting customer programs are out of Met-Flow's control, Met-Flow cannot be held responsible for any problems or bugs in customer-created programs.

The ActiveX functions are delivered "As Is".

In the Programmer's Guide, names and meaning of all variables agree with the UVP-DUO User's Guide, and frequent references to the User's Guide are made. For understanding this Programmer's Guide, it is therefore necessary to understand the UVP-DUO Monitor User's Guide first.

## UVPAX Library

UVPAX is the library of ActiveX objects for making measurements with UVP instruments and for accessing measured data:

- ❖ [MeasSession](#) control executes a measurement
- ❖ [MeasParameters](#) specifies parameters of measurement
- ❖ [MeasFile](#) stores measured data

There are several helper objects:

- ❖ [InstrumentInfo](#) specifies information on instrument
- ❖ [InstrumentLimits](#) is used for parameter checking
- ❖ [WindowLimits](#) is used for parameter checking
- ❖ [Multiplexer](#) and [MultiplexerItem](#) contain multiplexer parameters
- ❖ [Profiles](#) and [Profile](#) access profile data
- ❖ [CalcRawVelocity](#) calculates actual velocity from raw measured values
- ❖ And many more functions

## File compatibility with mfprof

UVPAX objects can store measurement data in a native UVP-DUO mfprof file format. UVPAX objects can also read measurement data in mfprof file format.

This allows combining work with Met-Flow Version 3 software and custom-made software using UVPAX objects. It is therefore possible to define a measurement set-up with Version 3 software and later run UVP measurements from custom-made software, while using the pre-set and pre-tested parameters.

Conversely, it is possible to analyze data acquired by customer program with Met-Flow Version 3 software.

## Customer programming support

If programming of measurement tasks is not your core business or if you do not feel sufficiently fluent in programming, Met-Flow can suggest you a skilled third-party software developer, who can program your tasks for you on a pay-per-hour base.