Last update: DESY, October 2006
M. Seebach, DESY/MST

This page: http://adweb.desy.de/mst/GAN_MVL/GSI2006/workshop.html
GANMVL VI Server Installation Page: (http://adweb.desy.de/mst/GAN_MVL/GANMVL_VI_Server.html

GANMVL Workshop October 2006, GSI

(1) What have we added and improved since the last Workshop at DESY?

(2) Characteristics of IVI, future of IVI drivers and alternatives

(3) Some points we should solve in future (improvements)

(4) What do we plan to extend? What can be realized until the next year?

(1) What have we added and improved since the Instrumentation Workshop at DESY in august 2006?

(A) Some scope functions are added to the existing applications
(B) Test of a scope application based on IVI-COM instead of IVI-C driver type
(C) Effort to solve the trouble with the newer LabVIEW Version (Scope and DMM)

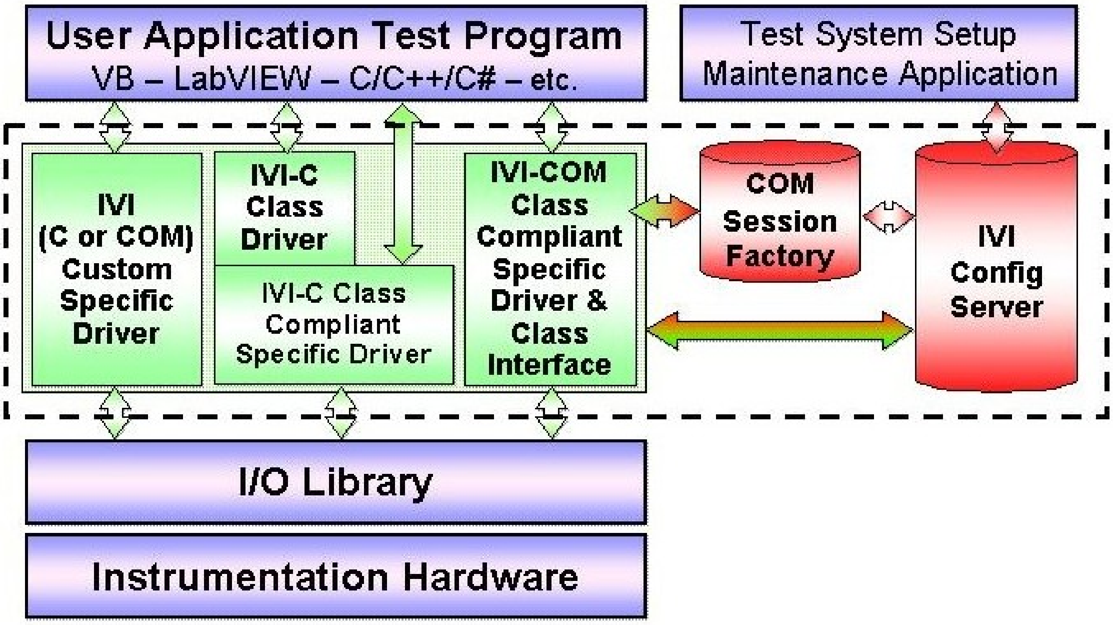
(A) Some scope functions are added to the existing applications

E.g. Scope application: channel offset, acquisition type.
After that I updated the installation ZIP file.

(B) Test of a scope application based on IVI-COM driver instead of IVI-C

Please remind of earlier presentations:

- IVI-C driver are based on DLL Calls. (DLL = Dynamic Link Library) **70%**
- IVI-COM driver use the MS COM architecture (API calls via references). **30% (increasing)**



IVI System Architecture

Result of IVI-COM test:
IVI-COM applications have a complete other driver type and source code, but run as stable as IVI-C applications. Not all functions are identical in IVI-C and IVI-COM, so some applications could work slightly differently.

(C) Effort to solve the trouble with the newer LabVIEW Version (Scope and DMM)

It appeared, that several instruments didn't run with the newer LabVIEW version anymore.
Examples: Tektronix Scopes series TDS3000, Keithley DMM K2701.
TDS3000: Should run, but I couldn't restart it after updating the LabVIEW version.
K2702 DMM: Keithley declared officially that this instrument runs only with the VISA Runtime 3.1,
a newer specific driver will follow later (months).

But:
The point isn't actually the LabVIEW version, the problem is mainly the version of the VISA Runtime.
Remember: VISA is the common address architecture (interface independent).
(VISA = Virtual Instrument Software Architecture)
Problems:
It can only be installed one version of the VISA Runtime!
(In contrast to LabVIEW itself and the LabVIEW Runtime Engine)
The LabVIEW versions change faster than the driver can be updated!
Recommendation:
It is advisable to run with the same LabVIEW Version for a longer time and to check the most important instruments before changing the version.

To the top of page

(2) Characteristics of IVI, future of IVI drivers and alternatives
(IVI = Interchangeable Virtual Instruments)

Here I want to put the issue "preferred instrument driver type" forward for discussion for the last time.

We decided to prefer IVI drivers for controlling T&M instruments.
Why do we use IVI drivers primarily?

Some characteristics of IVI instrument drivers:

- Many different instruments can be controlled with the same application (the same source code)
  This is the most important advantage of IVI!

- IVI drivers run only under the MS Windows operating system.

- At the moment there are two kinds of IVI drivers (IVI-C and IVI-COM)
  Unfortunately, the applications cannot be programmed with the same source code!

- Nearly all new instruments can run with IVI drivers.

- The instrument behaviour is different from instrument to instrument.
  E.g. the reaction to a wrong trigger level (oscilloscope) can be:
    - a decent error message (then stop or continue),
    - no error message, but also no data
    - no error message and untriggered waveform (the best case)
  E.g. the reaction to illegal or unsupported functions can be:
    - a decent error message (the most harmless case)
    - the worst case: the crash of the instrument (or even of the Windows Operating System)
  Result: After insertion of a new instrument, every function has to be tested!

- It is advisable to be careful when the LabVIEW version (and driver version) has to change (see above)
  (Concerns traditional drivers too)
  Recommendation: Work with the same version for a long time!
  (And test all functions accurately before changing the version!)

- IVI drivers are not optimized for a special instrument type, but for common functions, methods and properties. For a special rare feature one has to insert source code for a traditional driver.

Here once more the typical advantages of IVI instrument drivers:

- Many different instruments can be controlled with the same application (the same source code)
  (This is the most important advantage)

- IVI programming has once to be learned (Once learned, always mastered)

- The same repeating configuration style for different instruments (more rapid configuration)

The future of IVI instrument driver

The IVI Foundation ***(http://www.ivifoundation.org/)*** was established, because the vendors have similar interests. The most important vendors of T&M Instruments support IVI drivers, but they also offer their own specific drivers additionally. Some vendors prefer IVI-COM (Agilent, Tektronix) and other prefer IVI-C drivers (LeCroy, National Instrument).
The quota of IVI-COM is going to increase, but IVI-COM driver are not the end of the driver development.
The development was:
1993: VXIPlug&Play
1998: IVI-C
2003: IVI-COM
2008: ???
Do IVI drivers have a future?
Yes, they do, but the IVI driver implementation and configuration are developing and modifying permanently.

Alternatives to IVI Instrument drivers for the GANMVL project could be:

- Implementation of traditional drivers, but usage of only a few instrument types and series
  E.g. Only Tektronix Series TDS7000 and Agilent DSO6000 = two applications for one instrument kind
  But this is not really an alternative!

- Usage of instruments, which can run as web server themselves (without any installation)

- Some instruments have an implemented MS Windows Operating System (e.g. Tektronix TDS5000, LeCroy WR6000). The Windows System could be used, so that the instruments could be controlled remotely with a web browser (e.g. via VNC).

Suggestion:
We should prefer IVI instrument drivers as default.
Additionally, we should be able to use instruments which can run as web server themselves, because nothing is to be installed (e.g. Tektronix series TDS3000). As well we should provide to use instruments with a Windows OS so that they are full controllable via VNC.
Eventually we must define a list of instruments, which are preferred by the GANMVL, e.g. all IVI controllable instruments, instruments with web server support and instruments with an implemented Windows OS.

To the top of page

(3) Still to do and to solve in future

What are the objectives of the VIs in general?
- One goal is to develop applications:
  1. run stable (with a decent error handling and without crashes)
  2. usable for many different T&M Instrument types
- The applications have to be as most as possible independent from new driver trends and more instrument

variety
- Furthermore to have a concept for Virtual Instruments, that can be used a long time.

Three other points we should solve (demands)

(A) Show, which instruments are available and specify them to a particular instrument class (the proper application and URL)

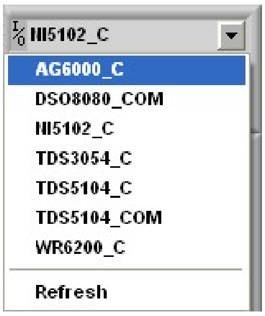
(B) Create a possibility to configure the IVI setups remotely

(C) Create a possibility, to search for installable instruments remotely

All three points refer to the NI "Measurement & Automation Explorer" (MAX), which is used for the configuration of the IVI addresses, driver sessions, software modules and hardware assets with a proper VISA address.
Image of a MAX (Version 4.1.0):
(http://adweb.desy.de/mst/GAN_MVL/GSI2006/MAX.jpg)

!!! And all three points are not provided or supported by National Instruments !!!
(i.e. it is only possible by additional software development.)

(A) Show available instruments and specify them to a particular instrument class (the proper application and URL)



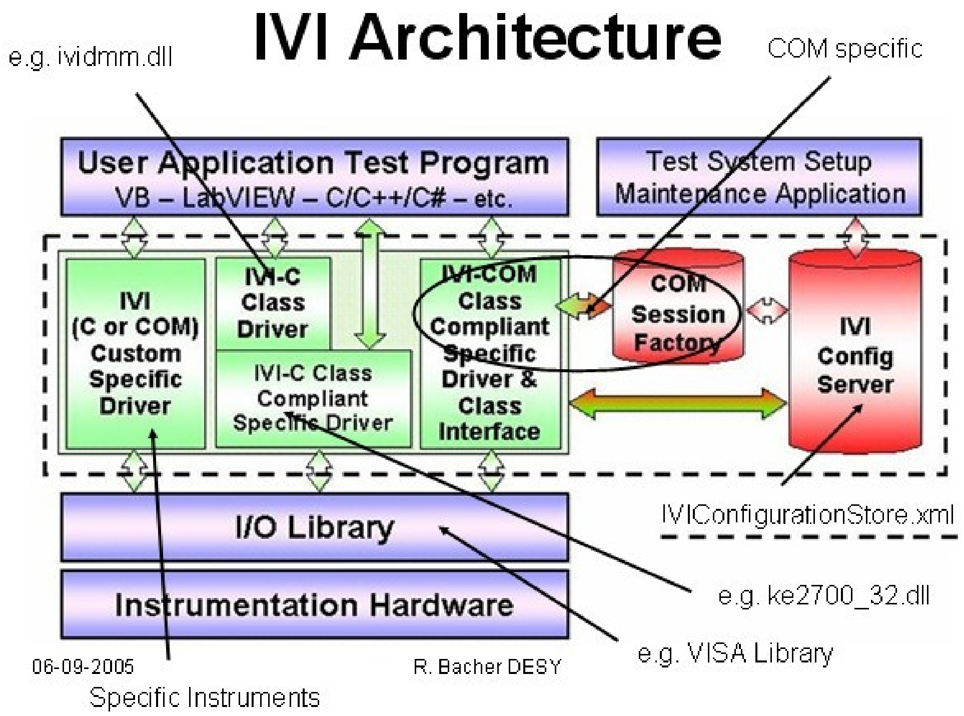
LabVIEW has a frontpanel control for the IVI addresses, it shows all installed IVI instruments of a special IVI class, but the control don't show, if the instruments are available.

In order to show all available instruments it is necessary to develop a special application separately (or sub source code in an application).
One possible way is to send a query (e.g. *IDN? as string) to all installed instruments and they will return their IDs or one get an error code. Then one can sign the address as available (or not), allocate the IVI address to a IVI class and to the right application and the proper URL.
It is quite difficult to design this program part, but we should include this feature because it is very advantageous.

(B) Create a possibility to configure the IVI setups remotely

Remote configuration of the MAX:

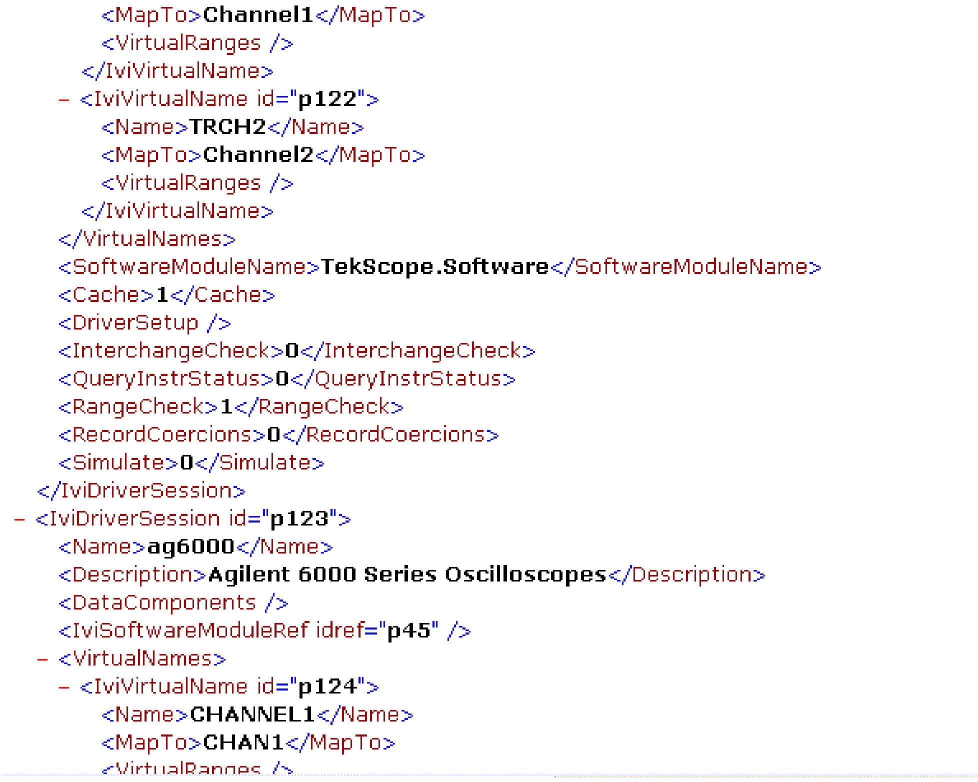
The NI MAX is not provided for remote control. Eventually it can be configured remotely by editing the XML file or by remote control via VNC log-in.



(From the PPTof R.B. Oct 2005)

The IVI configuration is stored automatically at:
…\Program Files\IVI\Data\IviConfigurationStore.xml

A short part of the XML configuration file:



```
         <MapTo>Channel1</MapTo>
         <VirtualRanges />
       </IviVirtualName>
     - <IviVirtualName id="p122">
         <Name>TRCH2</Name>
         <MapTo>Channel2</MapTo>
         <VirtualRanges />
       </IviVirtualName>
     </VirtualNames>
     <SoftwareModuleName>TekScope.Software</SoftwareModuleName>
     <Cache>1</Cache>
     <DriverSetup />
     <InterchangeCheck>0</InterchangeCheck>
     <QueryInstrStatus>0</QueryInstrStatus>
     <RangeCheck>1</RangeCheck>
     <RecordCoercions>0</RecordCoercions>
     <Simulate>0</Simulate>
   </IviDriverSession>
 - <IviDriverSession id="p123">
     <Name>ag6000</Name>
     <Description>Agilent 6000 Series Oscilloscopes</Description>
     <DataComponents />
     <IviSoftwareModuleRef idref="p45" />
   - <VirtualNames>
     - <IviVirtualName id="p124">
         <Name>CHANNEL1</Name>
         <MapTo>CHAN1</MapTo>
         <VirtualRanges />
```

It is absolutely necessary to understand the structure of the XML file and it is risky to edit the XML file after changing the MAX Version due to an eventual new file structure.
The worst case: The XML file structure can be destroyed in case of wrong editing and nothing will run anymore!
Suggestion: I suggest not realizing this feature by editing the XML file.

(C) Create a possibility to search for available instruments which can be installed

The "Measurement & Automation Explorer" (MAX) can be used for searching for available installable instruments. If the procedure is started, the MAX searches in the same local subnet for instruments. To perform this procedure remotely, it is possible with a VNC connection.

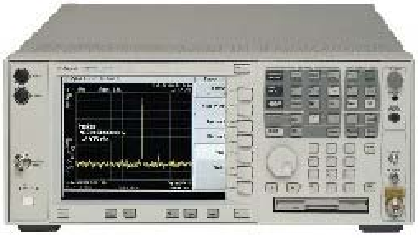
To the top of page

(4) What do we plan to extend? What can be realized until the next year?

Primarily: One idea is to add another IVI instrument class.
Which useful IVI T&M Instrument class should be added?

There are eight IVI classes defined by the IVI Foundation:
- DC power supply
- Digital multimeter (DMM)
- Function & Arbitrary generator
- Oscilloscope
- Power meter
- RF signal generator
- Spectrum analyzer
- Switch

The most useful IVI class after oscilloscope and DMM seems to be:
The Spectrum Analyzer class.

		
Rohde & Schwarz	**Keithley Instruments**	**Agilent Technologies (earlier HP)**

Furthermore, we should provide applications with IVI-COM source code for the VI server installation and realize the point 3 (A): Show available instruments!

To the top of page