Choosing the Best Software Tools to Design Automated Test Systems
Software-Defined, Modular Instruments

Flexible Programming Interface and APIs

PC-Based Measurements

Powerful and Intuitive Software

Multicore Processing
RF Instruments
FPGA and Coprocessing
Precision DC
Switching
Mixed Signal
A Sample Modular Software Architecture

Test Management Software

Application Programming Languages / Scripting Languages
- Test Code
- Test Code
- Test Code
- Test Code

Drivers and Measurement Services
- GPIB/Serial and VXI
- Data Acquisition & Signal Conditioning
- Modular Instrumentation
- PXI/CompactPCI
- Motion
- Vision
- Distributed I/O
- PLC

Database and Data Analytics
Importance of a Modular Approach

- Characteristics of modular components
  - Have a distinct purpose
  - Are decoupled from each other
- Benefits of modular components
  - Easier to modify and maintain
  - Ability to reuse components between projects
- Modular test system benefits
  - Flexibility to adapt to new technologies as required
  - Scalability to add new functionality (and instrumentation) as needed
  - Reduced risk of hardware obsolescence through hardware abstraction layers
1. Drivers and Measurement Services

*Keep drivers in mind when selecting your hardware.*

- Device driver selection
  - Choices guided by
    - Hardware device
    - Programming language
1. Drivers and Measurement Services

A good driver is not just a DLL with a help file!

• Features of a good driver
  • Integration with development environment
  • Thorough documentation
  • Example programs
  • Robust design with detailed error reporting
  • Simulated hardware
  • Both high-level API and low-level control
  • Multithreading support
Example of National Instruments Drivers
Demo time
(NI driver layer)
Instrument Driver Network

- Industry’s largest source of instrument drivers
  - 10,000+ instrument drivers
  - 360+ manufacturers
- Access free, NI-certified instrument driver downloads
  - NI LabVIEW
  - NI LabWindows™ /CVI
  - Microsoft Visual Studio .NET

"We typically check the NI Instrument Network for instrument driver availability prior to selecting equipment for purchase because we are well aware of the amount of development time lost when creating drivers ourselves," Ernest Clifford - Alpha Research and Technology
1. Drivers and Measurement Services

**Measurement services software provides additional tools.**

- Measurement services
  - Additional software tools that feature hardware management capabilities
    - Hardware listing
    - Configuration tools
    - Self-test/diagnostics
    - Test panels

---

Example: NI Measurement & Automation Explorer (MAX) with an NI-Scope Soft Front Panel
2. Test Code (Programming Language)

You have many, many options to choose from.
2. Test Code (Programming Language)

You have many, many options to choose from.

- How to pick a programming language
  - Determining factors
    - Proficiency
      - Do my developers know the language?
      - Can they get trained on it?
      - Is there external expertise?
    - “The right tool for the job”
      - Is the language efficient for the task?
    - Driver support
2. Test Code (Programming Language)

*C and C++ are the most versatile, but they can be complex for most people.*

**C/C++**

- **Benefits**
  - Most software engineers have experience with it
  - Standardized versions are good for regulatory certifications (Example: ANSI C)

- **Considerations**
  - Low level—plenty of room to “shoot yourself in the foot”
  - Difficult for nonprogrammers to pick up
  - User interface development can be challenging
  - Multithreading and synchronization is complex
  - Not tailored for test and measurement
2. Test Code (Programming Language)

.NET languages are easier than C/C++, but they are still not optimized for test applications.

.NET (IDE: Visual Studio)

- **Benefits**
  - Many software engineers have experience with it
  - Higher level than C—easier to pick up
  - User interface development is much easier
  - Web development tools

- **Considerations**
  - Still difficult for nonprogrammers to pick up
  - Lack of engineering/scientific UI components
  - Multithreading and synchronization is complex
  - Not tailored for test and measurement
2. Test Code (Programming Language)

LabWindows/CVI, Measurement Studio provide optimized tools for C/C++/.NET test developers.

NI LabWindows™/CVI
(a development environment for ANSI C)

NI Measurement Studio
(a Visual Studio add-on for .NET)

Optimized for Engineers and Scientists
✓ Engineering/scientific UI controls
✓ Analysis libraries
✓ Integrated driver support

The mark LabWindows is used under a license from Microsoft Corporation. Windows is a registered trademark of Microsoft Corporation in the United States and other countries.
2. Test Code (Programming Language)

LabVIEW is an easy-to-use graphical programming environment for T&M applications.

NI LabVIEW

• Benefits
  o Graphical programming is very easy to pick up
  o Optimized for test and measurement
    – Plenty of engineering-specific UI components
    – Analysis libraries
    – Tight integration with hardware
  o Multithreading is automatic

• Considerations
  o Most trained software engineers have backgrounds in text-based coding
    – LabVIEW training and certification programs are available
Demo time
(DMM as TestStand Adapter Module)
Build Virtually any System Faster and With More Confidence Using LabVIEW and NI PXI

Testing based on your needs
The software resides on the host PC or a user-programmable FPGA, allowing user-defined measurements and analysis in real time.

Industry’s highest bandwidth and lowest latency
NI PXI and LabVIEW brings commercial off-the-shelf technology, such as PC buses, multicore processors, and FPGAs into your test system, giving you access to all the latest tools to achieve shorter test times.

Future proof your investment
Modular instrumentation gives you the flexibility to purchase what you need today and scale your test system in the future.

Easily connect to any hardware
LabVIEW integrates with a wide variety of instruments, ranging from traditional boxes to software-defined PXI modular instruments, allowing nearly any measurement to be acquired.

Save time developing test systems
The LabVIEW graphical programming language is intuitive in nature, allowing you to spend less time addressing text-based syntax and more time solving complex test system challenges.

Quickly analyze and present your data
With NI PXI and LabVIEW, choose from an extensive portfolio of modular instruments paired with more than 850 built-in signal processing, analysis, and mathematics functions to quickly acquire and understand your data.

ni.com
3. Test Management Software

Test management software has two roles: sequencing and managing tests.

Role of Test Management Software

- Test Executive (Sequencer)
  - Sequence tests
  - Execute and debug tests
- Test Management Framework (Operations repeated for each device tested.)
  - Operator interfaces
  - User management
  - Unit tracking
  - Results processing (reporting/logging)
  - Test flow control
3. Test Management Software

_options: Build or Buy_

**Build a monolithic test application.**

+ Less architecture needed
- No modularity benefits (extremely coupled)
- Extremely high cost of ownership
- Must create custom training

**Build test management software in house.**

+ Self-owned
+ No unit cost
- Long up-front development time
- Very high cost of ownership
- Must create custom training

**Buy COTS test management software.**

+ Ready to run
+ Low cost of ownership
+ Industry proven
+ Availability of developers
- Do not own the code
- Up-front software cost

ni.com
3. Test Management Software

Up-Front Development Time of Building Versus Buying

- Development Environment
- Custom Operator Interface
- Sequence Execution Engine
- Database Logging
- Report Generation
- User Management
- Parallel Testing
- Unit (Serial Number) Tracking
- Interface to Coding Languages
- Deployment Tool
- Localization
- Documentation

Development Time (in Man Days)
3. Test Management Software

NI TestStand: Ready-to-Run yet Completely Customizable COTS Test Management Software

- Graphical sequence editor environment
- Ability to automate tests written in any language
- Multithreaded sequence execution
- ASCII, HTML/Web, XML, and ATML report generation
- Access, Oracle, SQL Server database connectivity
- Ability to completely customize
The NI TestStand Ecosystem

- 1000+ companies worldwide
- 10000+ existing developers
- 20+ partner products
- 1200+ new developers trained annually
- 40+ worldwide branches providing support
- 10+ programming languages supported
- 100+ consultants and integrators
- 2 levels of development certification
Motorola Uses NI TestStand To Test Cellular Base Station

The Challenge
• Developing a single-test software application capable of testing a variety of Motorola cellular base station products.

The Solution
• A test specification and procedures database was created in Oracle
• A database-driven test application was created using NI TestStand, LabVIEW, and the LabWindows/CVI SQL toolkit

“...the annual projected new product test development costs fell from roughly $200K to $25K and development and maintenance savings combined were $475K a year.”
4. Data Logging/Analytics

Choose from a variety of industry-standard database providers.

- Databases are a common method of storing/archiving test results
- Relational databases are the most common
  - Test data fits the relational model very well
  - Have to set up the database schema
- Common databases
  - Microsoft SQL Server
  - Oracle
  - MySQL
- Varying costs, scalability, and ease of use
- NI TestStand integrates with all of these and any ODBC/OLE DB database
4. Data Logging/Analytics

Data analytics tools can help you make sense of your data.

- Use data analytics tools to make sense of collected test data and improve manufacturing efficiency
- Can build or buy
  - COTS tools include Proligent (Averna), Maintainable Test (Maintainable), WATS (Virinco), IntraStage, and so on (more at ni.com/teststand/partners)
  - Can also use offline data analysis tools like NI DIAdem
  - Many commercial tools offer web-based UIs to view the data
5. Requirements Management

Use requirements to verify the quality and completeness of your application.

- Successfully deal with complexity
- Comply with company, government, and industry standards
- Track progress of development projects as a function of requirements implementation or coverage
- Streamline development efforts
- Gain certainty and confidence
5. Requirements Management

NI Requirements Gateway provides traceability into requirements and their implementation.
Case Study: Thales Communication

Company
Thales, an $18 billion company, is a global technology leader for the defense and security, aerospace, and transportation markets.

Challenge
Manual configuration and execution of verification tests created a bottleneck in the Thales Communications product development cycle.

Solution
Thales Communications implemented an automated verification test framework, based on NI LabVIEW software and PXI hardware, to automate and reuse test code.
Managing Deployed Test System Software

**User Code**
- User built VIs, EXEs, DLLs, assemblies, etc

**Application Software**
- LabVIEW, TestStand, CVI, etc
- Associated Run-time engines for above SW

**Drivers**
- NI-DAQmx, NI-SCOPE, NI-VISA, etc
- Associated Run-time engines for above drivers
Test Software Deployment Strategies

- Manual copying of files and driver installers
  - Easy for small deployments
  - Not scalable
- Disk imaging
  - Easy restore state
  - Difficult to upgrade
- Building installers
  - User can optionally install specific components
  - Versioning simplifies deployment management
TestStand Deployment Utility

- Builds a single installer with:
  - Sequence files
  - Configuration files
  - Source code/DLLs/EXEs
  - NI HW drivers and NI SW RTEs
- Automatically includes static dependencies
- Can run 3rd party installers using custom commands
Installer Deployment – Single Installer Approach

Development System

Test Code

TestStand Cfg

Custom Result Processing

TestStand Engine Run-time Engines Drivers

User Interfaces

Single Large Installer

Deployment System

1.0

Deployment System

1.1

Deployment System

1.2
Installer Deployment – Multiple Installer Approach

Development System

User Interfaces

Test Code

TestStand Cfg

Custom Result Processing

TestStand Engine Run-time Engines Drivers

Small code-only Installer

Deployment System

1.0

Deployment System

1.1

Deployment System

1.2
Installer Deployment – Patch Installer Approach

Development System

User Interfaces

Test Code

TestStand Cfg

Custom Result Processing

TestStand Engine Run-time Engines Drivers

Deployment System

Deployment System

Deployment System

Small patch Installers

1.0

1.1

1.2

ni.com
New in TestStand 2013: Patchable Deployments

- Automatically includes only new or changed files
- Only 6 MB of patch overhead
- Provides patching for both user applications and NI software
- Allows for multiple cumulative patches on a single deployment
**Enhanced Sequence Editor**

- Usability features requested on TestStand Idea Exchange
  - Multiple arguments in a single Case statement
  - Add flow control steps around selected steps
  - Sequence Editor tab closing shortcuts

---

**select case should support multiple values**  Status: In Beta

by warren_scott on 07-16-2012 05:45 PM

For some reason I was certain this was already in the idea exchange, but I couldn't find it -- so I'll post it.

Select Case structures are frustratingly difficult to use if you want one case to support multiple values.

In text languages you can often do something like
Demo Time
Architecture of demo

Customised Operator Interface

Test Stand

Test & Measurement Automation Modules

LabVIEW

PXI

Switch  DMM  DAQ  Digitiser  Vision  CompactRIO
Customised Operator Interface
Automatically result collection and report generation

IEEE ATML

Customised
Expandable System Architecture

Data Server

Production Management

LAN (TCP/IP)

Production Test

QA/Repair

QC

ni.com
Evaluate NI TestStand for Free Today

- ni.com/ideas
- ni.com/teststand/whatsnew

...or visit ni.com/teststand