

Advanced Architectures in LabVIEW™ Course Manual

Course Software Version 2011
February 2012 Edition
Part Number 372845B-01

Copyright

© 2009–2012 National Instruments Corporation. All rights reserved.

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

National Instruments respects the intellectual property of others, and we ask our users to do the same. NI software is protected by copyright and other intellectual property laws. Where NI software may be used to reproduce software or other materials belonging to others, you may use NI software only to reproduce materials that you may reproduce in accordance with the terms of any applicable license or other legal restriction.

For components used in USI (Xerces C++, ICU, HDF5, b64, Stingray, and STLport), the following copyright stipulations apply. For a listing of the conditions and disclaimers, refer to either the `USICopyrights.chm` or the *Copyrights* topic in your software.

Xerces C++. This product includes software that was developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright 1999 The Apache Software Foundation. All rights reserved.

ICU. Copyright 1995–2009 International Business Machines Corporation and others. All rights reserved.

HDF5. NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities

Copyright 1998, 1999, 2000, 2001, 2003 by the Board of Trustees of the University of Illinois. All rights reserved.

b64. Copyright © 2004–2006, Matthew Wilson and Synesis Software. All Rights Reserved.

Stingray. This software includes Stingray software developed by the Rogue Wave Software division of Quovadx, Inc. Copyright 1995–2006, Quovadx, Inc. All Rights Reserved.

STLport. Copyright 1999–2003 Boris Fomitchev

Trademarks

LabVIEW, National Instruments, NI, ni.com, the National Instruments corporate logo, and the Eagle logo are trademarks of National Instruments Corporation. Refer to the *Trademark Information* at ni.com/trademarks for other National Instruments trademarks.

Other product and company names mentioned herein are trademarks or trade names of their respective companies.

Members of the National Instruments Alliance Partner Program are business entities independent from National Instruments and have no agency, partnership, or joint-venture relationship with National Instruments.

Patents

For patents covering National Instruments products/technology, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your media, or the *National Instruments Patent Notice* at ni.com/patents.

SAMPLE

Worldwide Technical Support and Product Information

ni.com

Worldwide Offices

Visit ni.com/niglobal to access the branch office Web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

National Instruments Corporate Headquarters

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

For further support information, refer to the [Additional Information and Resources](#) appendix. To comment on National Instruments documentation, refer to the National Instruments Web site at ni.com/info and enter the Info Code *feedback*.

Contents

Student Guide

A. NI Certification	v
B. Course Description	vi
C. What You Need to Get Started	vi
D. Installing the Course Software.....	vii
E. Course Goals.....	vii
F. Course Conventions.....	viii

Lesson 1

Software Architecture – Introduction

A. Software Architecture – Quality	1-2
B. Documenting a Software Architecture	1-23
C. Architecture Quality	1-43

Lesson 2

Designing an API

A. APIs	2-2
B. API Design Techniques	2-6
C. Polymorphic VIs for an API.....	2-26
D. Project Libraries.....	2-35
E. Passing Data in an API.....	2-40

Lesson 3

Multiple Processes and Inter-Process Communication

A. Introduction – Processes	3-2
B. Foundational APIs	3-18
C. Foundational Design Patterns	3-47
D. Scaling Foundational Design Patterns	3-79
E. What is an Asynchronous Dynamic Process?.....	3-97
F. By Reference Inter-Process Data Storage.....	3-135
G. Inter-Target Communication	3-158
H. Analysis of LabVIEW Communication APIs.....	3-165
I. Communication Reference Architectures.....	3-171

Lesson 4

Advanced User Interface Techniques

A. Subpanels – Examples	4-4
B. XControls Overview – Review of Controls.....	4-17
C. XControl Abilities.....	4-30
D. XControl Properties and Methods	4-52
E. XControl Development – Editing XControls in Use.....	4-60

Lesson 5

Introduction to Object-Oriented Programming in LabVIEW

A. What is Object Oriented Programming?.....	5-2
B. Recommended Resources	5-26
C. Class Diagrams and Design (Optional Section)	5-30

Lesson 6

Plug-In Architecture

A. Using Plug-In Architecture with VI Server	6-4
B. Using Plug-In Architecture with LVOOP	6-11

Lesson 7

Tips, Tricks, & Other Techniques

A. Variant Attributes	7-2
B. Callback VIs	7-4
C. VI Scripting.....	7-7
D. What is a Drop In VI?.....	7-10
E. Other	7-25

Lesson 8

Error Handling

A. Basic Error Handling Strategies	8-2
B. Comprehensive Error Handling Strategies	8-5
C. LabVIEW Implementation Examples.....	8-21

Lesson 9

Additional Resources

A. Now What?	9-2
--------------------	-----

Appendix A

Additional XControl Programming Techniques

A. Refresh Section	A-2
B. Shortcut Menus	A-4
C. Resizing the XControl	A-7
D. Defining Custom Events.....	A-8

Appendix B

Additional Information and Resources

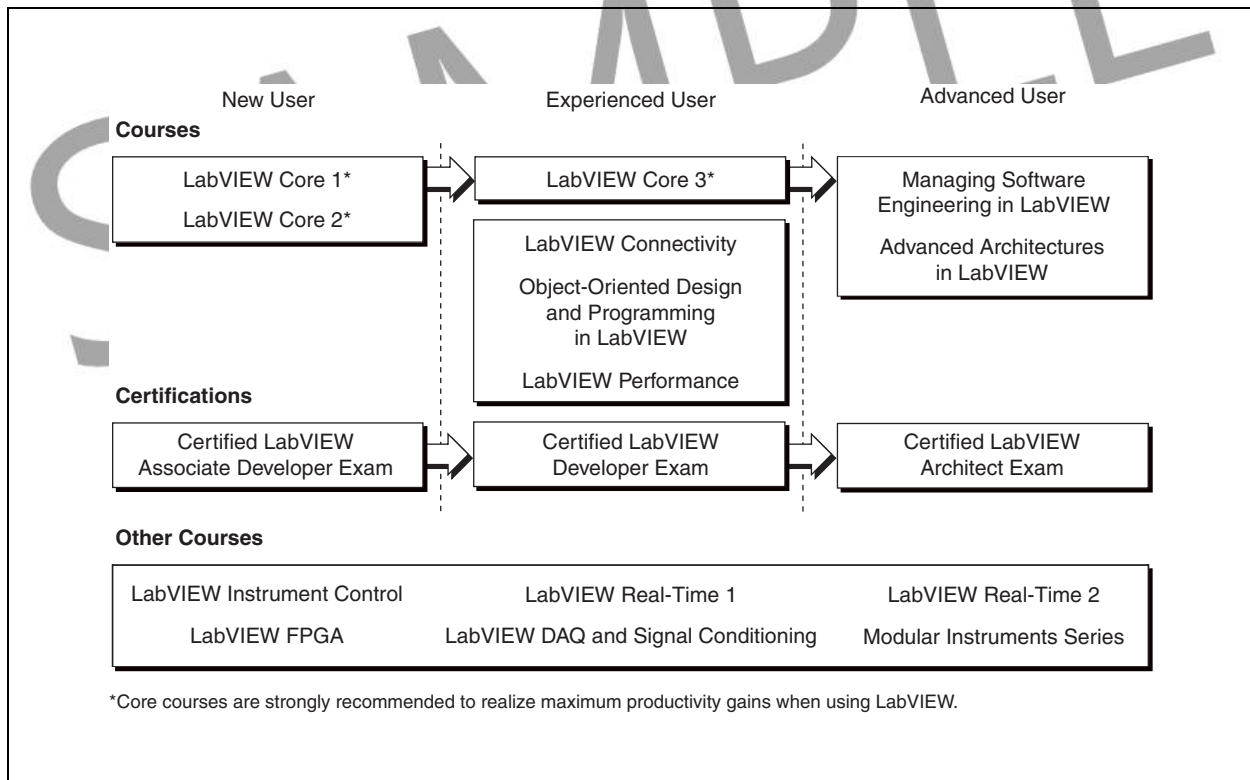
Student Guide

Thank you for purchasing the Advanced Architectures in LabVIEW course kit. This course manual and the accompanying software are used in the three-day, hands-on Advanced Architectures in LabVIEW course.

You can apply the full purchase price of this course kit toward the corresponding course registration fee if you register within 90 days of purchasing the kit. Visit ni.com/training to register for a course and to access course schedules, syllabi, and training center location information.

A. NI Certification

The *Advanced Architectures in LabVIEW* course is part of a series of courses designed to build your proficiency with LabVIEW and help you prepare for exams to become an NI Certified LabVIEW Developer and NI Certified LabVIEW Architect. The following illustration shows the courses that are part of the LabVIEW training series. Refer to ni.com/training for more information about NI Certification.



B. Course Description

In the *Advanced Architectures in LabVIEW* course you participate in discussions and work independently and collaboratively to learn how to architect an application and then design the components to support the architecture.

This course assumes you have taken the *LabVIEW Core 3* course or have equivalent experience.

The course is divided into lessons, each covering a topic or a set of topics. Each lesson consists of the following parts:

- An introduction that describes what you will learn.
- A discussion of the topics.
- A set of exercises that reinforces the topics presented in the discussion. Some lessons include optional exercises or challenge steps to complete if time permits.
- A summary that outlines important concepts and skills taught in the lesson.



Note For course manual updates and corrections, refer to ni.com/info and enter the Info Code aalerrata.

C. What You Need to Get Started

Before you use this course manual, make sure you have the following items:

- Computer running Windows 7/Vista/XP
- NI LabVIEW 2011
- Advanced Architectures in LabVIEW CD, which contains the following:

Directory	Description
Exercises	Folder for saving VIs created during the course and for completing certain course exercises
Solutions	Folder containing the solutions to all the course exercises
Advanced Architectures in LabVIEW 2011 - Course Manual.pdf	<i>Advanced Architectures in LabVIEW</i> Course manual

D. Installing the Course Software

Complete the following steps to install the course software.

1. Insert the course CD in your computer.
2. Complete the onscreen instructions to install the Exercises and Solutions files to the desired location.



Tip Folder names in angle brackets, such as <Exercises>, refer to folders on the root directory of your computer.

E. Course Goals

After completing this course you will be able to:

- Design a software architecture to be implemented in LabVIEW
- Design a consistent, organized, and usable API
- Analyze and evaluate several solutions to a problem
- Use advanced design patterns and techniques to build the components or subsystems of an architecture
- Understand the design trade-offs when selecting an advanced design pattern or technique
- Analyze, critique, and improve the architecture of a LabVIEW application

F. Course Conventions

The following conventions are used in this course manual:

»

The » symbol leads you through nested menu items and dialog box options to a final action. The sequence **Options»Settings»General** directs you to pull down the **Options** menu, select the **Settings** item, and select **General** from the last dialog box.



This icon denotes a tip, which alerts you to advisory information.



This icon denotes a note, which alerts you to important information.

bold

Bold text denotes items that you must select or click in the software, such as menu items and dialog box options. Bold text also denotes parameter names.

italic

Italic text denotes variables, emphasis, a cross-reference, or an introduction to a key concept. Italic text also denotes text that is a placeholder for a word or value that you must supply.

monospace

Text in this font denotes text or characters that you enter from the keyboard, sections of code, programming examples, and syntax examples. This font also is used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames, and extensions.