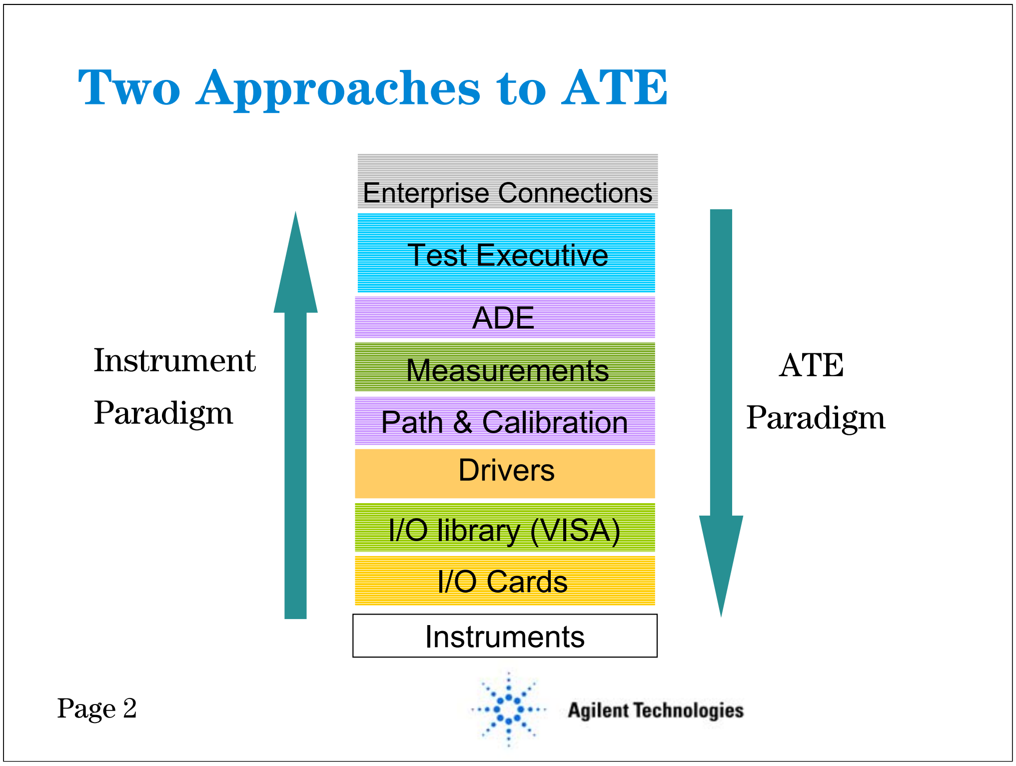
Different Instrument - Same Answer!

Roger Oblad
Bob Stern
June 28, 2000

Agilent Technologies
Innovating the HP Way



Instrument Paradigm

Before the general availability of commercial software development tools, test executives, and LANs to link computers, test system development began with a list of necessary measurements to be made. Then people would get out instrument catalogs and make a list of instruments with the required specifications that could make the measurements. They would also make sure that the instruments selected had IEEE-488 interfaces. After selecting a computer, and IEEE-488 interface, they decided which language that they wanted to write their code in. Then they would write instrument drivers and automatic measurements based on these drivers. Finally, for the benefit of test operators they would develop a simplified operator interface.

While this approach works, it provides no interchangeability, so significant software redevelopment is required whenever it becomes necessary to change an instrument (obsolescence, add new capabilities, etc.) and it is very difficult to use a measurement program written for one test system on another unless it is identical.

ATE Paradigm

PCs running Windows operating systems are the overwhelming choice of system computer. ATE developers spend more money developing and maintaining programs than the cost of the instruments in the systems they develop. So they naturally rely on commercial software tools to keep the scope of development reasonable. Developers now first choose their application development environment. When possible, they prefer to purchase software containing the measurements and other necessary elements, (drivers, I/O, etc.). Still, often it is necessary for them to write their own measurements. Nevertheless, developers are anxious not to have to develop instrument drivers themselves. The choice of instruments is influenced by those that have good drivers available in the application development environment (ADE) that they have selected. Finally, the desire for reuse of measurement components is driving many developers to define appropriate abstract interfaces within their own companies. The desire for reuse is also a prime motivator in various standards activities as well.

Key Problems With ATE Systems:

- Interchangeability
- Interoperability
- I/O interface issues
- Instrument drivers
- Distributed computing environments

Agilent Technologies

Most practical working ATE systems employ an overall architecture similar to the one shown on the previous slide. There are a number of issues that often arise. We have listed here the top areas where problems arise. Although we will focus mainly on interchangeability and necessarily software drivers, we will partially address the other areas as well given our time available.

Interchangeability

- Ability to replace a hardware asset (instrument) with one of a different design
  - Test assets become obsolete
  - Test assets need to be modernized

- NO changes in user application software or TPS*

Different Instrument: Same Answer!

Agilent Technologies

You hear and read a lot about interchangeability. In fact, there has been confusion over claims of interchangeability. We have found it is useful to define exactly what we mean. The definition we use is based directly on the inputs of many customers, over a number of years.

Interchangeability is defined as the ability to change an instrument (and corresponding driver component/system configuration data) and to get the same measurement result! As you will see from this presentation there is no simple silver bullet solution to instrument interchangeability. However, there are industry standard efforts that provide increasing levels of interchangeability including solutions that can guarantee the same answer after changing an instrument.

Note: TPS = Test Program Set, the application software and any associated special cables required to connect a device under test to the test system.

Interoperability

Ability to move a test sequence from one system to another for measurement reuse.

- Ability to run TPS on multiple configurations
- Very difficult problem to solve
- Asset interchangeability key step towards solution

Interoperability is another term that you often hear mentioned and there is probably more confusion about it than even interchangeability. We thought it would be a good idea to include the operating definition that we use at Agilent, again based on listening to many customers for a number of years.

It is beyond the scope of what we can realistically cover In our limited time today. Nevertheless, we do want to point out that achieving robust interchangeability is a very necessary step towards achieving interoperability.

Characteristics of ASCII Interfaces
I/O Interface Issues

- Primary automation interface
  - supported & warranted by instrument suppliers
- High quality - part of product definition
- Complete, full functionality exposed
- SCPI is the Industry standard

"FREQ1930.08MZ"

Agilent Technologies

For the last 20 years instrument manufacturers have been focusing on the ASCII programming interface to their instruments (usually sent over GPIB)

(a) This is the primary automation interface supported by instrument manufacturers; manufacturers warrant that these ASCII functions will work

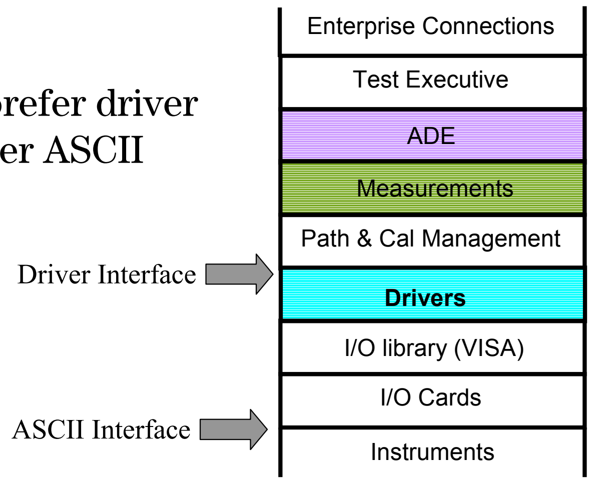
(b) High quality - the best firmware engineers are assigned to this as part of product development

(c) Complete, full functionality exposed

(d) The industry standard for using the same terminology when possible is driven by the SCPI Consortium (Standard Codes for Programmable Instruments)

Instrument Drivers

Developers prefer driver interfaces over ASCII commands



Agilent Technologies

Developers using modern application development environments prefer using drivers to control instruments over using native ASCII interfaces. Drivers provide the necessary formatting to allow computer programs to setup instruments and return measurement and status information using variables. Visual or graphical application development environments (ADE's) allow instrument control without programming when appropriate drivers are available. For ADE's where programming is necessary, drivers make it easier to control instruments than it is with ASCII string programming.

Problems with Instrument Drivers

- Incomplete: all features and functions not exposed
- Different ADEs have different driver requirements
- Often unsupported by instrument manufacturer
- Users often have to write their own

Agilent Technologies

Problems with Drivers

(1) Reputation of poor quality - often not current.

(2) Sometimes they are incomplete, don't expose all features and functions, or may be limited in parameters accepted.

(3) Different application development environments (ADE) have different driver requirements. This results in multiple drivers needed for each instrument (G-drivers for LabVIEW, Agilent VEE drivers, VXI *Plug&play).*

(4) Instrument manufacturers have found it difficult and expensive to develop and maintain multiple versions of instrument drivers.

(5) Test system developers have been frustrated when they select an ADE from one company and instruments from another because high quality drivers were not always available. Too often,this has meant that they had to write their own drivers.

Industry Standard Efforts Underway

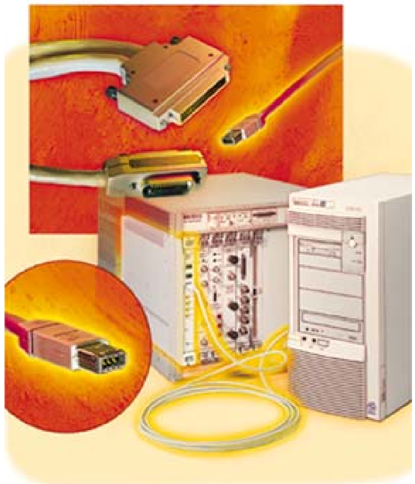
- New computer standard I/O interfaces
- COM computer standard technology
- Interchangeable Virtual Instruments (IVI) Foundation (a T&M industry effort)

Agilent Technologies

This paper presents several industry efforts now underway which address these problems resulting in significant improvements for ATE customers.

- 25 years ago there were no PC's. HP, as a maker of both instruments and desktop computers, realized the need for standard digital communication with instruments and led industry efforts to standardize on IEEE-488 (also known as GP-IB). IEEE-488 has been a very popular test and measurement communication bus. Today, PC's have standard high capacity I/O interfaces, several of which are well suited to instrument I/O.
- COM and DCOM allow reuse across a broad variety of computer standard and T&M industry std. development tools.
- The IVI Foundation effort will improve the uniformity of drivers. (this is necessary as a T&M collaborative effort since computer standards do not address instrument specific issues.

IEEE-1394 Firewire



MiniDV with i.LINK

Several new computer standard I/O interfaces are now available which could eliminate the need for the GPIB Test &Measurement specific I/O interface.

USB (Universal Serial Bus) and IEEE-1394 ("Firewire") are leading candidates.

- USB comes already installed and supported by the operating system in PC's today. It offers inexpensive cabling, reduced cable length restrictions, easier to route cables. You can purchase cables from any computer store.
- Expect IEEE-1394 to become standard on PC's within a year.
- Future enhancements ongoing for both of these interfaces.

We see growing momentum in the computer industry for Firewire. Compaq and Sony now provide a Firewire I/O port with their PC's. Home digital video is driving this. [Note: Although the current version of IEEE-1394 used with video cameras is not compatible for instrumentation, IEEE-1394.2 is well suited for test and measurement applications] Eventually, we expect customers will demand Firewire over GPIB as the main connection between instruments and computers.

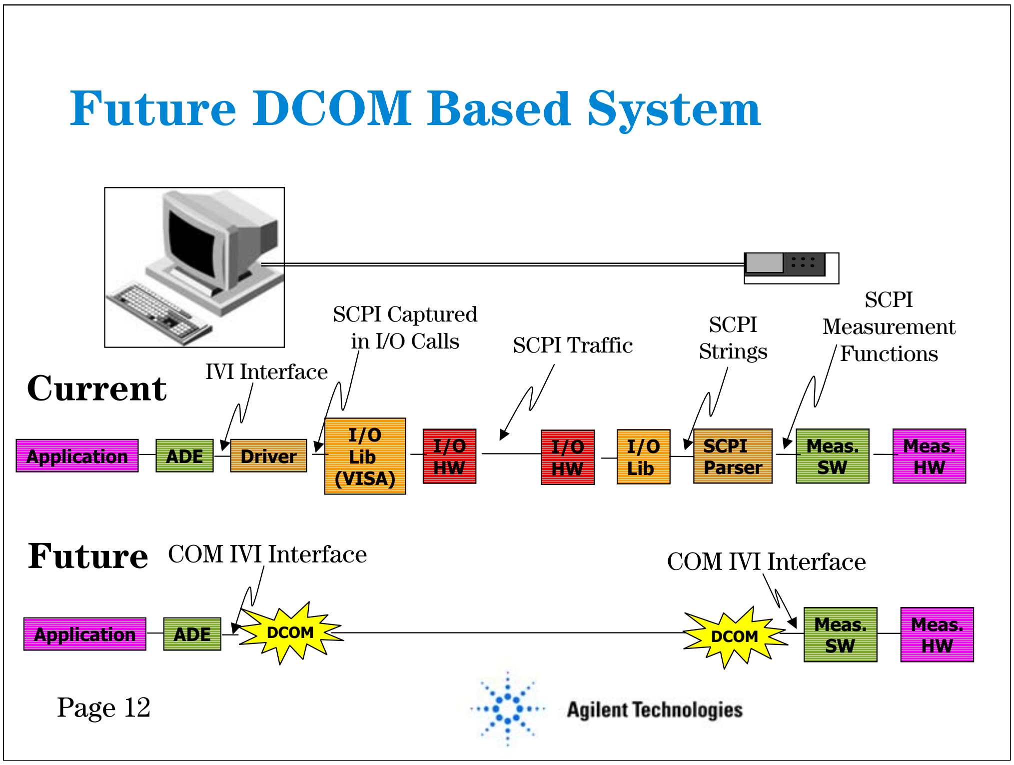
COM* -- Software Interface Specification.

- Eliminates the need for :
  - ASCII interfaces,
  - Proprietary I/O libraries.
- Provides natural solutions for
  - driver deployment/integration,
  - remote access (DCOM)
- Software component object reuse is now possible!

* COM = Component Object Model

Agilent Technologies

- Microsoft COM (Component Object Model) is being used to transform instrument driver technology. This standard interface includes off-the-shelf development tools and is consistent with the move to "Commercial Off-the-Shelf" (COTS) practices.
- This will eliminate the need for ASCII interfaces to control instruments, and the need for proprietary I/O libraries.
- COM also provides natural solutions for driver deployment/integration, remote access (DCOM, or Distributed COM), and a variety of other benefits.
- Software component object (e.g. driver) reuse is now possible!



COM is the correct technology for the future

It solves the VXI *plug&play* "name pollution" problem. (Names do not have to be unique.)

Facilitates software reuse

Easy Multi Platform (ADE) interoperability

> MSVC++, Java, Visual Basic, LabVIEW, LabWindows, Agilent VEE, PC, UNIX, Linux all supported

Makes it possible to build new instruments where the API is "built in". (Support, Quality, Completeness)

IVI Foundation - Standards

- IVI Foundation is an open consortium of
  - End-user companies
  - System integrators
  - Instrument manufacturers
- Founded in August 1998
- 53 companies

Working together to Achieve Interchangeability

We have just spoken about two computer industry standardization efforts that will promote better connectivity and improved instrument interchangeability. There is also a test and measurement industry initiative which is the Interchangeable Virtual Instruments (IVI) Foundation. It was founded in August 1988 and currently has 53 companies (37 voting members and 16 associate members). These 53 companies are working together to achieve instrument interchangeability.

IVI Foundation Members

Test, Measurement & S/W

- Advantest
- Agilent Technologies
- Anritsu
- ASCOR
- GenRad
- Hamilton Software
- IFR
- Keithley Instruments
- LeCroy
- National Instruments
- PX Instrument Tech.
- Racal Instruments
- Rohde & Schwarz
- Tektronix
- Teradyne
- TYX
- Vektrex
- Wavetek

End Users/Sys. Integrators/Other

- AVL North America
- B Squared Technologies
- BAe Systems
- Boeing
- CACI-ASG
- CERP_AIGER Program
- Condor Engineering
- Data Science Automation
- Defence Logistics Organisation
- Ericsson Radio Systems
- Excalibur Systems
- Frontline Systems
- Lockheed-Martin Aeronautical Systems
- Lucent Technologies
- Northrop Grumman ESSS
- Raytheon TI Systems
- Rockwell Collins
- Serco Test Systems
- Serendipity Systems

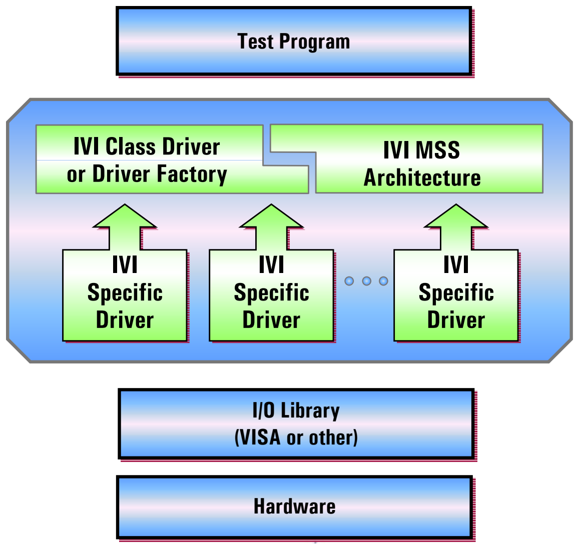
Agilent Technologies

So who or what is this foundation?

The foundation includes 37 voting members and 16 associate members. The 53 companies are comprised of:

- Test & measurement
- Interface card providers (GPIB, PXI, etc.)
- System integrators
- Users (self integrators)
- Software driver developers
- General purpose software tool providers

IVI Architecture



Agilent Technologies

The IVI Foundation is working on two things. First is standardizing programming interfaces for the common instrument types such as:

- Digital multi-meters (DMM)
- Counters
- Switches
- Function generators
- Power supplies
- Spectrum analyzers

Second, is the standardization of software architectures used for instrument Drivers as well as more complex test and measurement solutions referred to on this figure as IVI-MSS for Measurement and Stimulus Subsystem. We will talk about Instrument Drivers first then IVI-MSS.

Key IVI-Driver Benefits

- Standardized semantics for instrument control
  - Assists with interchangeability
- Common look and feel for application programmers
- COM, allows easy interoperability with all common application development environments

Agilent Technologies

The primary contribution of the IVI Foundation to date is "Standardizing the semantics" of instrument control. This has to do with agreeing on the spellings that will be used for the names of driver functions or methods. The Foundation has identified the common instrument types along with their common features and has standardized how these features will be invoked. Before IVI, application programmers had to work with drivers that had different interfaces even for common features found in instruments. This made it very difficult to achieve interchangeability. The IVI semantics are called IVI Classes.

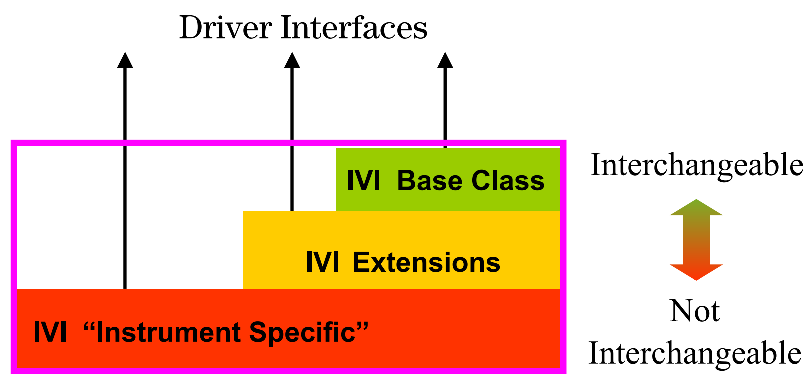
Digital multi-meters (DMM), Counters, Switches, Function generators

Power supplies, Spectrum analyzers

Another key benefit of the new driver standards is that Application programs will be able to apply their experience with one IVI-Driver to help them understand how to use other IVI-Drivers. This is because of the common look and feel IVI-Drivers provide to the programmer.

Finally, the availability of Microsoft COM as a de facto standard for SW component reuse, has made it possible for a single instrument driver implementation to work with all of the the major Test and Measurement application development environments such as VB, C++, LabView, LabWindows, VEE, etc.

IVI Drivers & Interchangeability



Agilent Technologies

The "IVI Base Class" interfaces provide instrument features that are common to a given class or type of instrument. These interfaces are required.

The "IVI Extension" interfaces provide features that are common to a given class or type of instrument but are not found in all. These interfaces are optional.

When the common functions are not sufficient, the application developer can access the "Instrument Specific" interface. The instrument specific interface exposes the full (and possibly unique) capability of that specific instrument. Obviously, when this is done, any SW that directly uses these interfaces will not be interchangeable using different instruments.

Limitations of IVI-Drivers

- Can not interchange an instrument of one class with that of another
- Syntactic interchangeability does not assure the same behavior
- No guarantee of identical measurement results
- No provision for aggregation
- No provision for the "Reuse" of measurement code

Having standardized, function or method names as well as a standardized interface technology is a big step forward and is very beneficial, however there are limitations.

An Oscilloscope that can perform Voltage measurements can't be used in place of the DVM if the mechanism for interchange is IVI-Drivers.

IVI Standardizes on the spellings of function names but does not fully define behaviors. This can lead to 2nd order effects that will result in different results after an interchange.

The simple IVI-Driver model doesn't deal with applications where two or more instruments are to be used together to perform some aggregated functionality.

There is value in "reusing" common measurements that are not implemented inside of instruments. The current IVI-Class specifications are limited to an instrument level abstraction.

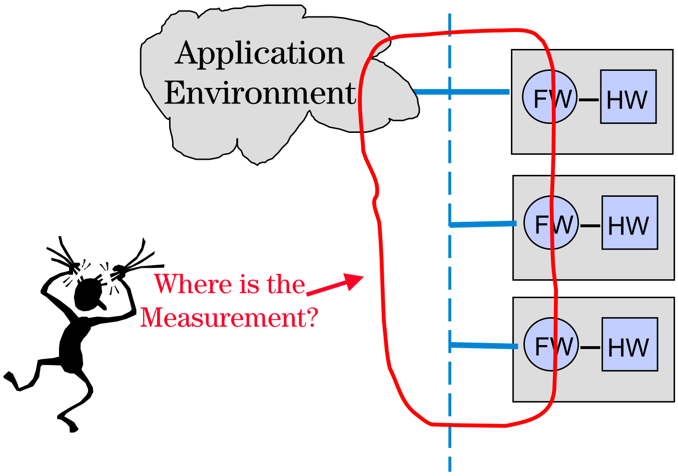
Architecture for Improved Interchangeability (IVI-MSS)

- Measurement and Stimulus Subsystems
- Resolves IVI-Driver limitations.
- An architecture, originally developed by HP, now being standardized in the IVI Foundation
- Can deliver "Same Answer" interchangeability.

Agilent Technologies

IVI-MSS stands for Measurement and Stimulus Subsystems. This is one of the standing working groups of the IVI Foundation. IVI-MSS is an architectural approach that resolves many of the limitations just mentioned for standard IVI drivers. This work was originally developed by Hewlett Packard and the proof of concept has been demonstrated in several products. *Note: Agilent Technologies is a new company comprised of the former Hewlett-Packard test and measurement, chemical analysis, semiconductor components, and medical products.* IVI-MSS does deliver same answer interchangeability. The IVI-MSS working group is now standardizing the original work as part of the overall IVI Foundation efforts.

Architecting for Interchangeability

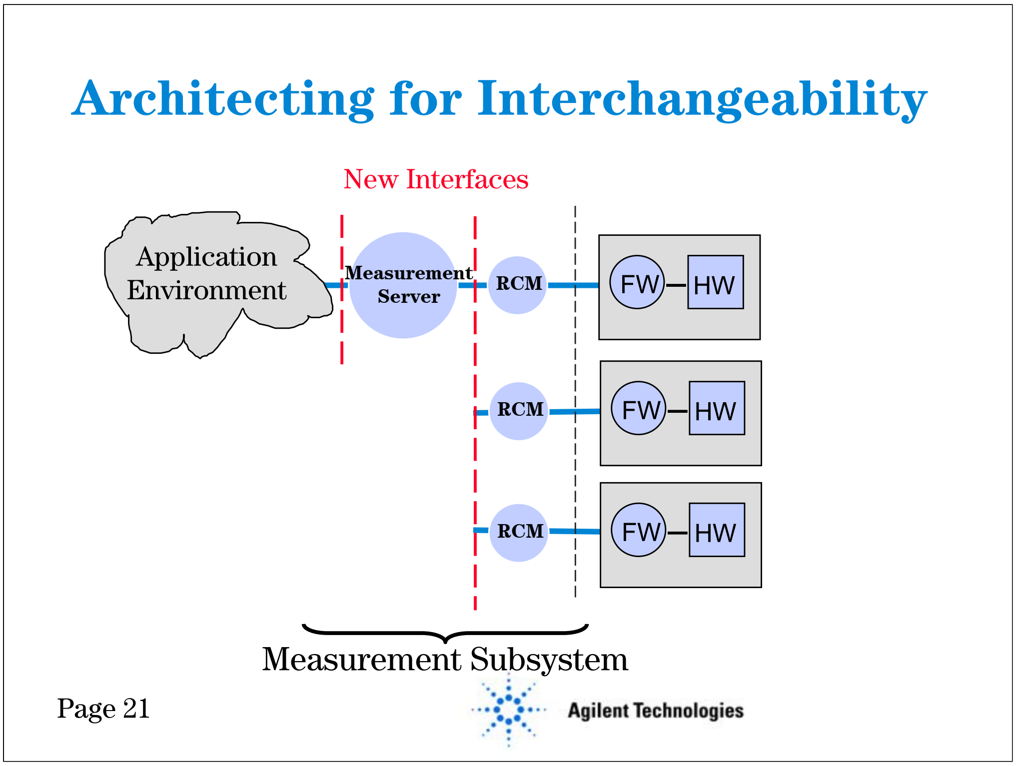


Agilent Technologies

Many application programs are written this way.

If you have ever experienced the problems of changing an instrument and having to rewrite your test software then you can relate to the frustration of the stick figure on the left. Often the instrument firmware and the measure program are so tightly integrated that a change in instrument (or sometimes even a firmware revision) can change the actual measurement result. When this occurs, the existence of the measurement discrepancy is almost immediately apparent, but the fix is often quite elusive.

In answering the question, where is the measurement?, it can be found partly in instrument firmware and partly in the end application program. The reason this type of system design does not deliver either interoperability or interchangeability is because the physical I/O interface cuts right through the middle of the measurement. Clearly, an architectural approach is needed.



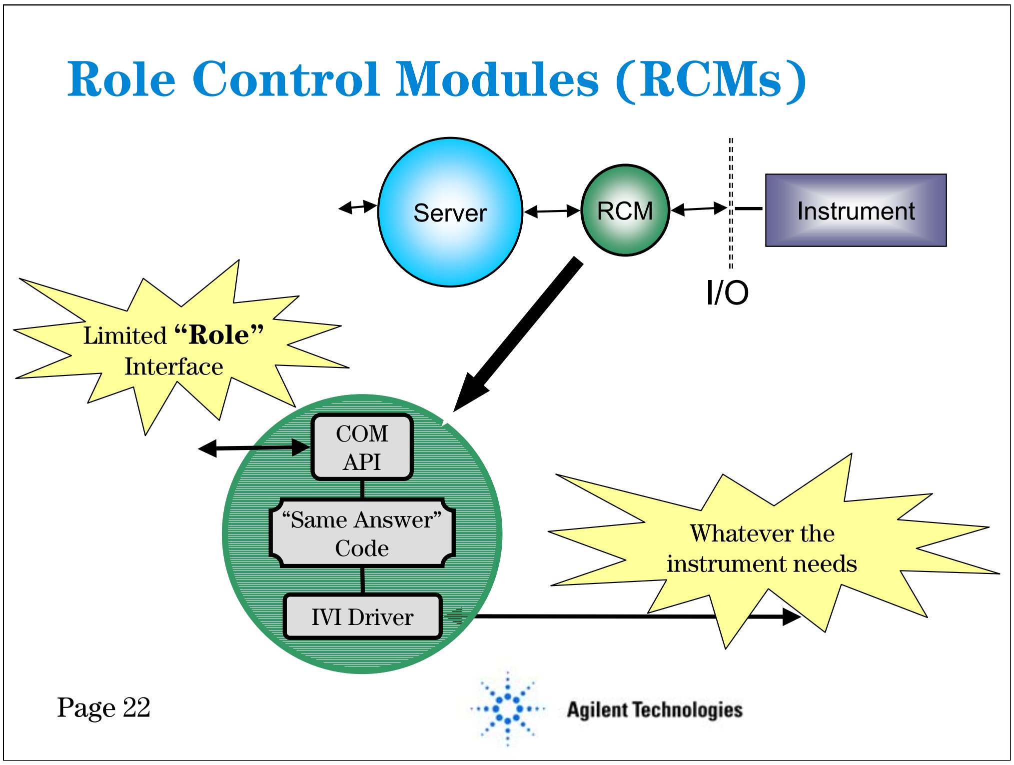
The solution to this problem just highlighted is provide two additional interfaces that separate the end application program from the test instrumentation. The two new elements added are:

- Measurement or Stimulus Servers

  These SW components encapsulate as much of the problem at hand as possible in a way that is truly test asset independent.

- RCM's for Role Control Modules

  In this picture a RCM may look like a Driver, but it has some very different responsibilities. Which will be reviewed next.



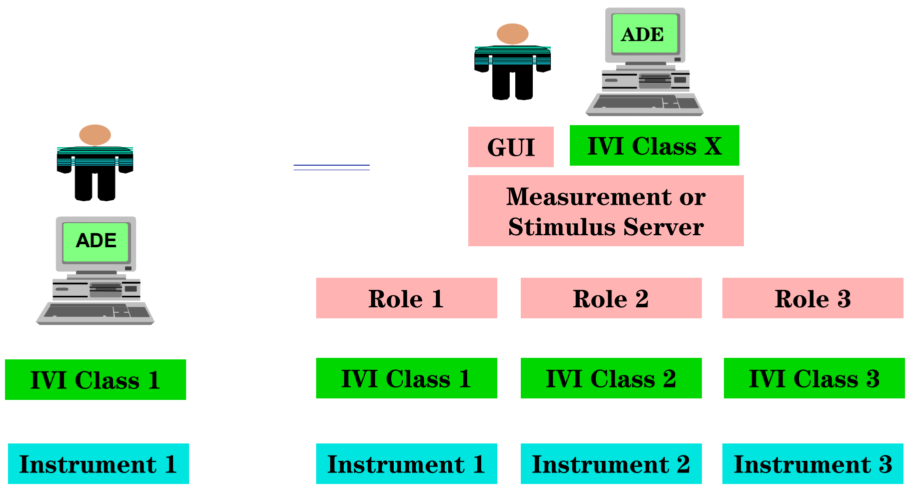
A **RCM** is required for each Asset used by a Measurement Server in a subsystem. The left side of the RCM in this picture shows interface to the Measurement Server. This interface presents only the features and capabilities required for a single "ROLE" in a subsystem. When necessary, an RCM contains the software that will make a lesser asset "measure up" to that of a more capable one. This code goes in the "Same Answer" block. Any measurement software that would have to be rewritten for a new asset must be put in the RCM and not the Measurement Server. RCMs are part of the value added for a measurement subsystem. A key point is that they are owned by the "Solution Provider" not the general purpose instrument manufacture. RCMs are the key element in providing asset interchangeability.

What is a ROLE: A "band pass filter" on the feature set of physical assets. An interface to a Measurement Server providing just what it needs and NOTHING else. A role has a rough correspondence to various types of instrumentation. A real Measurement Subsystem will probably require several roles.

A Role is an interface contract between a IVI-MSS Server and a particular hardware asset.

The RCM in this picture shows an IVI Driver being used. This is not required however. A RCM is able to use what ever is needed to communicate with its associated asset.

IVI Drivers vs. IVI-MSS



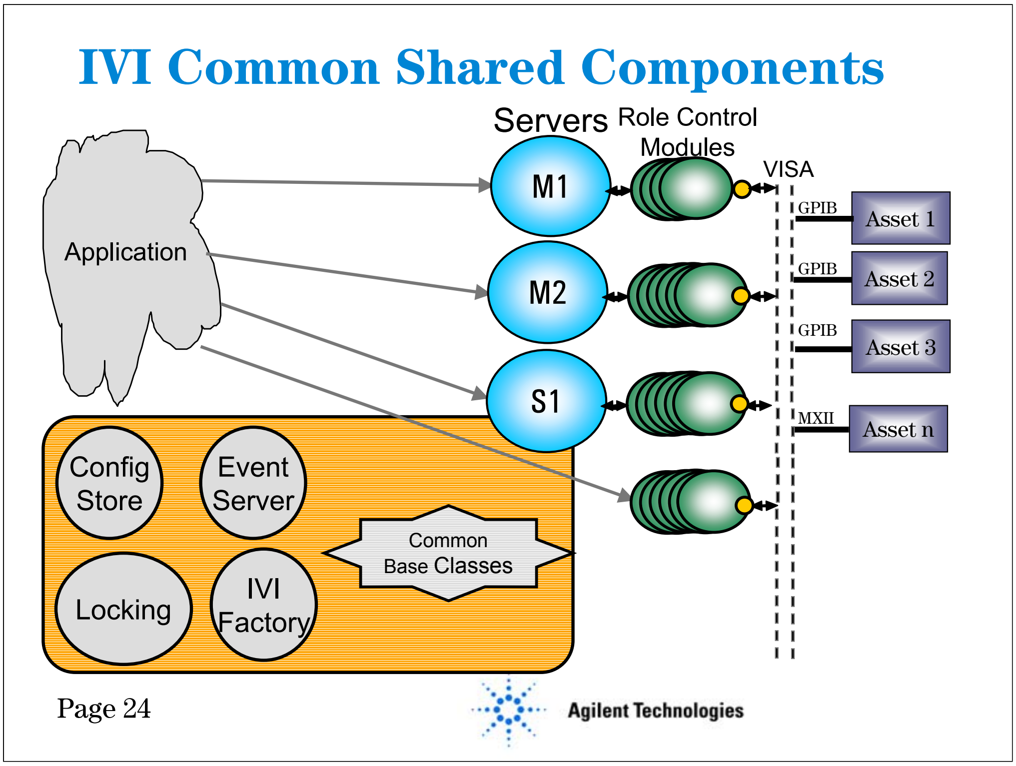
Agilent Technologies

Here is a simplified view of how a test system architecture would look in two examples.

On the left is the most common and simplest application of IVI technology. It shows a user interacting with an application that was developed using an Application Development Environment (ADE). The application uses the capabilities of a physical instrument through a IVI Class drivers. This is the base IVI approach.

On the right hand side you can see how IVI-MSS makes it possible to to create a Measurement and Stimulus Subsystem turn key solution. The key differences in are:

- The use of Role Control Modules to deliver "Same Answer" interchangeability
- The use of a Server to contain the reusable domain knowledge or measurements.
- The use of Aggregation to allow multiple instrument to be used together in a single solution.
- The provision of a GUI so the Solution can be used manually as well as automatically by a separate application program.



This figure shows how several IVI-MSS Solutions can be used together in a single ATS to provide a rich and powerful environment for TPS execution. M1 and M2 represent two different Measurement Subsystems built using IVI-MSS, S1 represents a Stimulus Subsystem. In addition to the Servers and Role Control modules that make up the IVI-MSS Solutions, there are a small set of common components that are needed. These are being developed in the IVI Foundation and will serve as systems glue.

Config Store

The Configuration Store is a COM Component and associated set of data the keeps track of the I/O addresses of assets in the system and also keeps track of which instruments are to be used for what purpose.

Event Server

The Event Server accepts callouts from the various components to collect asynchronous events. It time stamps and records this data and makes it available for a variety of uses.

IVI Factory

The IVI Factory is responsible for instantiating the various COM components in the system such as IVI Drivers, Role Control Modules and Measurement or Stimulus servers. It also provides handles to the clients of these components.

Locking

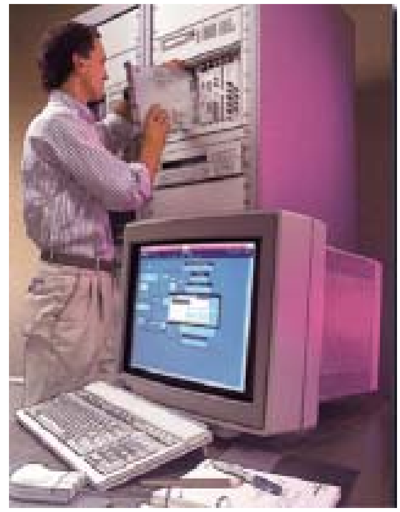
The Locking component keeps track of which instruments are being used and allows for sharing them or blocking access to them if they are busy.

Common Base Classes

Common Base classes for both Measurement Servers and Role Control Modules can make it easier to build Measurement Subsystems and will contribute to their overall quality and common “look and feel”

IVI-MSS Solutions

Permit upgrading a test system while retaining existing test software



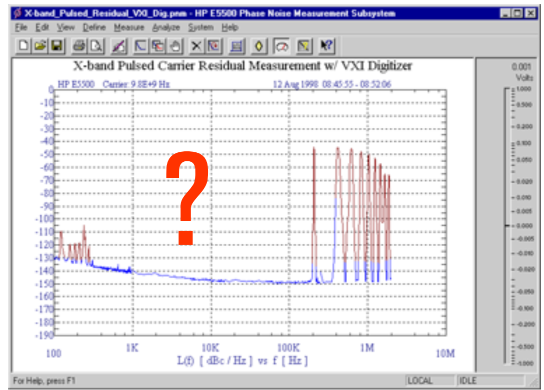
Page 25

Agilent Technologies

The following slides walk through a real example of instrument (asset) substitution. While the reason for the substitution is not important (it could be due to obsolescence, improved performance, cost, or other operating attributes to name a few examples), we will assume that issue is obsolescence.

Test-Asset Interchangeability

Obsolete Asset



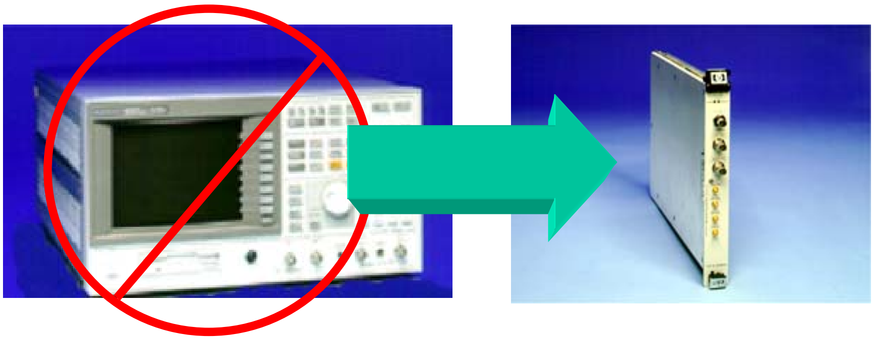
Agilent Technologies

This represents a Phase Noise Measurement solution where three instruments are used together to produce the results shown on the graph.

What if one of the instruments becomes obsolete? Will it be possible to recreate the data shown on this graph after moving to a new instrument? Can there be a guarantee that the results will still be valid?

Test-Asset Interchangeability

“Box” Instrument Replaced With VXI Digitizer



Agilent Technologies

The intent is to replace one of the “rack and stack” instruments with a VXI solution.

To make the switch, a Role Control Module must be switched along with the hardware asset.

The original Role Control Module used a “box” style instrument to digitize the test waveform, and perform the needed digital filtering and FFT.

The Role Control Module for the new digitizer provides the same data in the same form as required by the measurement component, but now the FFT and digital filtering are performed in the new Role Control Module.

The measurement component cannot detect the difference. The results are the same except that the overall measurement speed is faster with the digitizer.

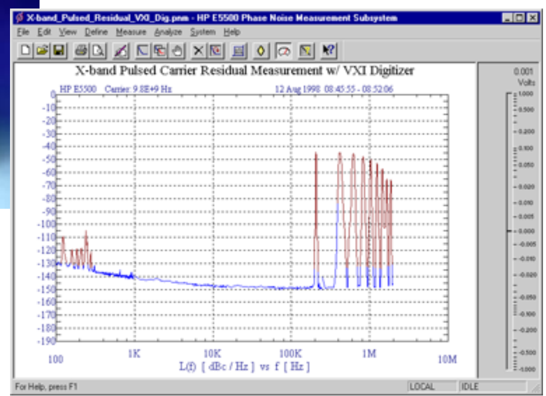
Because the RCM interface is tightly and narrowly defined from the perspective of the measurement, it is possible to verify that the new device performs as needed and the results can be guaranteed.

Note that the interchanged instruments are not of the same IVI Class.

Test-Asset Interchangeability

Different Equipment, Same Measurement Results

With No TPS Change!



Agilent Technologies

We now see a new stack of hardware on the left.

The software accommodates the new hardware and the measurement results can still be be guaranteed by the solution provider.

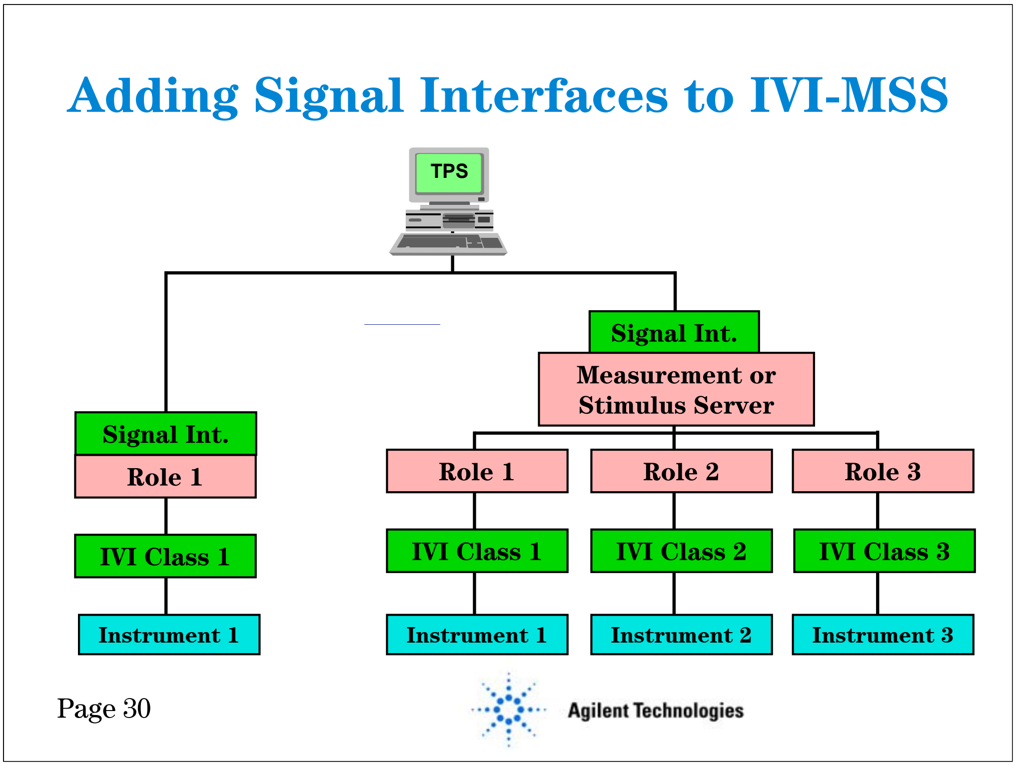
What users care most about is the results are the same as before.

No TPS Changes were required.

TPS = Test Program Set

Signal Interfaces, the Final Step

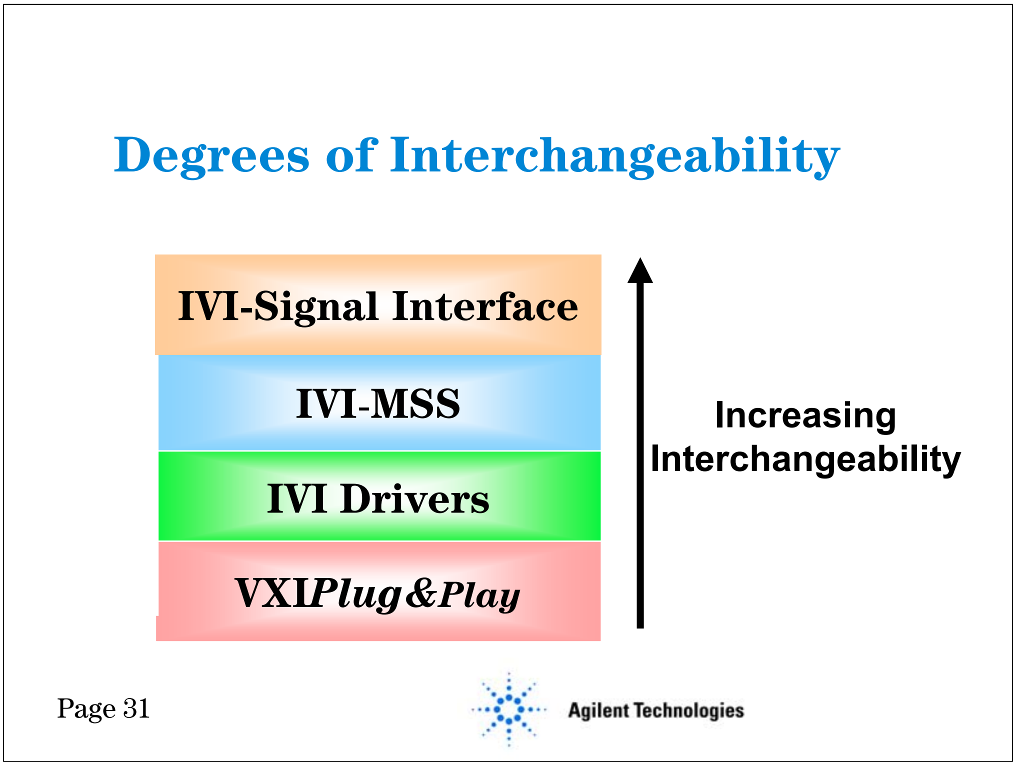
- Semantics that focus on signals applied to or taken from a DUT
- Ability to query the specifications and uncertainties of an automatic test system
- Ability to specify the physical connections to a DUT



In this architecture diagram you can see how A Signal Style interface can be applied on top of either a single instrument or a full IVI-MSS aggregated solution.

The signal interface represents the exposed semantics of either Role 1 shown on the left, or the Measurement or Stimulus Server as shown on the right.

When a TPS uses the test equipment through these Signal Interfaces it will be possible for automatic resource allocation to be performed as well as automatic determination of the ability of the test equipment to deliver the accuracies that are required by a given TPS.



As we have said there are varying degrees of interchangeability.

Make the assumption that an end user's application program is written three ways. 1st it is written to directly use VXIplug&play driver APIs, 2nd it is written to use IVI-Class Compliant instrument Drivers making the assumption that these are "off the shelf" commercial IVI-drivers, 3rd it is written to use an IVI-MSS based solution.

The VXIplug&play version of the application will not provide interchangeability. As soon as an instrument is interchanged and its associated driver is placed in the system; the application will no longer work. Function names will be different and there will be compiling and linking errors.

In the case of the IVI-Driver version of the application; the application should run and there will be no need to recompile or relink. One must make the assumption that no Instrument specific functions have been called. If there are no instrument peculiarities or "second order" effects the application should provide expected results.

In the case of the IVI-MSS version of the application; At the cost of additional complexity, a "Solution Provider" can guarantee that the "same answer" will be obtained after a "supported interchange". The interchanged asset can have instrument peculiarities or "second order" effects. The interchanged assets can even be of a different instrument type as long as it is physically capable of performing the physical stimulus or measurement that is required by the solution. There is no magic here. The benefit is achieved by doing customization work which someone has to do. IVI-MSS specifies how and where this work should be done. The validity of this has been established by work done at Agilent Technologies.

Finally, the Signal Interface permits an ATS to query for the actual specifications and capabilities of the measurement or stimulus hardware. This makes it possible to support automatic resource selection and allocation. In additional the signal interface layer offers a way to describe connects of signa s to D.U.T test ports.

It is critical to identify the owner of the various interfaces. Having the several layers makes it possible for various participants in the ATS business to have places to express their requirements or to deliver the value of their solutions.

E5500A Phase Noise Test Subsystem

Page 32

Agilent Technologies

This is an example of a Measurement subsystem. The example system happens to measure phase noise. Although it is not obvious from this photo, the system contains:

1. A PC computer controller
2. A test interface
3. An instrument to measure the downconverted phase noise (can be PC card digitizers, VXI digitizers, and various spectrum analyzers)
4. A RF/MW downconverter (optional, not shown)
5. Reference [sig-gen] sources (optional, not shown)
6. Measurement Sub-system S/W based on IVI-MSS.

Using the architecture that Roger has just explained Agilent can:

- Easily maintain multiple configurations
- Guarantee the same answer over multiple configurations
- Easily add new role control modules in response to customer requests or replace system components when they eventually become obsolete (Wasn't like this in the old system - you could have any digitizer/SA as long as it was the HP 3561A!)

Because IVI-MS is an open standard there is no reason why you can't expect the same benefits or ask for them from other instrument manufacturers . . . Or you can use this same open standard in your own development work and realize the same benefits.

E7501A Stimulus Signal Subsystem

Available Sept, 2000

Agilent Technologies

This is an example of a Stimulus Subsystem. It consists of :

1. E6432A 10 MHz-20GHz Microwave Synthesizer (3 C-size slots)
2. A Racal triple output ARB connected to the AM, FM, and Pulse external modulation inputs of the E6432A (1 C-size slot)
3. A 1 slot IEEE-1394 Firewire interface
4. Optional PC
5. Stimulus subsystem software containing role control modules for each of the system components and a user API that allows ATE developers to specify signal parameters directly

Most ATE systems generate signals by driving the external modulation inputs of signal generators with external sources. These days, many systems use Arbitrary Waveform Generators (ARB's) for max. flexibility. If you have done this yourself you are well aware of the hassle and confusion of compensating for a 50 ohm ARB output impedance driving a 600 Ohm input impedance where rated modulation (say FM peak deviation) occurs when a 1 volt peak signal is applied). The Signal Studio software frees you from having to worry about any of these considerations. Instead, you can focus on defining the actual signals that you want the stimulus subsystem to generate.

If there is ever a reason to change one of the assets, for example for a new ARB), Agilent can do this for you easily (and guarantee the signals you use won't change) because the subsystem uses the IVI-MSS architecture.

Conclusions

- Computer Standard I/O will eventually replace GPIB
- COM drivers work with all major ADEs
- Many companies will standardize on IVI-COM drivers
- IVI-Class Drivers provide drivers with standard semantics
- IVI-MSS provides "same answer" Instrument Interchangeability

Agilent Technologies

- We've examined some problems that many ATE systems have
- Surveyed some of the exciting new developments in computer I/O interfaces, new component object standards, and S/W driver standards..
- We showed how these new interfaces and S/W technology can solve these traditional problems.
- We showed how different solution are required depending on the measurement complexity.

For more information on IVI, the Measurement Subsystems Architecture, etc . . .

- **IVI Foundation Home Page**
  http://www.ivifoundation.org/
- **IVI-MSS Working Group**
  http://www.ivifoundation.org/groups/MSS/Default.htm
- **Papers on IVI-MSS**
  http://www.ivifoundation.org/groups/MSS/MSS-History.htm
- **Agilent IVI-related information**
  http://www.agilent.com/find/IVI
- **Phase Noise Solutions**
  http://www.tm.agilent.com/tmo/Products/English/PhaseNoiseMeasuringSystems.html
- **Agilent E6432A Microwave Source**
  http://www.tm.agilent.com/tmo/datasheets/English/HPE6432A.html
- **Sony i.LINK (IEEE-1394) information**
  http://www.sel.sony.com/SEL/consumer/handycam/worryfree/digital8.html#iLINK

Agilent Technologies

Here are some URL's to explore additional information. We have included the home page of the IVI-Foundation, the IVI-MSS working group, and some additional perspective on IVI activities from Agilent.

There is product information for the examples that use the IVI-MSS architecture. Please note that information on the E7501A Stimulus Subsystem will be available in Sept.,2000. The Microwave signal generator, the E6432A, is already available and the link to it is provided.

Finally, if you interested in following how IEEE-1394 is being used with camcorders in home video editing applications we provided a link to some information from Sony.