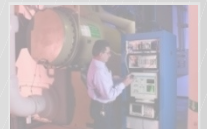
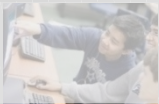
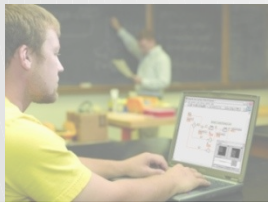
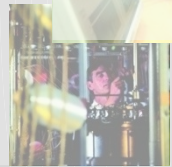


NATIONAL INSTRUMENTS

LabVIEW™



Click to edit Master subtitle style

LabVIEW 2010

What's New

Zileriu Vlad
Sales Manager Romania
National Instruments

What's New in **LabVIEW 2010**?

Improved Performance

New Data Acquisition Options

Environment Enhancements

Large Application Development

Target-to-Host Data Transfer

What's New in the LabVIEW Modules

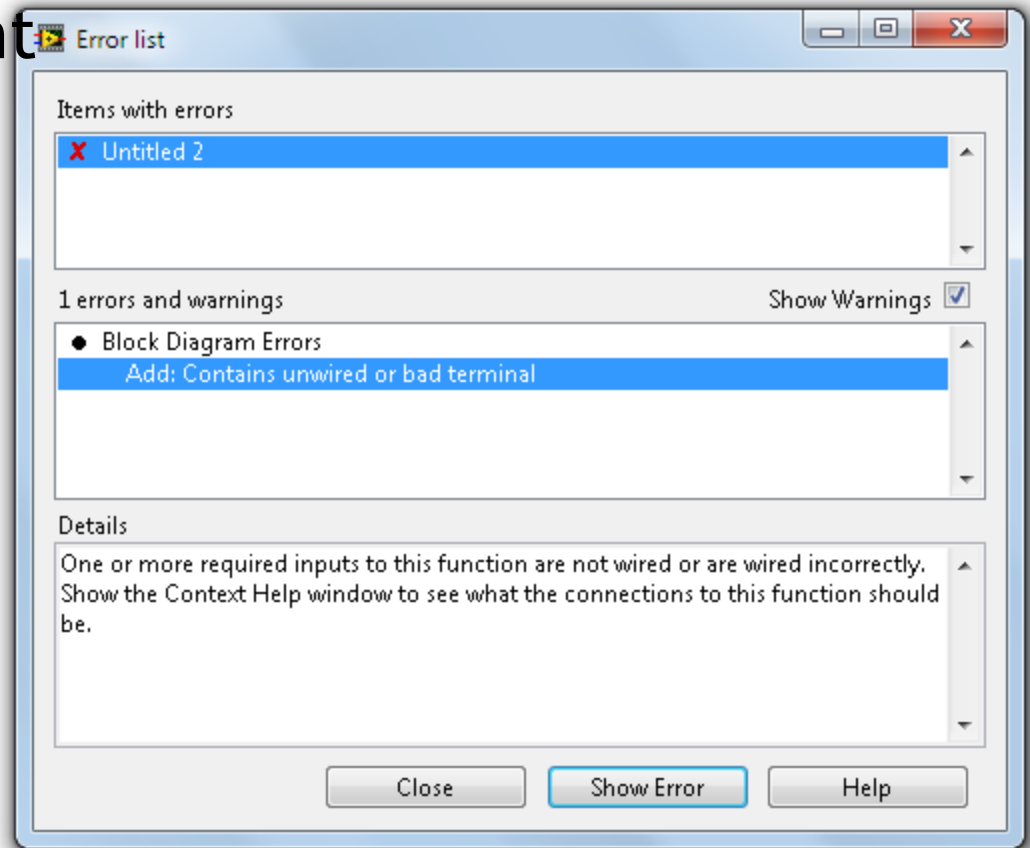
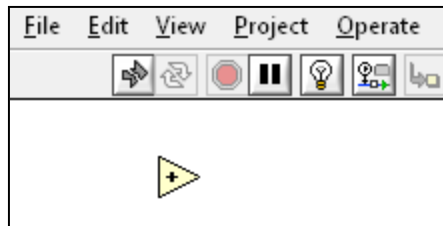
Building LabVIEW Add-ons

Improved Performance

LabVIEW Compiler

Abstracts the complexities of programming

- Memory management
- Thread allocation
- Language syntax



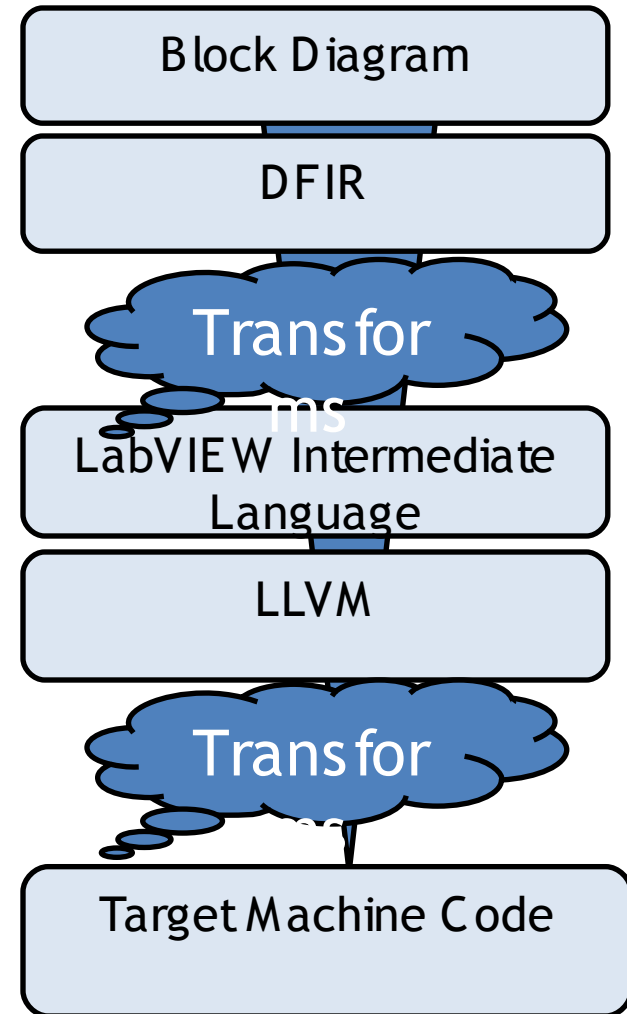
Optimizing the LabVIEW Compiler

DataFlow Intermediate Representation (DFIR)

- High-level representation
- Graph-based
- Preserves dataflow, parallelism, and execution semantics

Low-Level Virtual Machine (LLVM)

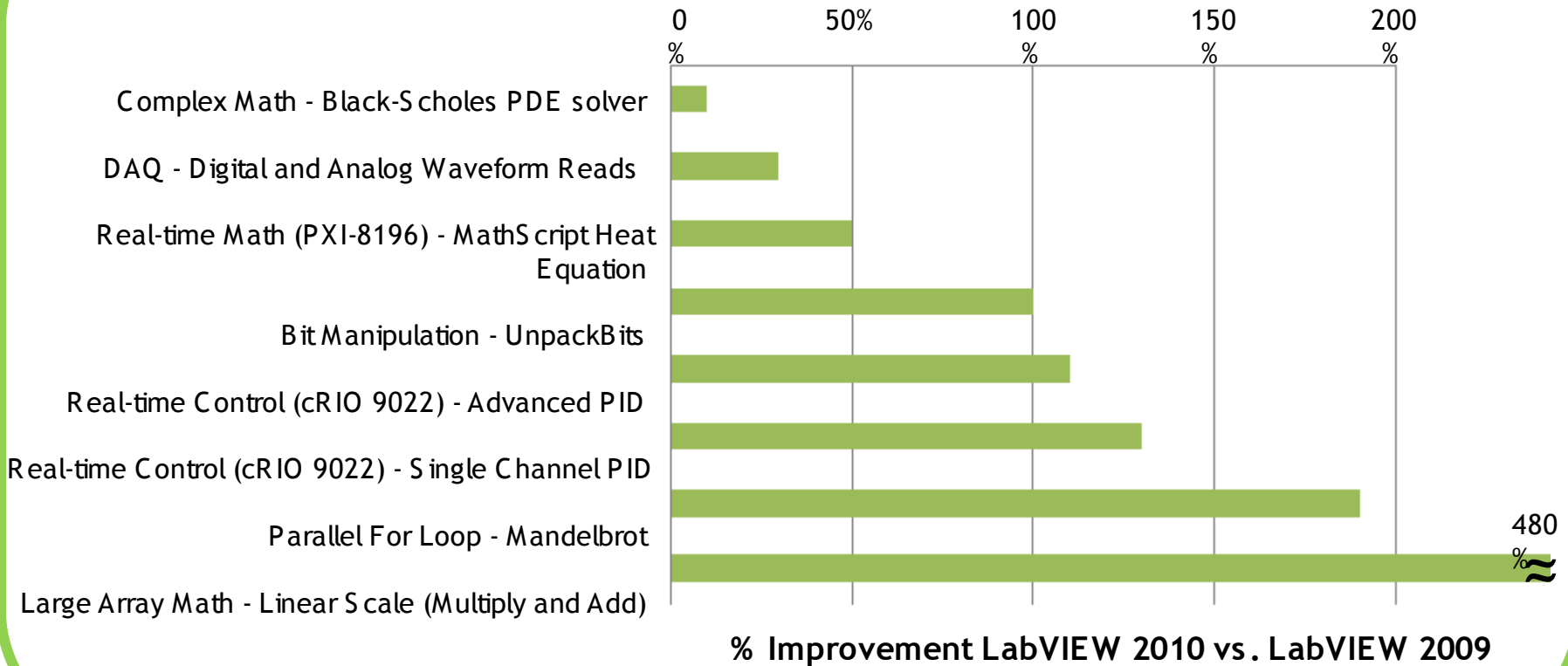
- Low-level representation
- Sequential
- Knowledge of target machine characteristics, instruction sets, alignment, etc.



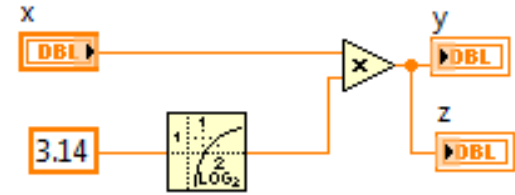
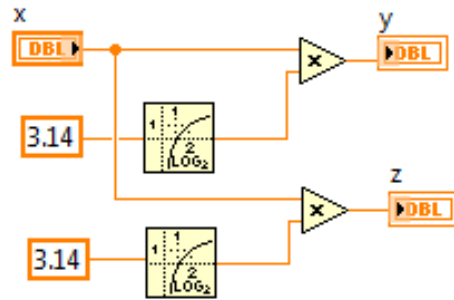
LabVIEW 2010 Performance Metrics

Dem
o

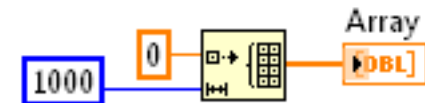
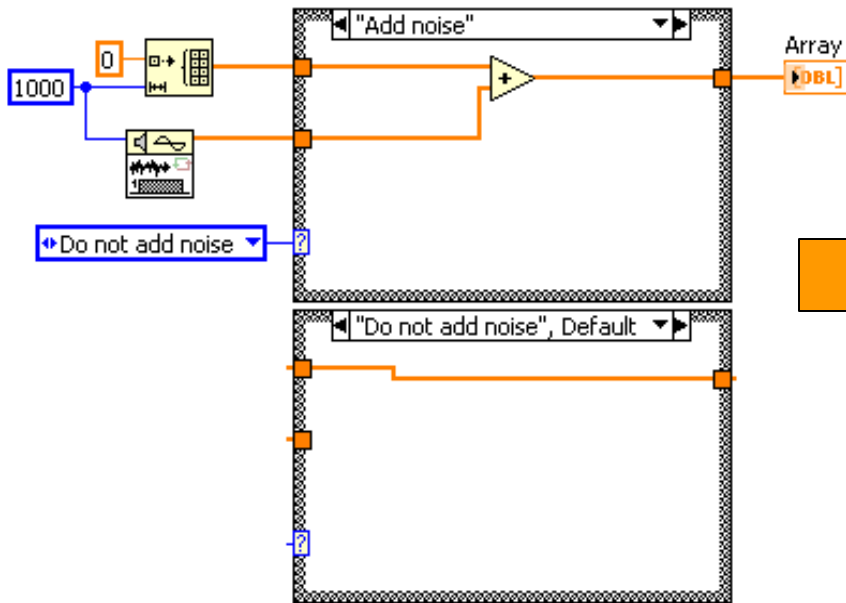
Run-Time Performance Improvement in LabVIEW 2010



LabVIEW Compiler Transforms

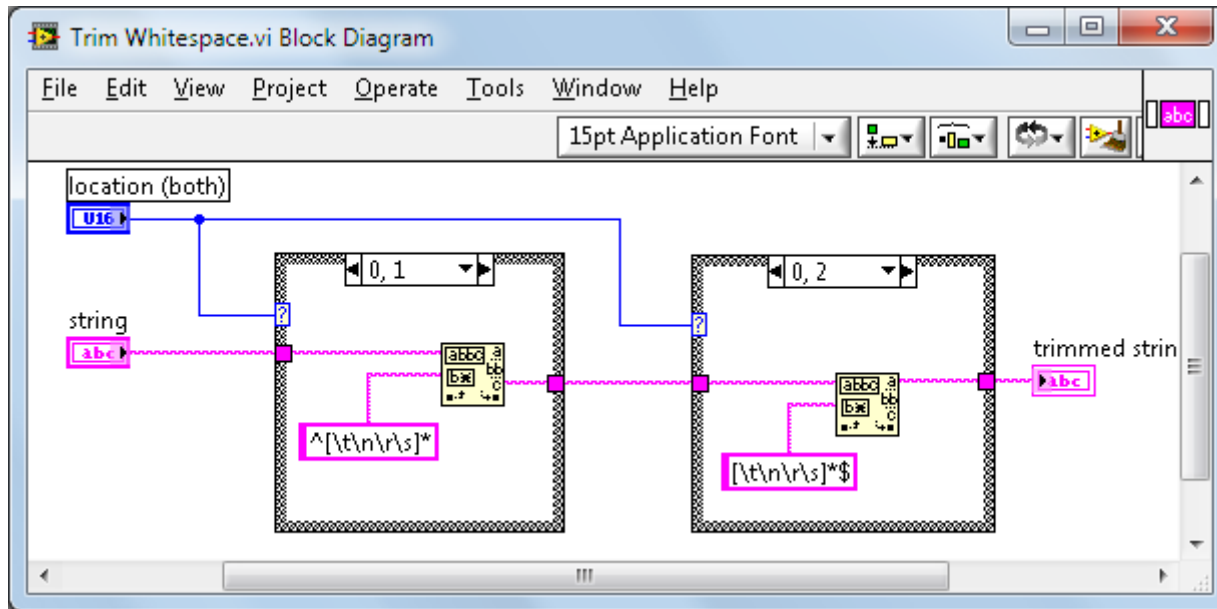
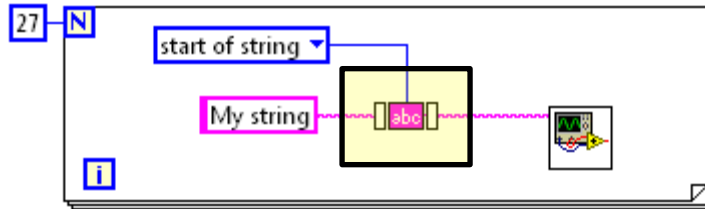


Common Subexpression Elimination

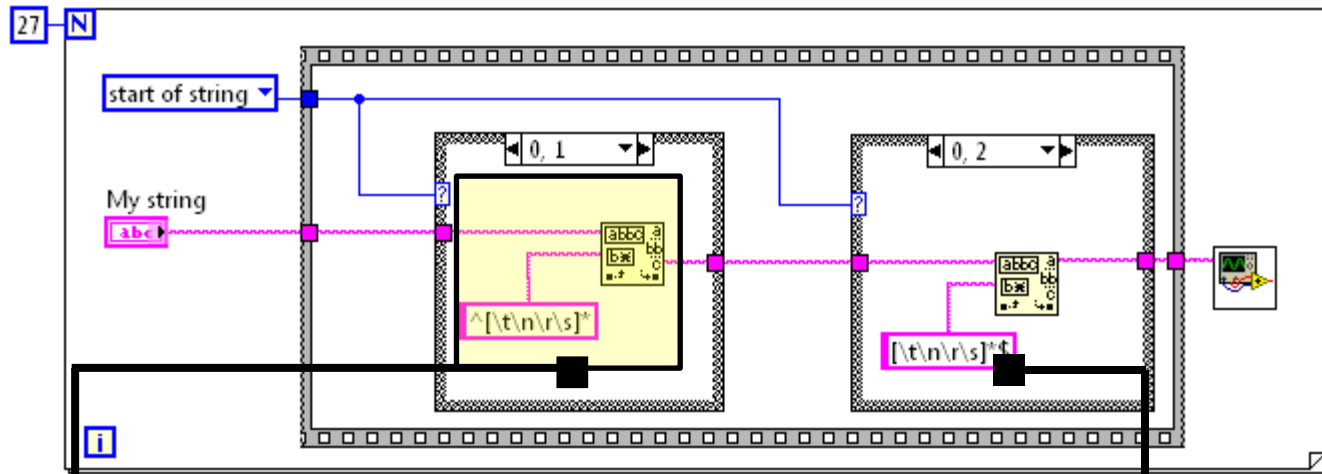


Unreachable Code Elimination

LabVIEW Compiler Optimizes Your Code



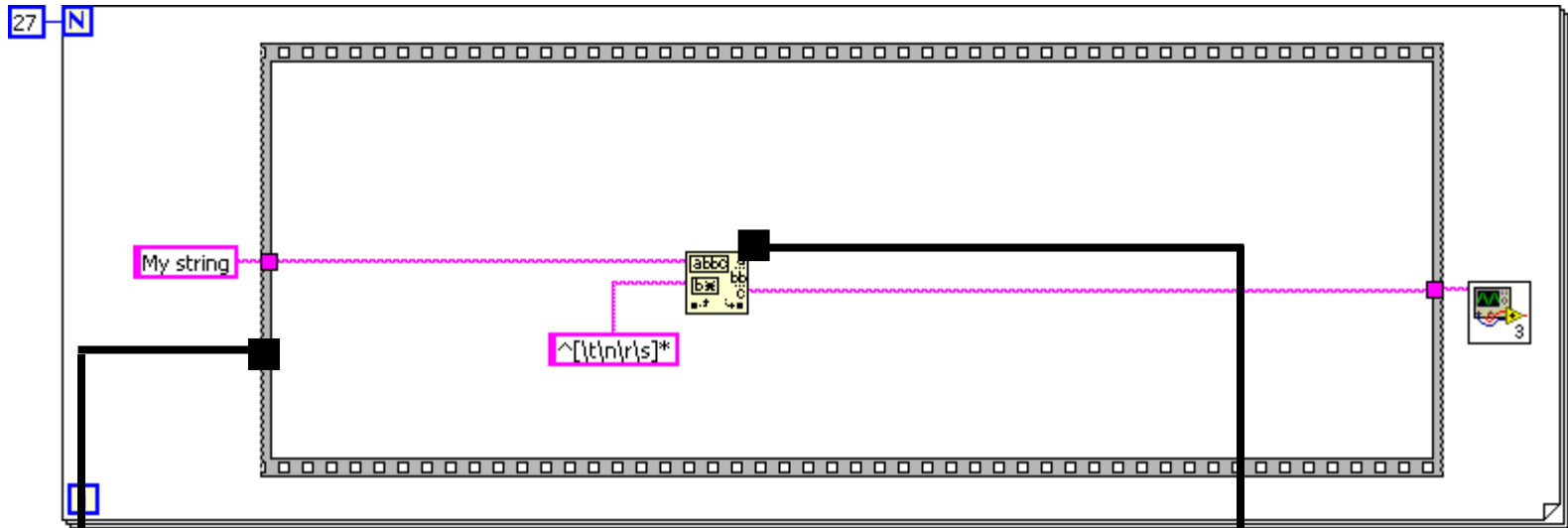
LabVIEW Compiler Optimizes Your Code



Only this portion of the code will execute

Because the input is constant, the compiler can determine which code will execute, and remove the unnecessary code

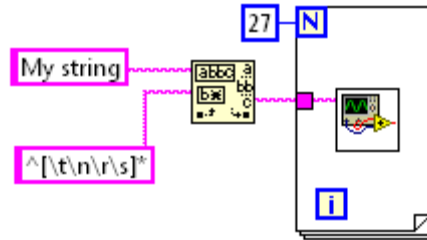
LabVIEW Compiler Optimizes Your Code



Sequence Structure is now unnecessary

The Match Pattern primitive will not change from iteration to iteration

LabVIEW Compiler Optimizes Your Code



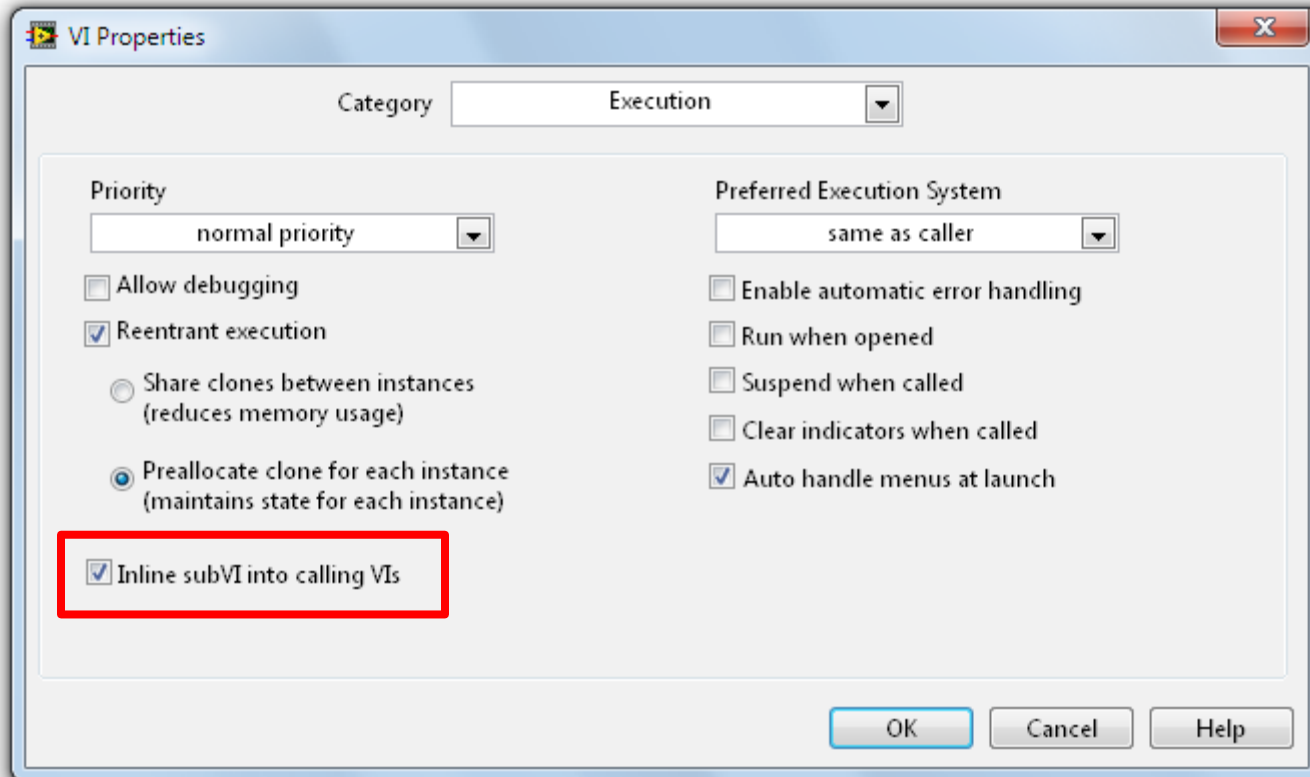
Transforms Used

- SubVI Inlining
- Unreachable Code Elimination
- Dead Code Elimination
- Loop Invariant Code Motion

SubVI Inlining

Dem

Maintain Code Modularity With Minimum Overhead



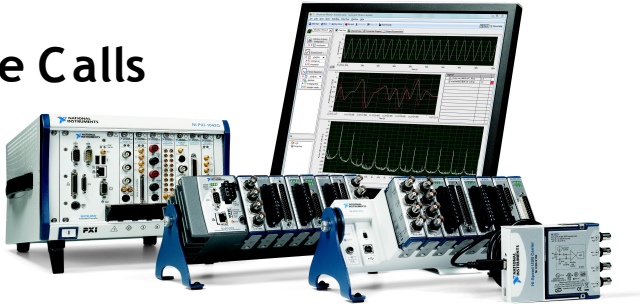
Removes all subVI call overhead

Automatically updates callers when callee's code changes

LabVIEW 2010 Performance Metrics

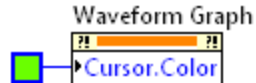
What Won't Get Faster?

Hardware Calls



DLL Calls /
Optimized C Code

User Interface
Interaction



Increased Compile Time

Compile Time **5x**

Mass Compile Time **35%**

Application Build Time **35%**

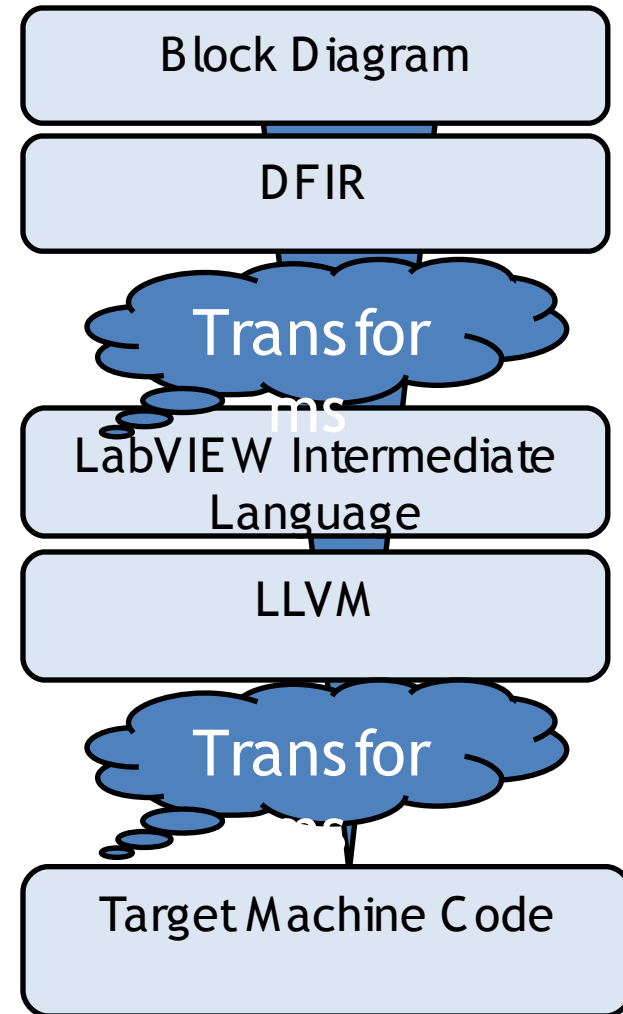
Optimizing the LabVIEW Compiler

DataFlow Intermediate Representation (DFIR)

- High-level representation
- Graph-based
- Preserves dataflow, parallelism, and execution semantics

Low-Level Virtual Machine (LLVM)

- Low-level representation
- Sequential
- Knowledge of target machine characteristics, instruction sets, alignment, etc.



New Data Acquisition Options

Evolution of Multifunction DAQ

NI-STC



NI-STC2



NI-STC3



2000

2010



Introducing US B X Series

8 new high performance US B DAQ devices, from 500 kS/s to 2 MS/s

Options for simultaneous sampling: up to 2 MS/s/ch on 8 channels

Improved performance and ease of use for the same prices as

most popular US B M Series



USB Multifunction DAQ Overview

Price

e

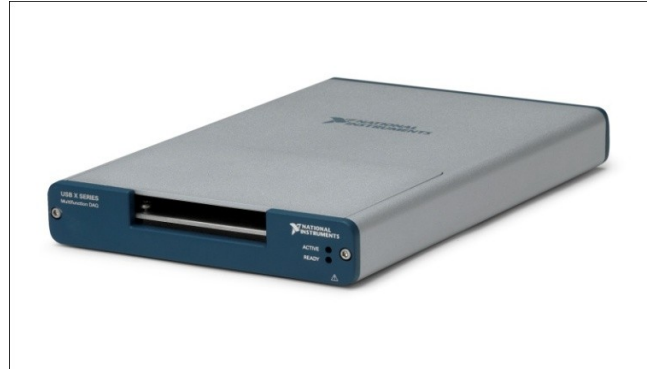
USB-6008/9
12 to 48 kS/s



Bus-powered M Series
250 to 400 kS/s



X Series: 500 kS/s to 2 MS/s



Performance

e



What's New in LabVIEW 2010

ni.com/labview

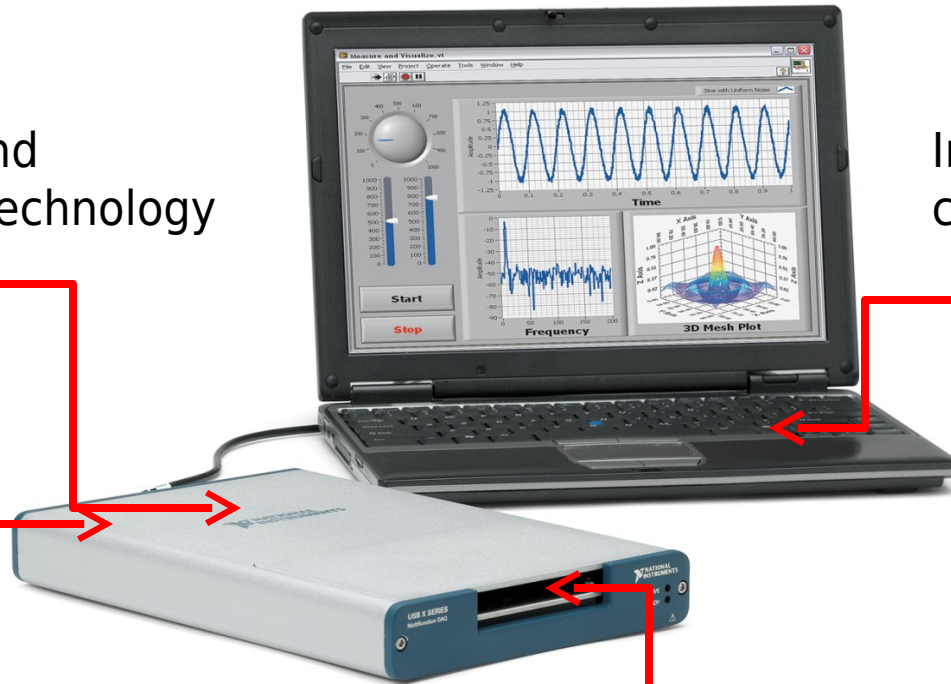
Key Features

NI-STC3 timing and synchronization technology

Improved data logging capabilities

NI Signal Streaming

Available Simultaneous Sampling



NI Signal Streaming

Traditional Devices

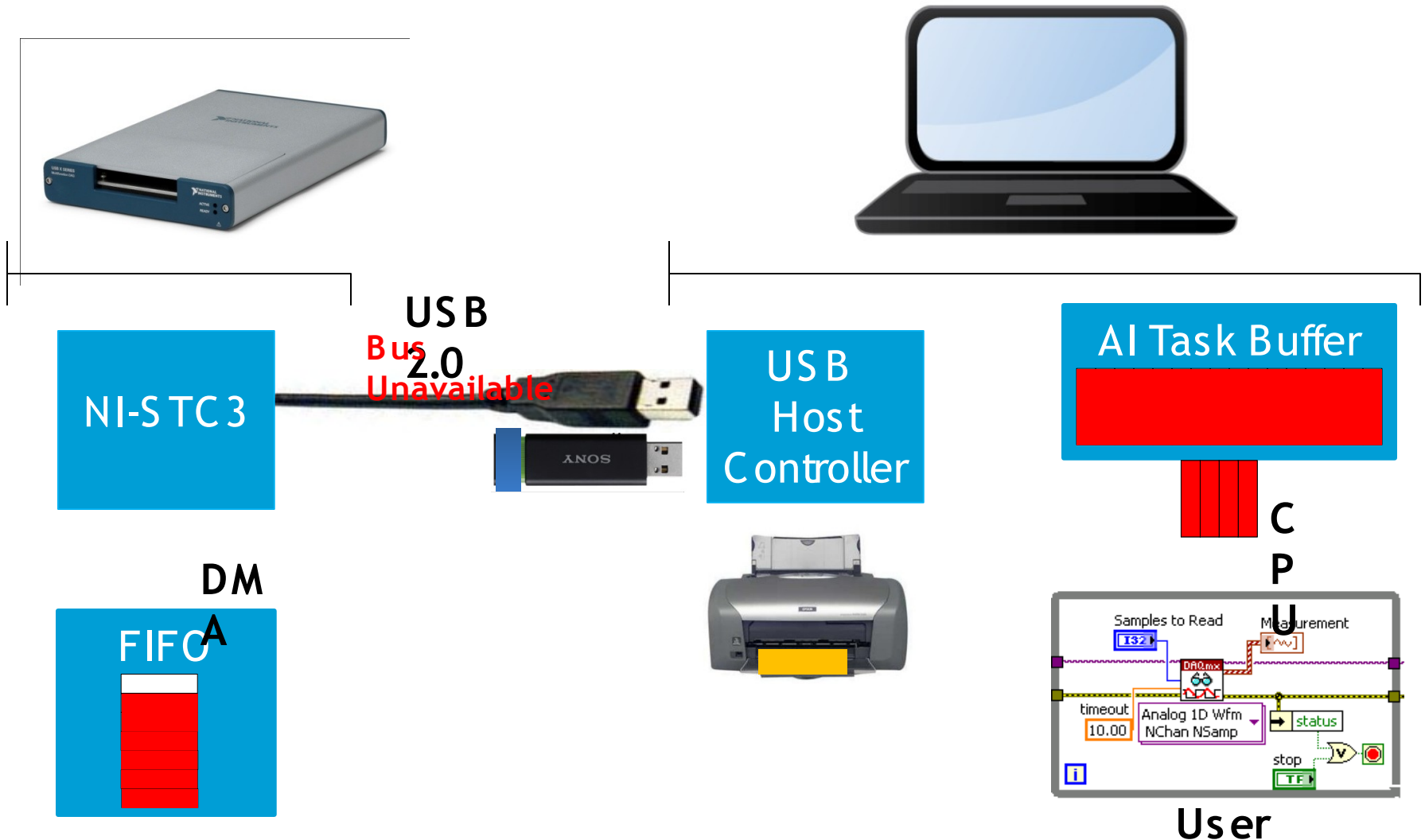
- Off-the shelf controllers
- Handles one data stream at the time
- Higher-priority tasks cause low throughput
- Higher latency due to register-level access instructions

NI Signal Streaming

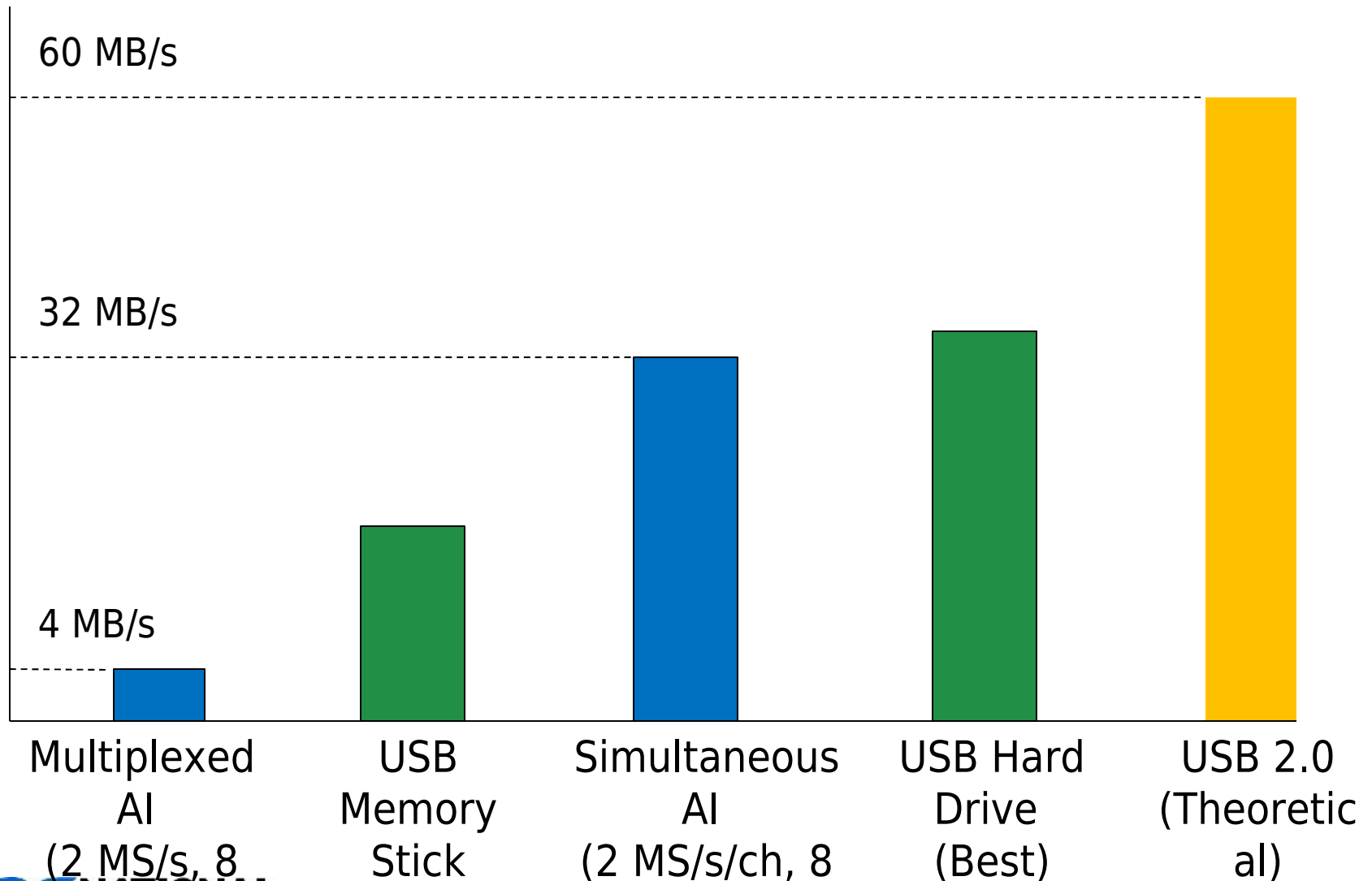
- Patented custom controller
- Handles multiple tasks at the same time
- Data streams mapped directly to USB endpoints resulting in high throughput
- Lower latency due to Message-based instructions



Streaming Input Data



USB Bandwidth Usage



Onboard Memory

USB does not guarantee bandwidth or latency

- Onboard Memory for Analog Input on S simultaneous boards
 - 32 MS and 64 MS options
 - Ensure finite acquisition, even with heavy latency or contending bus traffic

NI CompactDAQ

Simple. Complete. Data Acquisition.

“Any Sensor. Any Bus.”



DAQmx API + C Series Modules

Portable
Ch < 32
\$400 □ \$2,000

Synchronization
Mixed Signal
Ch < 128
\$2,000 □
\$7,000

Synchronization
Mixed Signal
Ch 128+
Distributed Systems
\$3,000 □ \$14,000

NI cDAQ-9188 High-Level Specifications

Ethernet version of cDAQ-9178
Gigabit Ethernet (IEEE 802.3ab)
50+ cDAQ C Series modules
Integrated signal conditioning
NI-DAQmx driver software
STC-3 (4 counters, timing engines)
NI Signal Streaming technology
Zero Configuration Networking
Web Interface Configuration

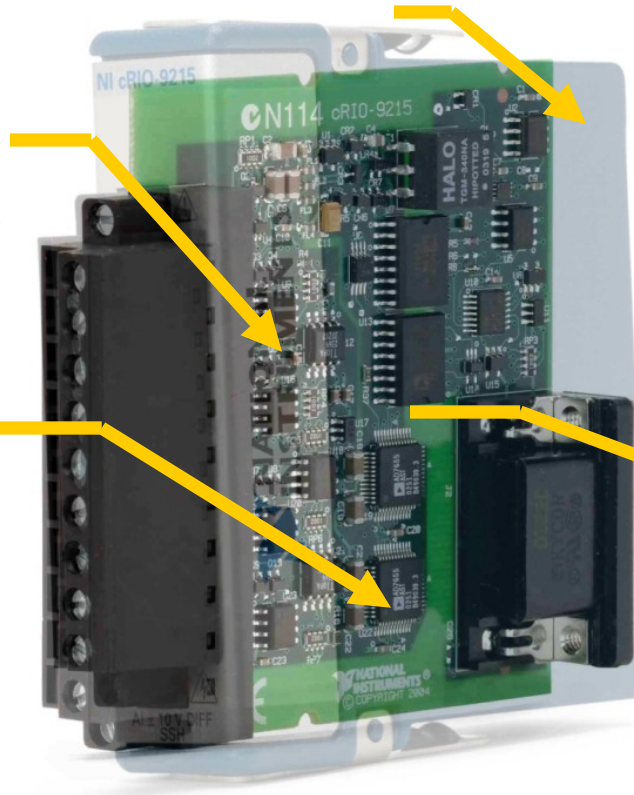
Integrated DAQ, Signal Conditioning, & Connectivity

Click to edit Main Title **Built-in Signal Conditioning**

Direct connection to sensors for temperature, pressure, acceleration, strain, load cell, current and more

High Quality Measurements

Streaming up to 1 MS/s/ch
Resolution up to 24-bit resolution



Guaranteed Accuracy

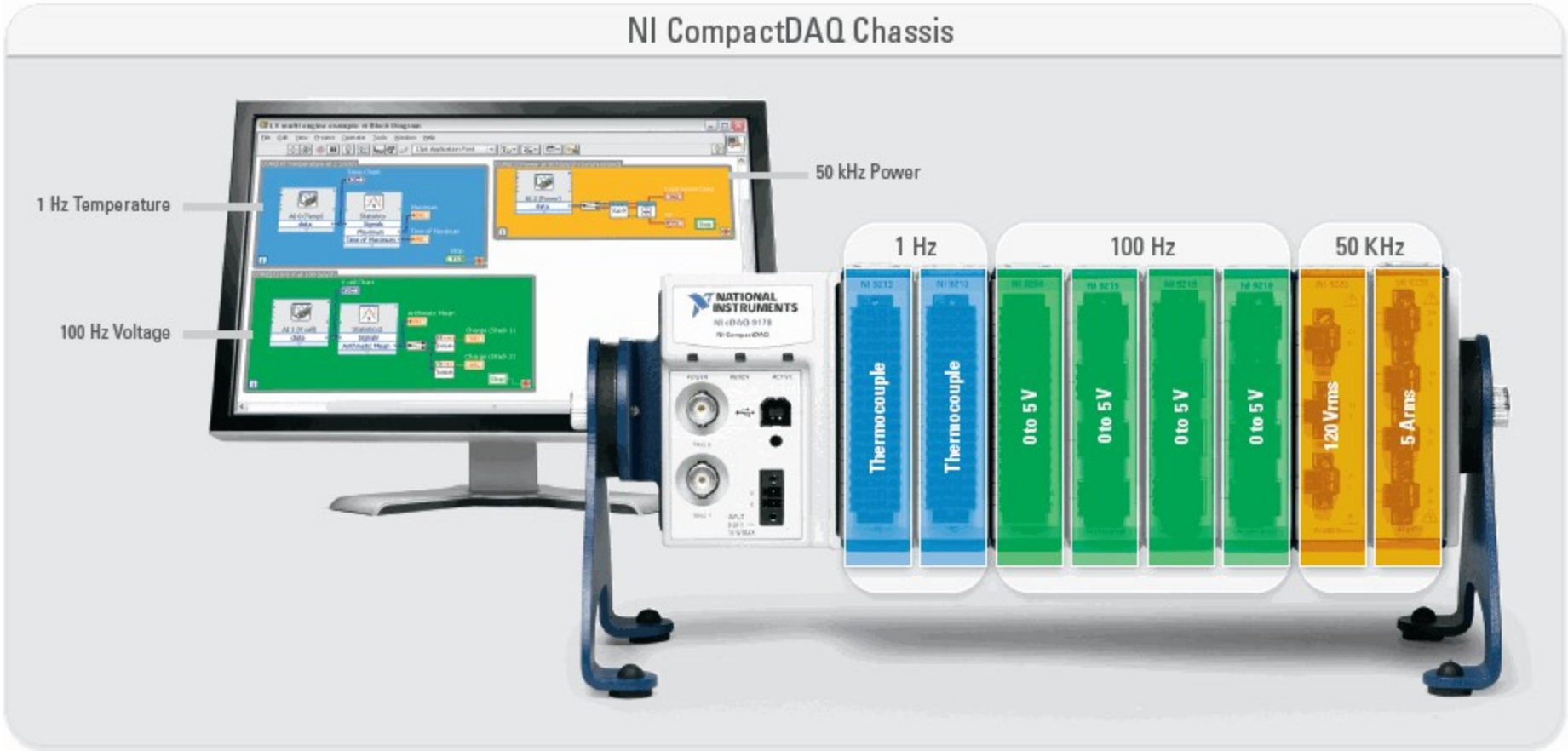
NIST traceable calibration

Signal to Backplane Isolation barrier

Isolation barrier

Safety, noise immunity, common mode rejection

Multiple Timing Engines



Web-Based Configuration

Systems Settings
Network Settings
NI Auth Settings
View modules
Update firmware
Built-in web server
Built on MS Silverlight

The screenshot shows the National Instruments Web Configuration and Monitoring interface for an NI cDAQ-9188 chassis. The interface is displayed in a Mozilla Firefox browser window. The main content area shows the following information:

Chassis Information

Model	cDAQ-9188
Serial Number	140A2EA
MAC Address	00:80:c0:2e:73:81
State	Reserved by millertime.amer.corp.natinst.com
Hostname	myChassis
Comment	Chassis in 7N.E08

Module Information

Slot 5	NI 9221	1E3A42
Slot 6	NI 9205 (DSUB)	COE70E
Slot 7	NI 9263	147CCB
Slot 8	NI 9472 (DSUB)	E9A33C



Example NI cDAQ-9188 Applications

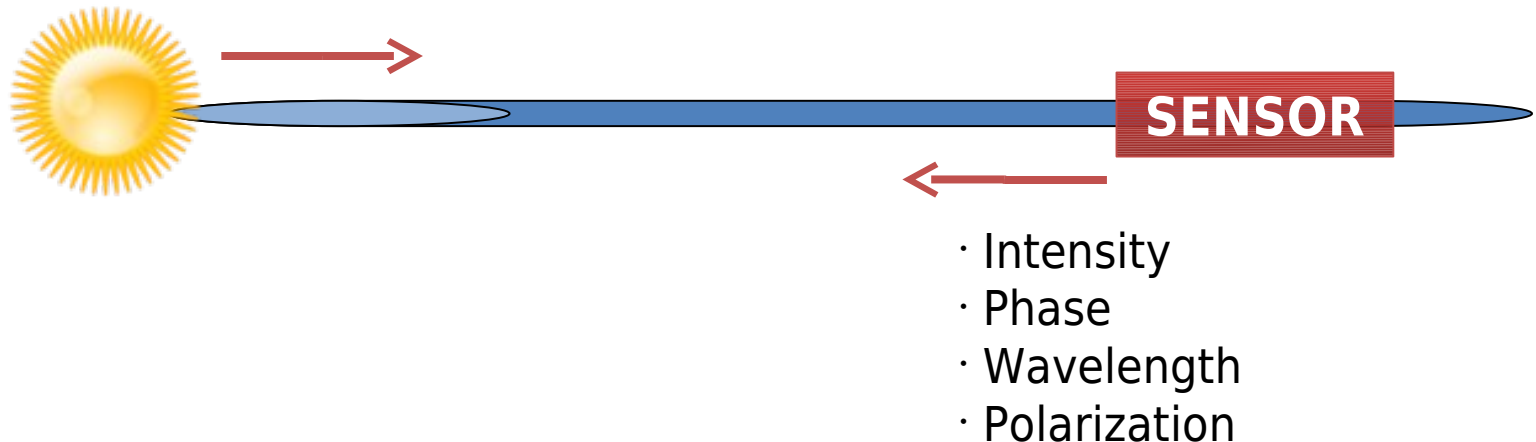
Distributed, Mixed-Signal
Measurement Systems

- Counters
- Analog Input/Output
- Digital Input/Output

Modular Laboratory Equipment
Portable Structure Health
Monitoring

Optical Sensing

Properties of light sent through an optical fiber change in response to temperature and strain
Immune to electrical effects



NI Optical Sensor Interrogator

NI PXIe-4844

4 Optical Channels

10 Hz Scan Rate

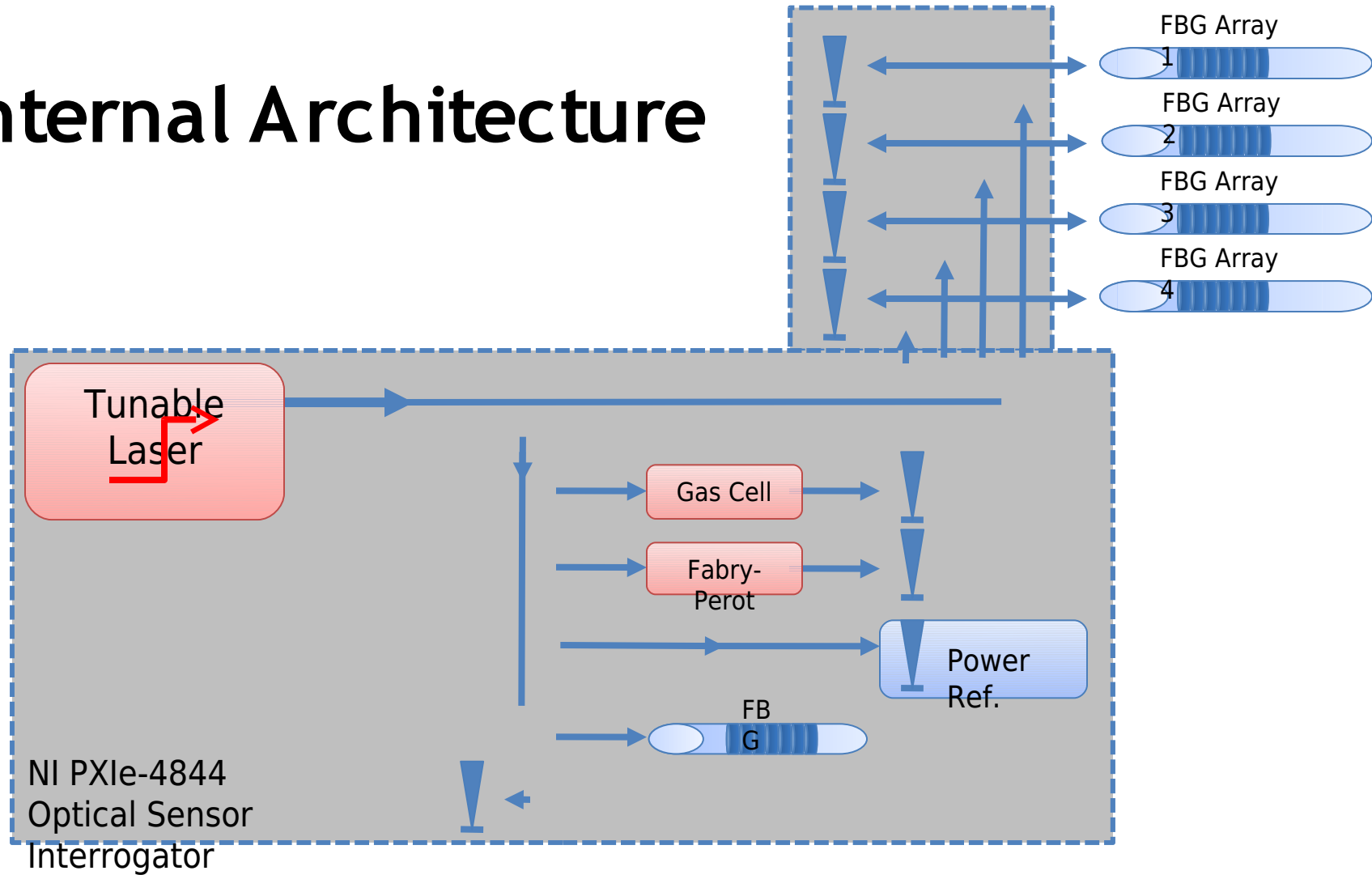
1510 to 1590 nm wavelength range

1 pm Accuracy ($\sim 1.2 \mu\text{m}$, $\sim 0.1 \mu\text{C}$)

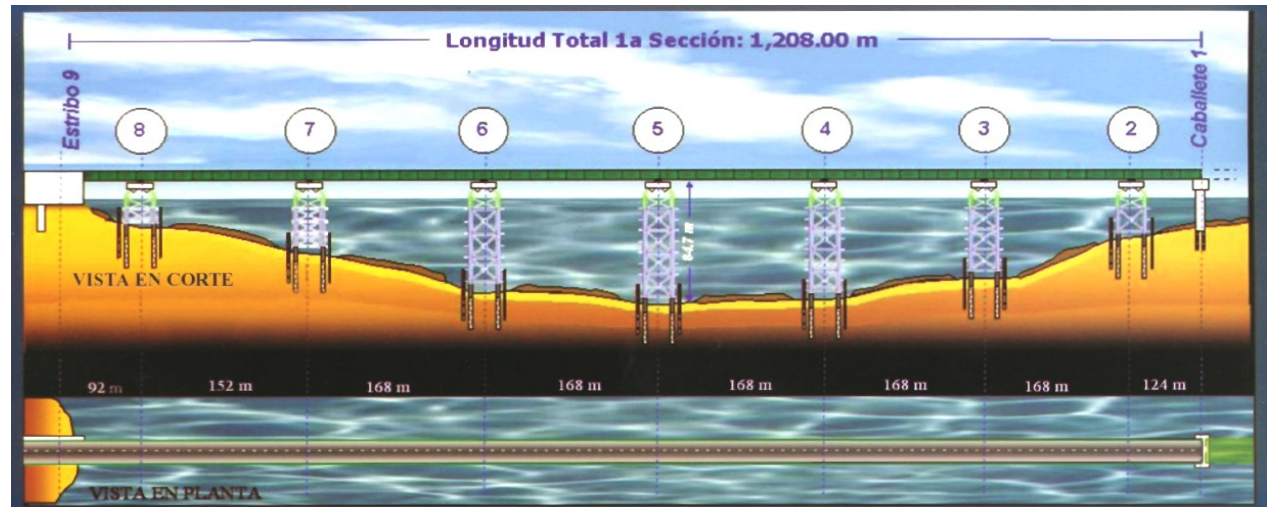
PXI Triggering



Internal Architecture



Example Applications



Environment Enhancements

Smarter Installer

Select Your Software Based on Serial Number(s)

Select Installation Option

NATIONAL INSTRUMENTS™

I have serial numbers for one or more products in this installer.

Select and activate products based on my serial numbers (Internet connection required).

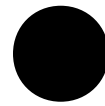
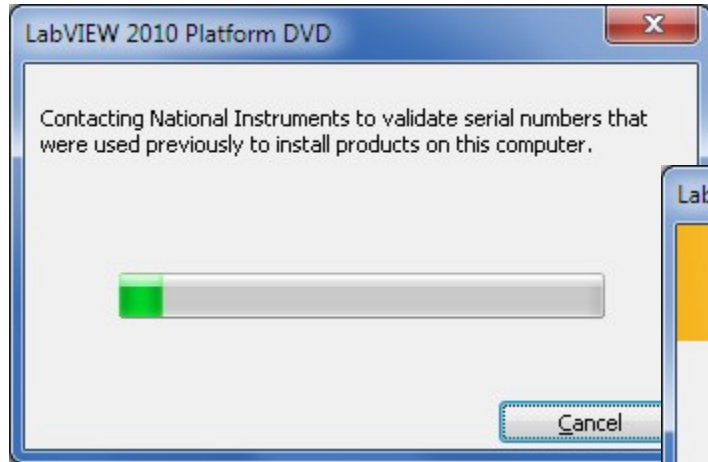
[Privacy Statement](#)

I only want to evaluate products.

<< Back Next >> Cancel

Enter your Serial Number(s), and the installer will determine what to install

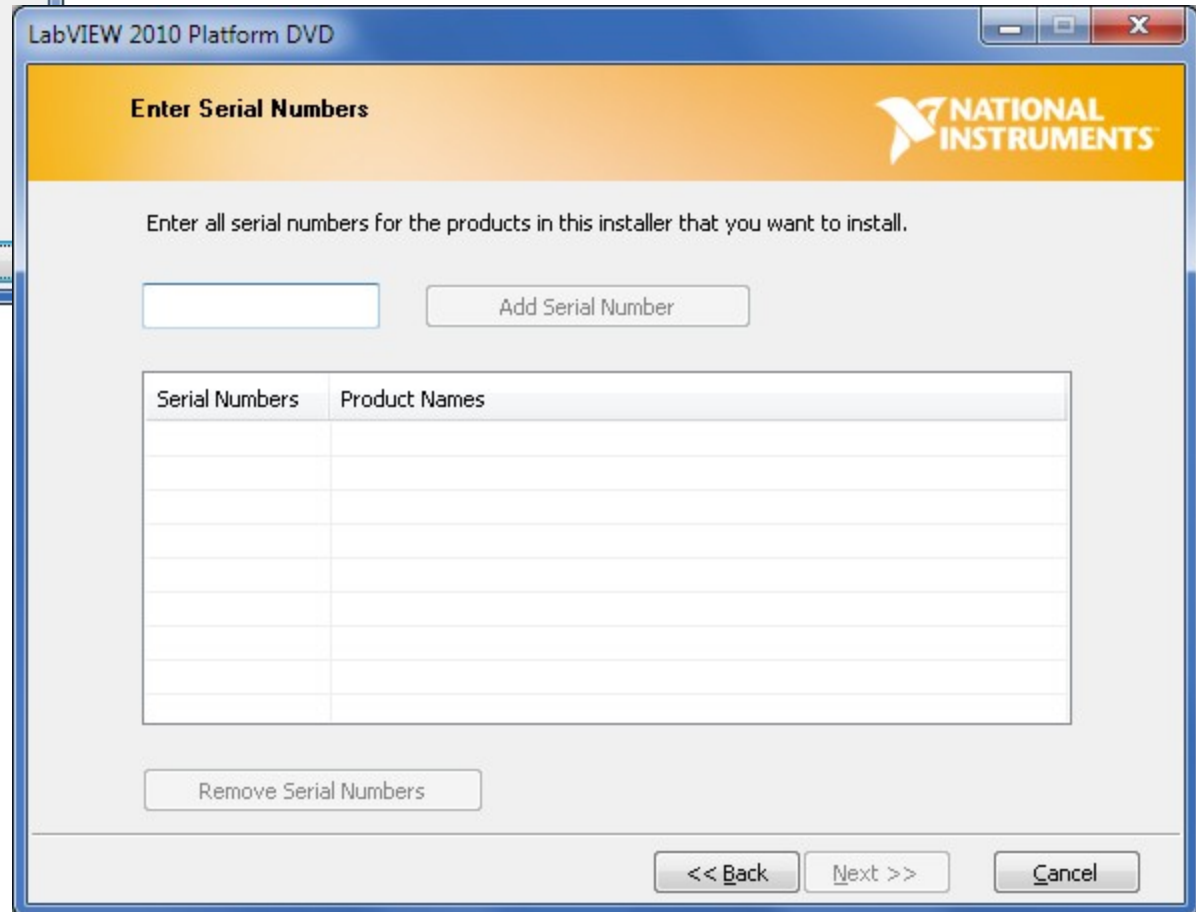
Smarter Installer



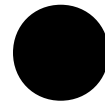
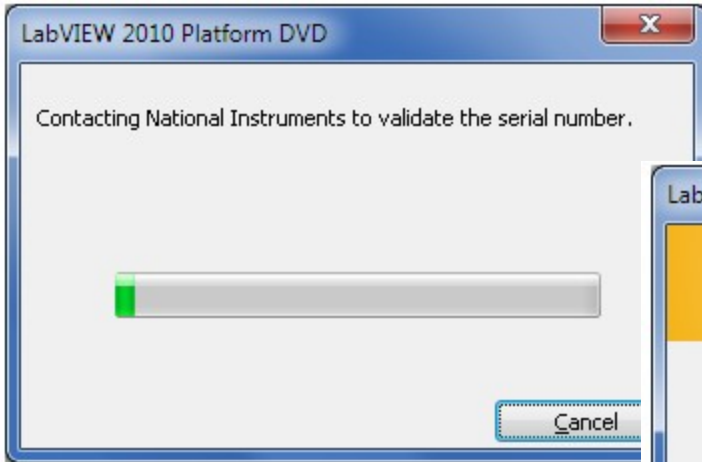
The installer will search your PC for previous serial numbers



If none found, you can enter each serial number



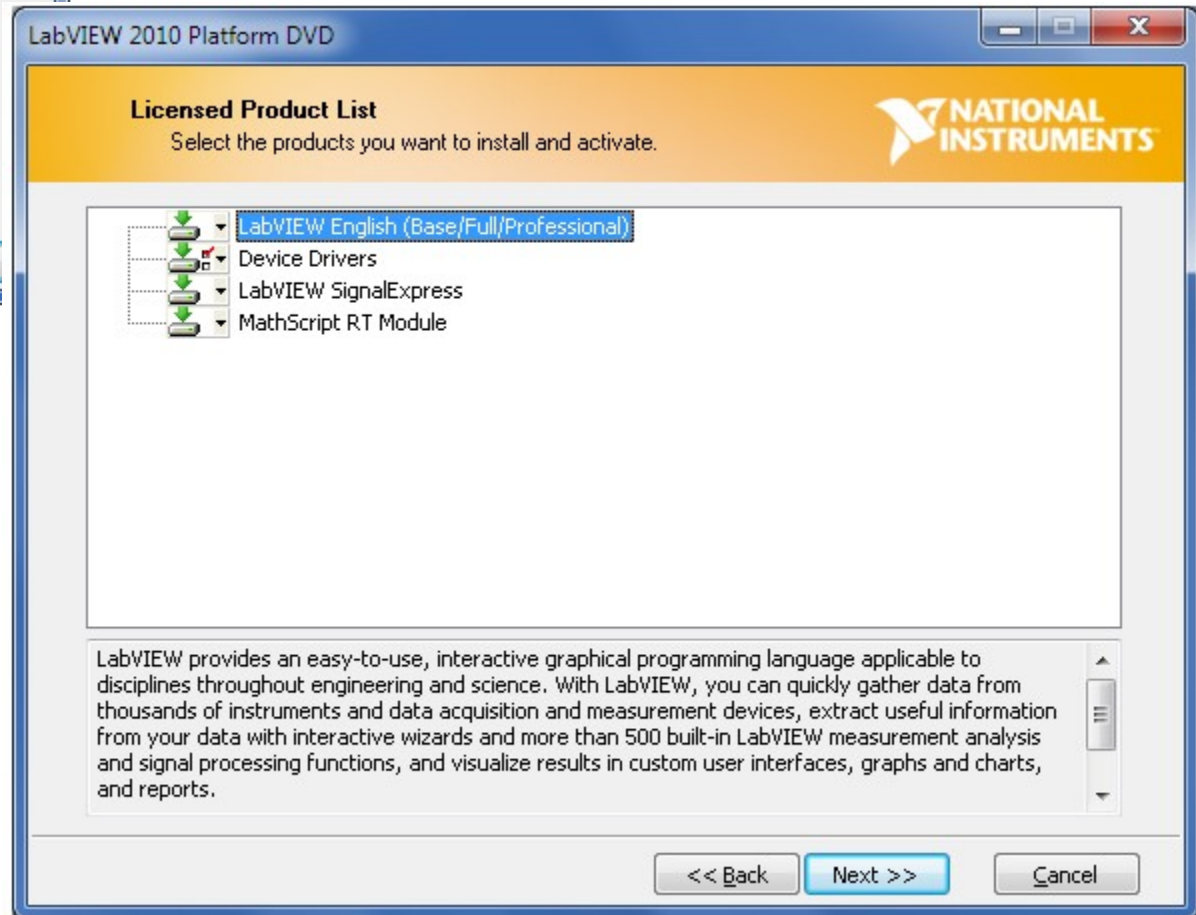
Smarter Installer



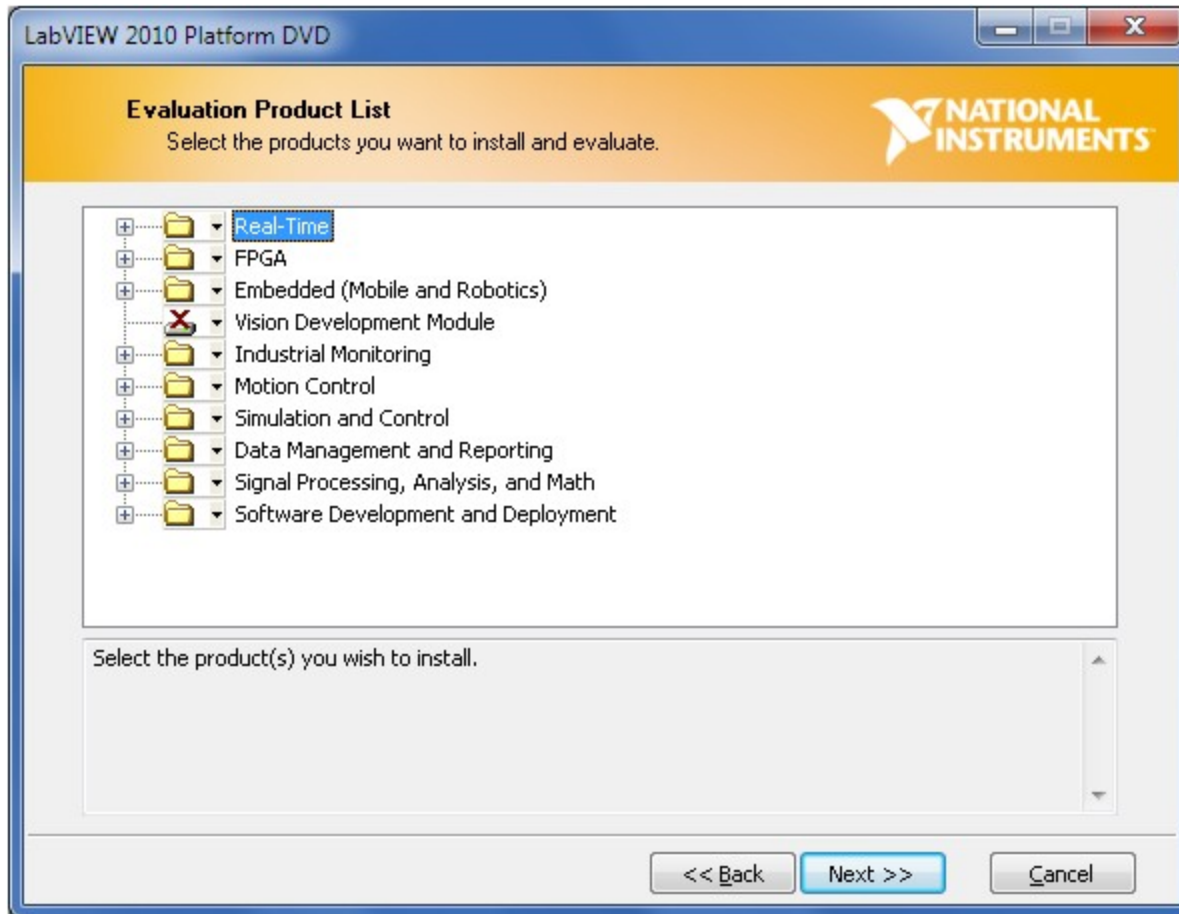
The installer will validate your serial number(s)



Then, show you what products will be installed and licensed



Smarter Installer



Next, you can choose additional products to install for a fully-functional 30-day evaluation

LabVIEW Idea Exchange



New Boolean Diagram constant design!

Labels: UI & Usability Status: In Beta

by altenbach

07-03-2009 04:37 PM

The current boolean diagram constant is potentially confusing and too elaborate.

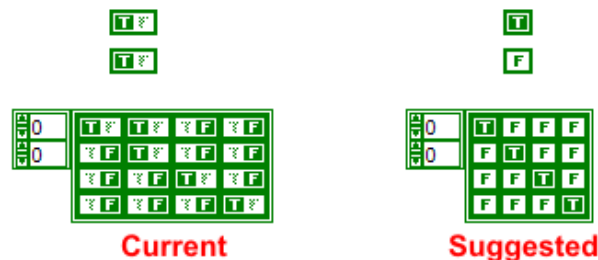
Confusing, because it almost looks like a toggle switch, so the new user might click on the **right half**, expecting an unconditional FALSE. However, there are no active areas, and an inversion of the current value occurs no matter where we click.

Too elaborate. All we need to see is the current value! Why do we need to see the "other" value greyed out??? We can guess that by simple elimination. 😊 There is too much redundant information, wasting twice as much diagram space than actually needed to display relevant information. The current design also makes e.g. 2D boolean diagram constant very confusing. Have a look at the image. Can you immediately tell that the 2D array on the left is only true on the diagonal? (I did not think so!). Now look at the suggestion on the right. Ahh... much better! 😊

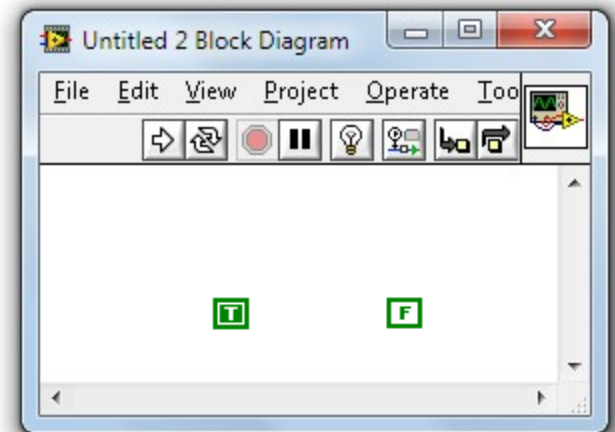
Suggestion:

The boolean diagram constant should be **smaller, simpler, and cleaner**.

The image shows the current design on the left and the suggested design on the right.



What a difference in clarity and economy!!



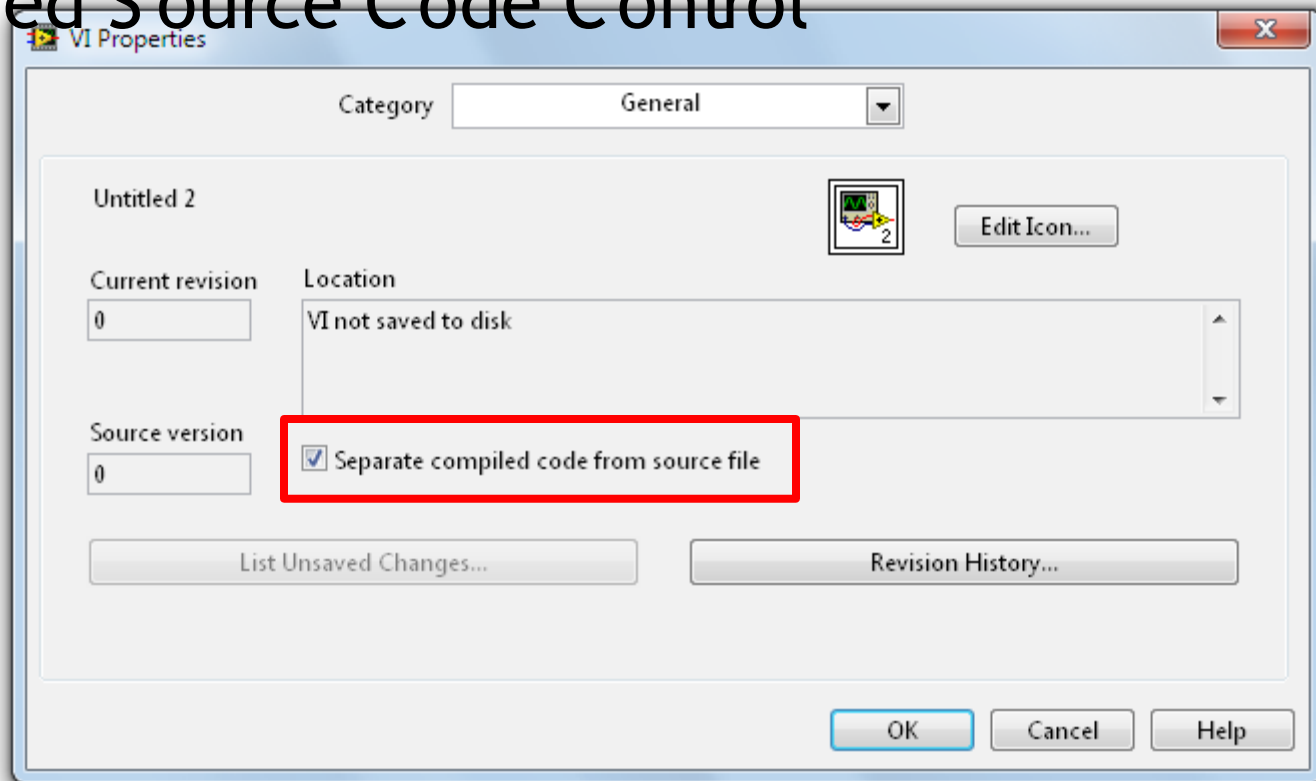
LabVIEW 2010 Idea Exchange

Feature Name	LabVIEW 2009	LabVIEW 2010	User
<i>Default Number of Undo Steps</i>			<i>PJM_LabVIEW</i>
<i>Local Variable Redesign</i>			<i>Altenbach</i>
<i>String Radix</i>			<i>Altenbach</i>
<i>Wire Labels</i>			<i>Falkpl</i>
<i>Growable Merge Error Node</i>			<i>Dany.</i>
<i>Move Switch Items in the connector pane</i>	<p>8 Mouse Clicks</p>	<p>2 Mouse Clicks</p>	<i>tst</i>

Large Application Development

Separate Compiled Code From Source File

Improved Source Code Control



Eliminate the need to re-save and re-submit files to source code control unless the graphical source code has been changed by the developer

Packed Project Libraries

Distribute and Reuse LabVIEW Code Easily

- Deploy the VI hierarchy with a single file
- Shorter build times for calling VIs
- Simplified code deployment
- .lvlibp file type

Example	# Source VIs	EXE Build Time	# VIs Built Into PPL	EXE Build Time	Build Time Improvement
Agilent 34401 Acquire and Graph - SW Triggered.vi	53	6.3 s	22	5.15 s	18.2%
E-Mail Notification.vi	102	8.66 s	68	5.82 s	32.8%
Update Weather Data.vi	71	12.97 s	46	5.48 s	57.8%
Custom Example	1000	53.93 s	999	15.94 s	70.4%

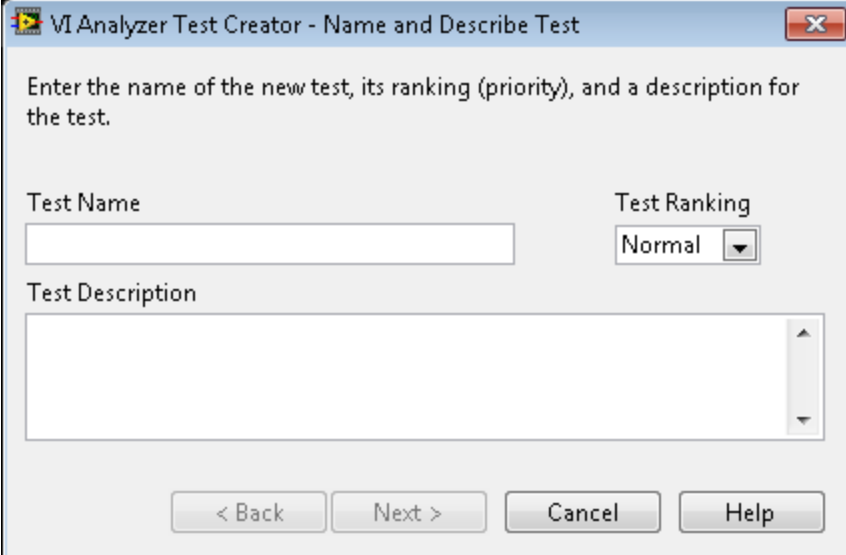
LabVIEW Add-Ons for Software Validation

Unit Test Framework Toolkit

- 30% faster test execution
- Custom definition of test vector ranges

VI Analyzer Toolkit

- Create your own tests in VI Analyzer using LabVIEW Scripting



The screenshot shows a dialog box titled "VI Analyzer Test Creator - Name and Describe Test". The dialog contains the following fields and controls:

- Test Name:** A text input field.
- Test Ranking:** A dropdown menu currently set to "Normal".
- Test Description:** A large text area for entering a description.
- Navigation Buttons:** "< Back", "Next >", "Cancel", and "Help".

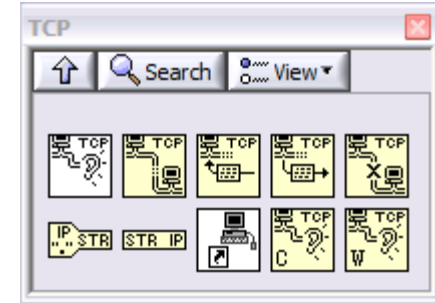
Instructions inside the dialog: "Enter the name of the new test, its ranking (priority), and a description for the test."

Target-to-Host Data Transfer

Network Connectivity Options in LabVIEW

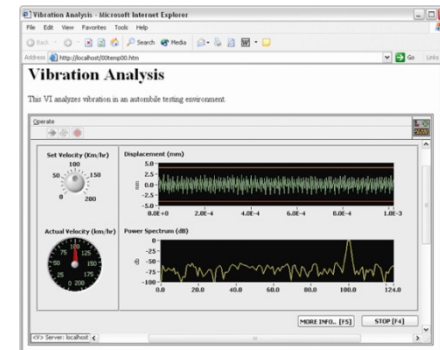
TCP/IP and UDP

Define low-level communication protocols



Remote Front Panels

Quickly embed a front panel in a browser



Shared Variables

Quickly develop distributed systems through drag-and-drop configuration



Network Streaming in LabVIEW

Dem o

Based on TCP

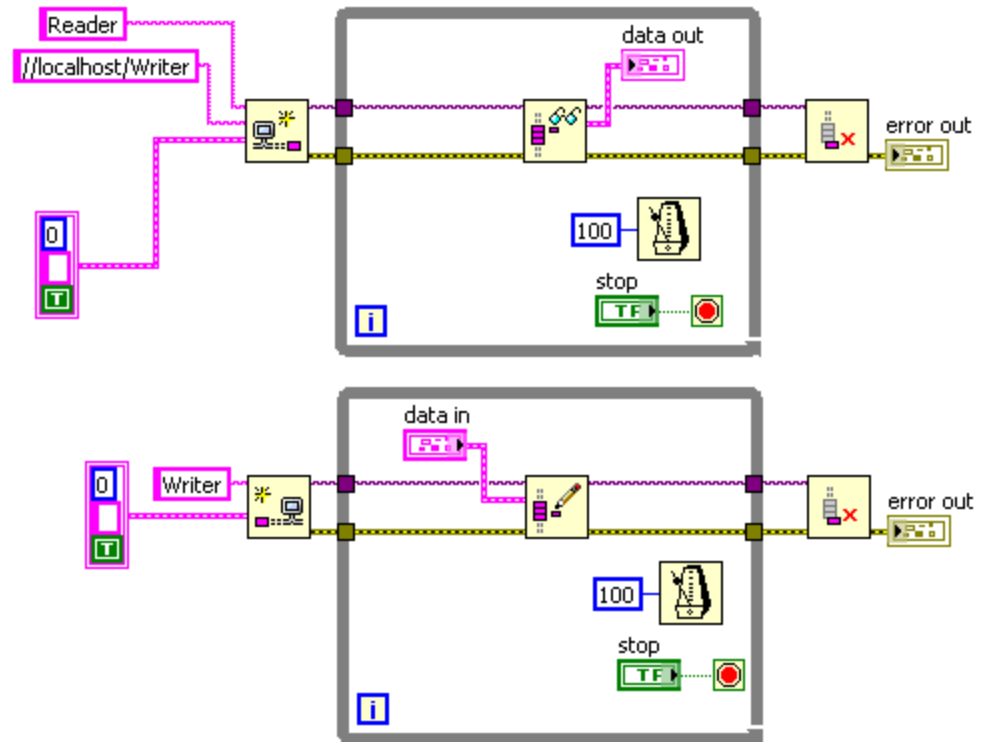
High throughput

Queues-like Experience

Easy to program

Adaptive Data Type

No need to type cast



What's New

LabVIEW 2010 Modules

LabVIEW 2010 Real-Time Products

- **LabVIEW Real-Time Module**

- Web-based configuration and monitoring of networked targets
- Simpler host-to-target transfer of data using Network Streams
- Publish variables via Web Services
- Software IEEE 1588 as timing source for Timed Loop

- **NI-Real-Time Hypervisor 2.0 - One PC, Two OSs!**

- Shared memory for higher data transfer rates between OSs
- Higher customization for CPU partitioning

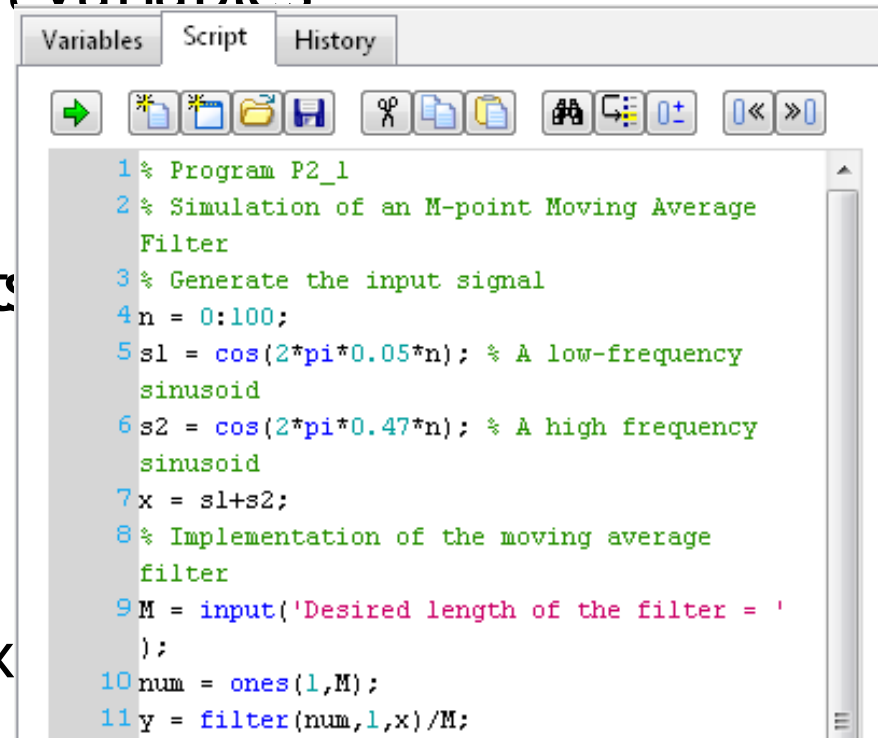
LabVIEW 2010 MathScript RT Module

MathScript Node

- Validate your custom .m files for deterministic behavior
- Automatically create output variables

MathScript Window

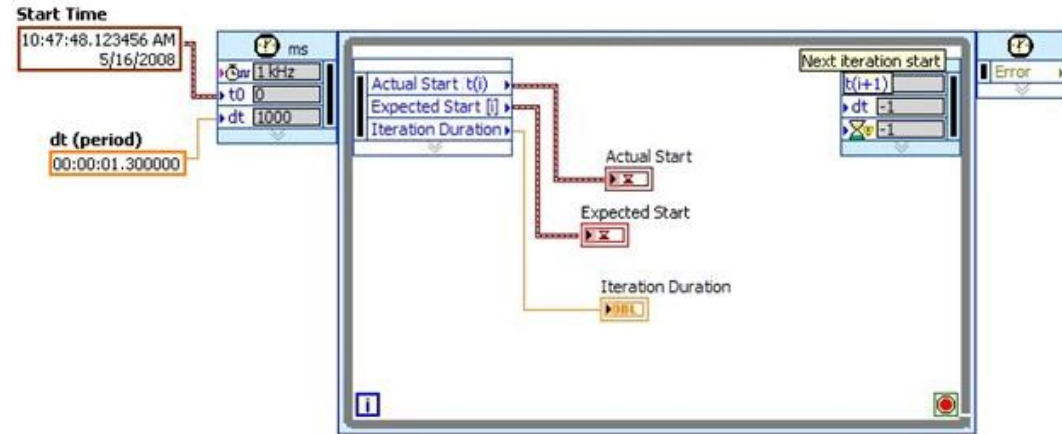
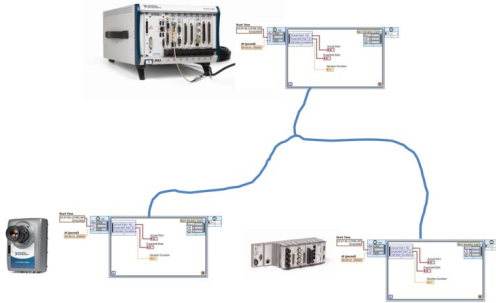
- Performance improvements
- Enhanced text-editor
 - Syntax highlighting
 - Line numbers
 - Find/replace text dialog box
 - Bookmarks



```
1 % Program P2_1
2 % Simulation of an M-point Moving Average
  Filter
3 % Generate the input signal
4 n = 0:100;
5 s1 = cos(2*pi*0.05*n); % A low-frequency
  sinusoid
6 s2 = cos(2*pi*0.47*n); % A high frequency
  sinusoid
7 x = s1+s2;
8 % Implementation of the moving average
  filter
9 M = input('Desired length of the filter = ');
  );
10 num = ones(1,M);
11 y = filter(num,1,x)/M;
```

Free Software-1588 Synchronization

Synchronization across NI Platforms



Specifications

- 10's nanosecond event resolution on PCI and PXI
- Sub millisecond event resolution on other LabVIEW RT targets

Features

- Control the timed loop with



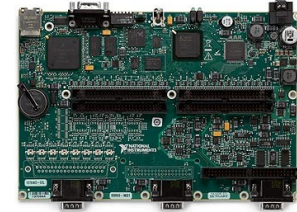
absolute time driven by 1588 or GPS

What's New in LabVIEW 2010

- Connect over standard network/switch hardware

ni.com/labview

Dem NI TimeSync Platform Support



PXI

CompactRIO

Single-BoardRIO

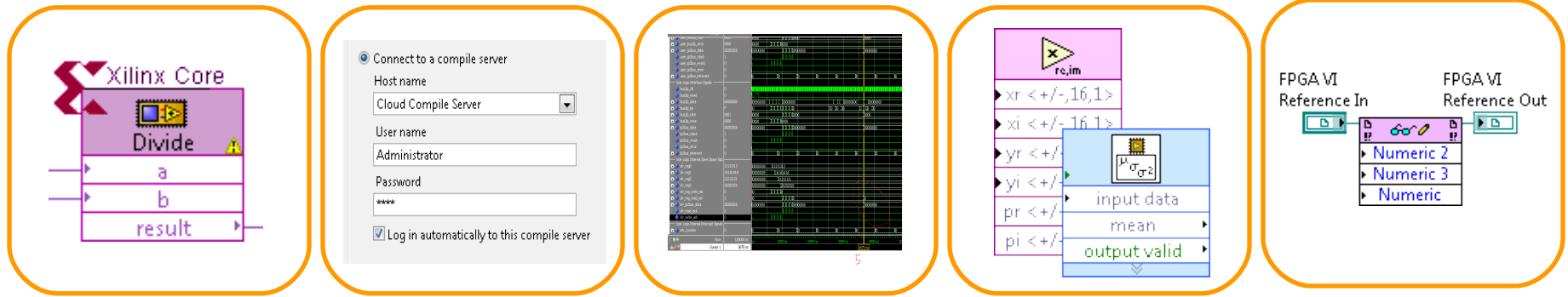
Compact FieldPoint

Smart Cameras

What's New in LabVIEW 2010

ni.com/labview

LabVIEW 2010 FPGA Module



IP Integration Node - Directly import Xilinx .xco files or your own VHDL easily

New Compilation Flow - Earlier Compilation Estimates and Build Specifications

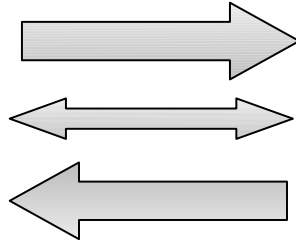
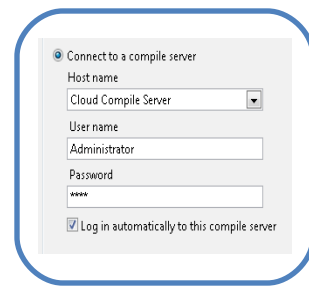
Cycle-Accurate Simulation - Use ModelSim for Cycle-Accurate Simulation

More IP Blocks - New IP for Statistics, Complex Multiplication, and More

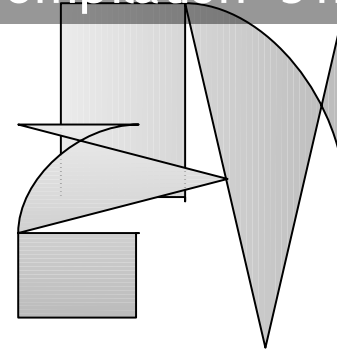
Host Improvements - New Dynamic reference for Host VI reuse

LabVIEW 2010 FPGA Compilation

LabVIEW FPGA Compile Farm Toolkit



Compilation "Smart" Server



Compilation Workers

LabVIEW FPGA Development

New Rugged CompactRIO: cRIO 9023/9025 Real-Time Controllers



533 and 800 MHz Processors

Dual 9-30V Inputs

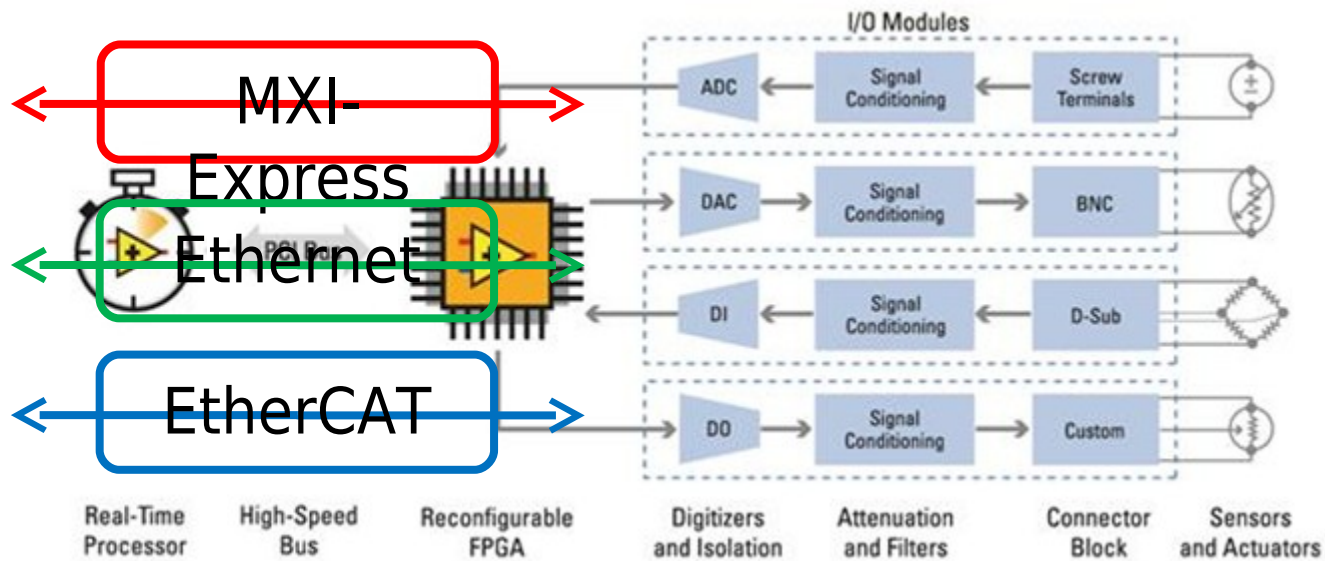
Up to 4 GB internal S storage

-40 to 70C Operating temp

Dual I/O Expansion | What's New in LabVIEW 2010

ni.com/labview

New cRIO (with FPGA) Chassis



NI RIO Expansion Chassis

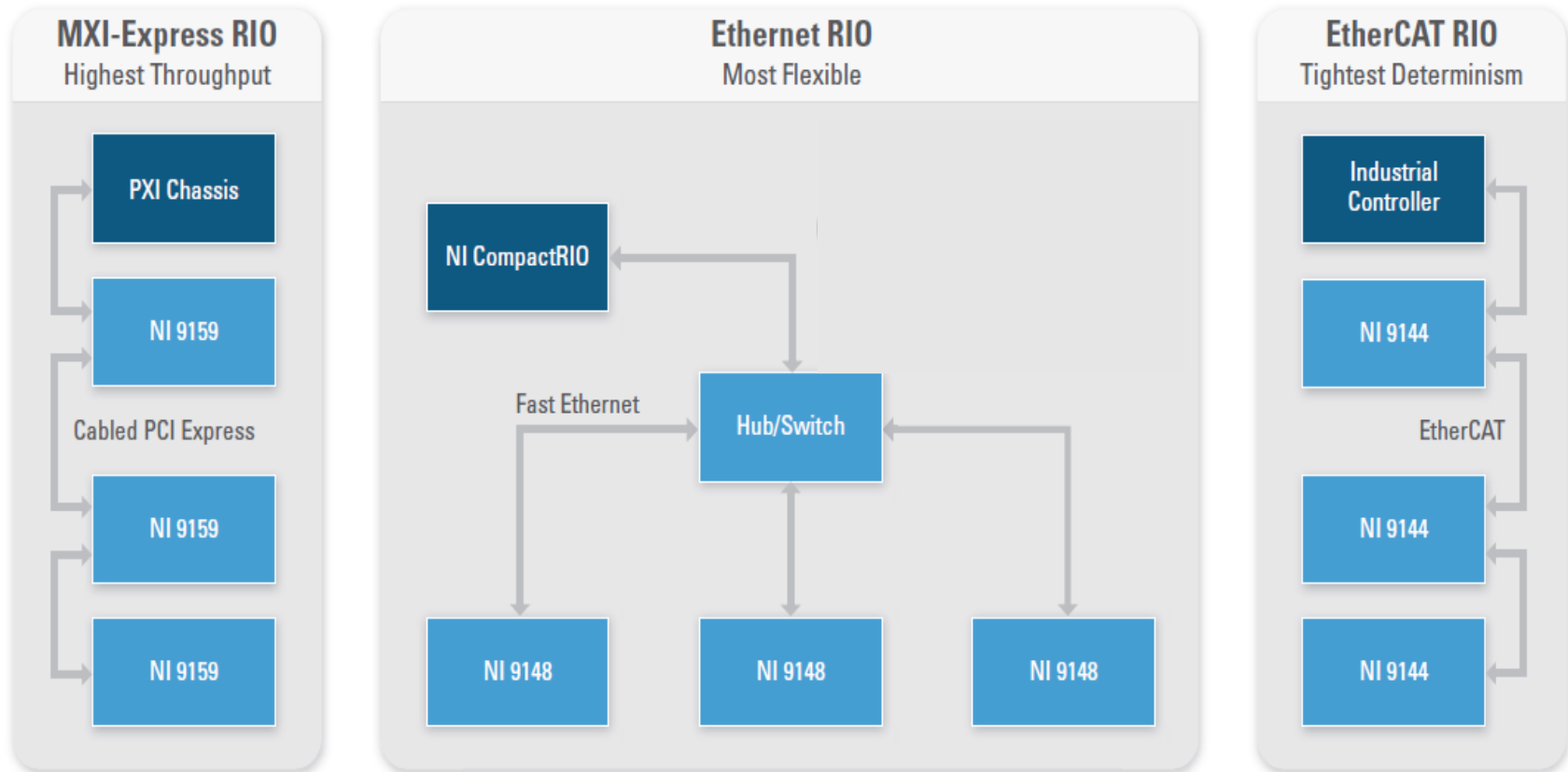


Figure 1. RIO expansion I/O provides solutions for high-performance rack-mount test, distributed deterministic control, and more.

Expansion Chassis Details

	MXI-Express RIO	Ethernet RIO	EtherCAT RIO
	NI 9157, 9159	NI 9148	NI 9144
# Slots	14	8	8
FPGA	Virtex5 (LX85 or LX110)	Spartan 2M	Spartan 2M
Network Topology	Daisy-Chain	Same as Ethernet	Daisy-Chain
Distance	7m between nodes	100m before repeater	100m before repeater
Synchronization	FPGA based DIO	FPGA based DIO	Implicit in bus operation
Jitter	<10 μ s	No Spec	<1 μ s
Bus Performance	250MB/s	100Mb/s	100Mb/s
API Support	Host Interface	Host Interface/RSI	RSI
Windows / RT	Yes / Yes	Yes / Yes	No / Yes

Building LabVIEW Add-Ons

LabVIEW 2010

Extending the Platform

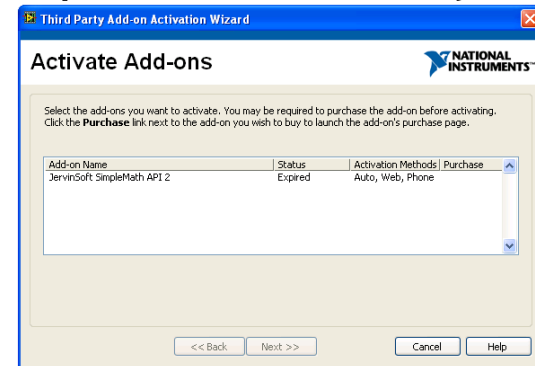
Dem o

Licensing and Activation for 3rd Party Add-ons

Commercial Grade Activation solution from Concept S software

Allows for LabVIEW Add-ons created by the development community to implement 30-day software trials

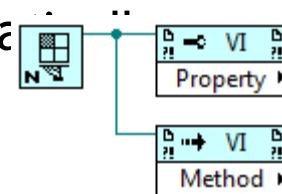
Fully integrated in LabVIEW 2010



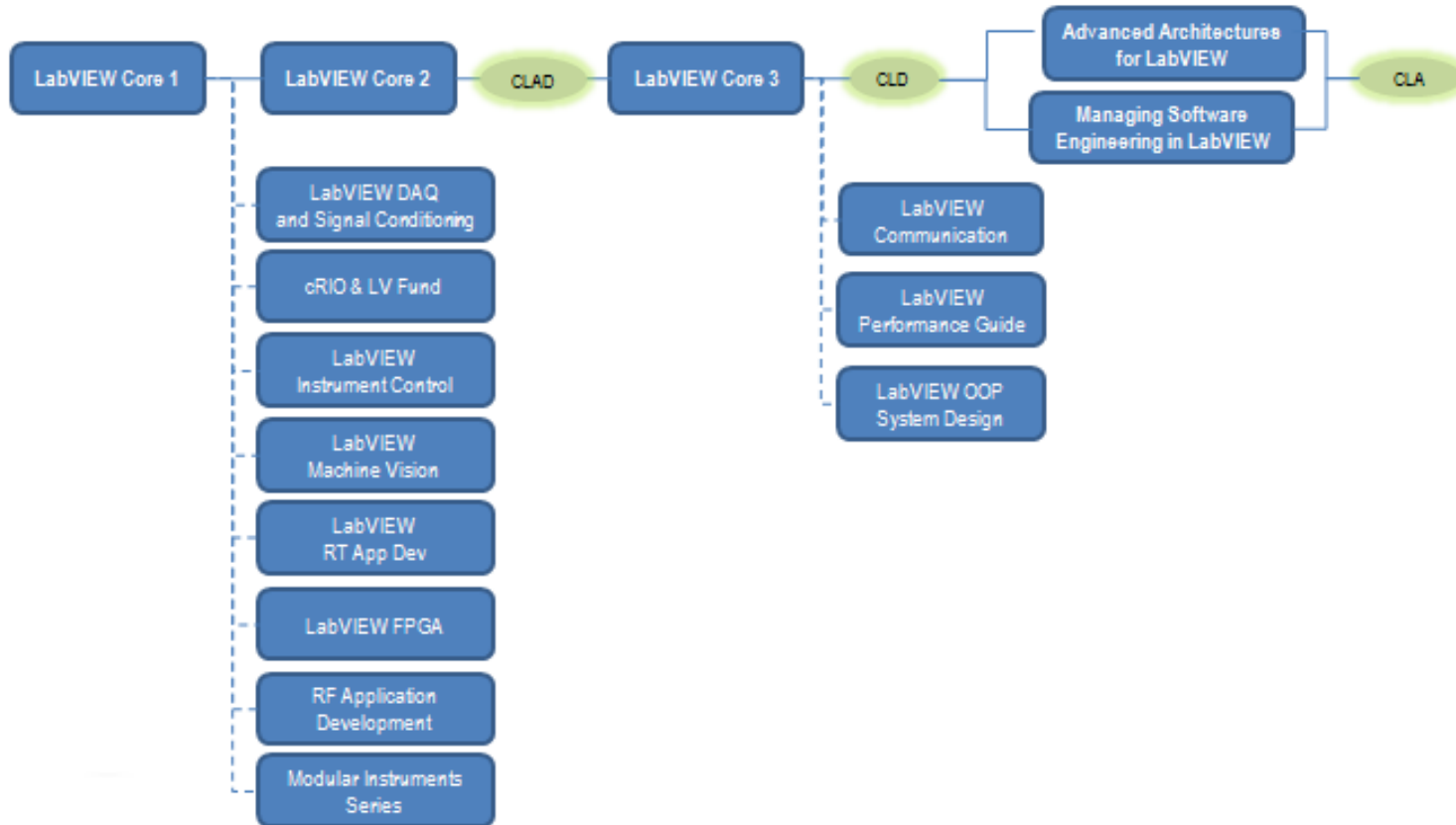
LabVIEW Scripting

Intended for power users to enhance the capabilities of LabVIEW during editing

Used to inspect, modify, or generate LabVIEW code automatically



LabVIEW Training and Certification Path



ni.com/romania/training

Training and Certification Membership Program



Flat rate training program with

- ✓ unlimited access to all scheduled courses for one or two years,
- ✓ personalized training programs,
- ✓ option to retake all courses,
- ✓ skill validation with professional credentials,
- ✓ and money-back satisfaction guarantee.

Thank you for your attention!