QUICK REFERENCE GUIDE

Measurement Studio™ Visual C++ Class Hierarchy Chart

National Instruments Measurement Studio delivers an interactive design approach for developing virtual instruments in Visual C++. The Measurement Studio tools integrate into the Visual C++ environment so you can use them exactly as you would native Microsoft tools. Use the Measurement Studio AppWizard to create your measurement system. The AppWizard creates a project, according to your specifications, with an included code template and the measurement tools you need to design your application. Measurement Studio includes C++ classes to provide data analysis, transfer data across the Internet, and interface with hardware—data acquisition, distributed I/O, and instrument control. Measurement Studio also includes custom-wrapped ActiveX controls to create your user interface. The link between the measurement classes and interface controls are data object classes that seamlessly encapsulate and pass data from acquisition to analysis to presentation.

Legend

- Measurement Studio Class
- MFC Class
- Abstract Base Class
- **bold** — Bold text denotes a top-level class that you frequently create.
- Multiple Inheritance
- < > — Templatized Class
- Type Definitions

322637C-01 Jun03

DataSocket

DataSocket includes a set of classes that encapsulates the NI-DataSocket Interfaces, which simplify the exchange of data between clients and servers across a wide variety of transport protocols.

- CNiDispatch
  - CNiDataSocketData
  - **CNiDataSocketServer**
- **CNiDataSocket**
- CNiDataSocketAttributeList
- CNiOpcQuality

NI-488.2

The 488.2 class library includes a set of classes that encapsulates the NI-488.2 (GPIB) interface. Use `CNi4882Device` to control IEEE-488.*x* devices, such as oscilloscopes and digital multimeters. Use `CNi4882Board` to control NI-488.2 interface boards such as PCI-GPIB+.

- CNi4882
  - **CNi4882Board**
  - **CNi4882Device**
- CNi4882Address
- CNi4882AddressList

LabVIEW RT Interface

LabVIEW Real-Time Interface includes a set of classes you use to read from and write to shared memory on a LabVIEW RT series processor board. Use this class library to pass data between LabVIEW RT VIs and an application.

- CNiLvRealTimeSharedMemLoc
  - CNiLvRealTimeSharedInt16
  - CNiLvRealTimeSharedInt32
  - CNiLvRealTimeSharedReal32
  - CNiLvRealTimeSharedReal64
  - CNiLvRealTimeSharedUInt8

Analysis

The Analysis classes include `CNiMath`, which you can use to perform signal generation, frequency and time domain analysis, windowing, digital filtering, curve fitting, statistics, waveform measurements, and linear algebra. Use the `CNiPeakDetector` class to detect peaks in waveforms. Use the `CNiIIRCascadeFilter` class to create cascading infinite impulse response filters.

- CNiIIRCascadeFilter
- CNiMath
- CNiPeakDetector

Utility

Utility includes a set of utility classes that encapsulates various interfaces such as file I/O, asynchronous timers, sound generation, and system services. All classes appear in the Utility class library.

- CStdioFile
  - CNiFile
    - CNiTemplate
- CNiThreadLocalVar< >
  - CNiThreadLocalVarBase< >
- CNiRegKey
- CNiRegKeys
- CNiRegValue
- CNiRegValues
- CNiSound
- CNiSystem
- CNiSystemTrayIcon
- CNiTimer

Microsoft Excel

Microsoft Excel includes a set of classes that encapsulates the Microsoft Excel application. Use the Microsoft Excel class library to create Excel worksheets and workbooks and open existing Excel worksheets and workbooks. You also can use this class library to create graphs, import images, and format your spreadsheets using formulas and functions.

- CNiExcelWorkbook::CNiWebOptions
- CNiInterface
  - CNiDispatch
    - CNiExcelApplication
    - CNiExcelCellRange
    - CNiExcelChart
    - CNiExcelWorkbook
    - CNiExcelWorksheet

Microsoft Word

Microsoft Word includes a set of classes that encapsulates the Microsoft Word application. You can use the Microsoft Word class library to create Word documents, open existing Word documents, add tables and images to Word documents, modify the appearance of Word documents, and close the Word application.

- CNiWebOptions
- CNiInterface
  - CNiDispatch
    - CNiWordApplication
    - CNiWordDocument
    - CNiWordImage
    - CNiWordTable

NI-VISA

VISA includes a set of classes that encapsulates the NI-VISA interface. Use `CNiVisaSession` to control an IEEE-488.2, serial, VXI, PXI, or TCP/IP device. Use `CNiVisaEvent` and derived classes to respond to VISA events such as triggers and interrupts.

- CNiVisaObject
  - CNiVisaEvent
    - CNiVisaClearEvent
    - CNiVisaExceptionEvent
    - CNiVisaGpibCicEvent
    - CNiVisaGpibListenEvent
    - CNiVisaGpibTalkEvent
    - CNiVisaIoCompletionEvent
    - CNiVisaPxiInterruptEvent
    - CNiVisaSerialAnyCharReceivedEvent
    - CNiVisaSerialBreakEvent
    - CNiVisaSerialCtsEvent
    - CNiVisaSerialDcdEvent
    - CNiVisaSerialDsrEvent
    - CNiVisaSerialRIEvent
    - CNiVisaSerialTermCharReceivedEvent
    - CNiVisaServiceRequestEvent
    - CNiVisaTriggerEvent
    - CNiVisaVxiInterruptEvent
    - CNiVisaVxiSignalEvent
    - CNiVisaVxiSysFailEvent
    - CNiVisaVxiSysResetEvent
    - CNiVisaUsbInterruptEvent
  - **CNiVisaSession**
    - CNiVisaResourceManager
- CNiVisaEventHandlerId
- CNiVisaJobId
- CNiVisaResourceList

Exception

The exception classes describe various exceptions that Measurement Studio for Visual C++ libraries generate. The exception classes are defined in various Measurement Studio Visual C++ class libraries.

- CException
  - CNiException
    - CNiExceptionT< >
      - CNiIviException
      - CNiMathException
      - CNiSystemException
      - CNiVisaException
    - CNi4882Exception
    - CNiDAQmxException
    - CNiDataInitializationException
    - CNiFieldPointConnectionException
    - CNiFieldPointInvalidArgumentException
    - CNiFieldPointNotSupportedException
    - CNiIndexException
    - CNiMatrixAccessException
    - CNiObjectNotInUsableStateException
    - CNiTimerException
    - CNiVectorAccessException
  - COleException
    - CNiOleException

FieldPoint

FieldPoint includes a set of classes that you can use to design Visual C++ applications that communicate with National Instruments FieldPoint modules.

- CNiFieldPointChannel
- CNiFieldPointEventHandlerId
- CNiFieldPointIoModule
- CNiFieldPointNetworkModule

User Interface

User Interface includes a set of classes that encapsulates the ActiveX user interface controls such as graph, button, slide, and knob. Objects that derive from `CWnd` represent the actual instance of the control on a `CDialog` or `CFormView`. Objects that derive from `COleDispatchDriver` represent interfaces to subparts of the control, such as cursors, fonts, and plots. These class definitions are in the UI and 3D Graph class libraries.

- CWnd
  - CNiControl
    - **CNiButton**
    - **CNiGraph**
    - **CNiGraph3D**
    - **CNiKnob**
    - **CNiNumEdit**
    - **CNiSlide**
- CNiInterface
  - CNiAnnotation
  - CNiAnnotations
  - CNiArrow
  - CNiAxes
  - CNiAxes3D
  - CNiAxis
  - CNiAxis3D
  - CNiBinding
  - CNiBindings
  - CNiCaption
  - CNiContour
  - CNiContours
  - CNiControlMetrics
  - CNiCursor
  - CNiCursor3D
  - CNiCursors
  - CNiCursors3D
  - CNiDispatch
    - CNiFont
    - CNiPicture
  - CNiImage
  - CNiLabels
  - CNiLabels3D
  - CNiLight
  - CNiLights
  - CNiPlot
  - CNiPlot3D
  - CNiPlots
  - CNiPlots3D
  - CNiPointer
  - CNiPointers
  - CNiShape
  - CNiTicks
  - CNiTicks3D
  - CNiValuePair
  - CNiValuePairs

NI-Reports

NI-Reports includes a set of classes that encapsulates a formatted report. This class communicates with the National Instruments NI-Reports ActiveX Automation server. The NI-Reports ActiveX Automation server provides report generation, formatting, and printing functionality.

- CNiInterface
  - CNiReport
  - CNiReportData
  - CNiReportPicture
  - CNiReportTable
  - CNiReportTableCell
  - CNiReportTableCells
  - CNiReportTableRow

Data Objects

The data objects are complex, vector, string, and matrix data types. These data objects represent a common format for exchanging data between the acquisition, analysis, and user interface portions of an application. These class definitions are in the Common class library.

- CObject
  - CNiDataObject
    - CNiComplex
      - CNiComplexT< >
        - CNiComplexInt8
        - CNiComplexUInt8
        - CNiComplexInt16
        - CNiComplexUInt16
        - CNiComplexInt32
        - CNiComplexUInt32
        - CNiComplexInt
        - CNiComplexUInt
        - CNiComplexReal32
        - CNiComplexReal64
    - CNiLockableDataObject
      - CNiVector
        - CNiVectorT< > → CNiVectorT< >
          - CNiScalarVector< >
            - CNiInt8Vector
            - CNiUInt8Vector
            - CNiInt16Vector
            - CNiUInt16Vector
            - CNiInt32Vector
            - CNiUInt32Vector
            - CNiIntVector
            - CNiUIntVector
            - CNiReal32Vector
            - **CNiReal64Vector**
          - CNiComplexVector< >
            - CNiComplexInt8Vector
            - CNiComplexUInt8Vector
            - CNiComplexInt16Vector
            - CNiComplexUInt16Vector
            - CNiComplexInt32Vector
            - CNiComplexUInt32Vector
            - CNiComplexIntVector
            - CNiComplexUIntVector
            - CNiComplexReal32Vector
            - **CNiComplexReal64Vector**
        - CNiColorVector< > → CNiVectorT< >
      - CNiMatrix
        - CNiMatrixT< > → CNiDataType< >
          - CNiScalarMatrix< >
            - CNiInt8Matrix
            - CNiUInt8Matrix
            - CNiInt16Matrix
            - CNiUInt16Matrix
            - CNiInt32Matrix
            - CNiUInt32Matrix
            - CNiIntMatrix
            - CNiUIntMatrix
            - CNiReal32Matrix
            - **CNiReal64Matrix**
          - CNiComplexMatrix< >
            - CNiComplexInt8Matrix
            - CNiComplexUInt8Matrix
            - CNiComplexInt16Matrix
            - CNiComplexUInt16Matrix
            - CNiComplexInt32Matrix
            - CNiComplexUInt32Matrix
            - CNiComplexIntMatrix
            - CNiComplexUIntMatrix
            - CNiComplexReal32Matrix
            - **CNiComplexReal64Matrix**
    - CNiGenericLockableDataObject
      - CNiGenericMatrix< >
        - CNiBoolMatrix
      - CNiGenericVector< >
        - CNiBoolVector
        - CNiStringVector
        - CNiDAQmxSwitchRelayPositionVector
- CNiDataTypeT< >
- CString
  - CNiString
- COleVariant
  - CNiVariant
- CNiColor

IVI Class Drivers

The Measurement Studio IVI class driver classes provide native Measurement Studio interfaces to IVI class-compliant instrument drivers to allow you to seamlessly integrate instrument control into Measurement Studio applications. You can use the Measurement Studio IVI class drivers to control any instrument that has an IVI class-compliant instrument driver. You must install the appropriate IVI-specific driver for each instrument that you are using. Visit `ni.com/idnet` for the latest versions of the IVI-specific drivers.

- CNiIviClassDriver
  - **CNiIviDcPwr**
  - **CNiIviDmm**
  - **CNiIviFgen**
  - **CNiIviScope**
  - **CNiIviSwtch**
- CNiIviDcPwrSubObject
  - CNiIviDcPwrOutput
  - CNiIviDcPwrTrigger
- CNiIviDmmSubObject
  - CNiIviDmmAc
  - CNiIviDmmAdvanced
  - CNiIviDmmFrequency
  - CNiIviDmmMeasurement
  - CNiIviDmmMultipoint
  - CNiIviDmmRtd
  - CNiIviDmmTemperature
  - CNiIviDmmThermistor
  - CNiIviDmmThermocouple
  - CNiIviDmmTrigger
- CNiIviDriverSubObject
  - CNiIviClassDriverIdentity
  - CNiIviClassDriverOperation
  - CNiIviDriverOperation
  - CNiIviSpecificDriverIdentity
  - CNiIviUtility
- CNiIviFgenSubObject
  - CNiIviFgenAm
  - CNiIviFgenArbitrary
  - CNiIviFgenArbitrarySequence
  - CNiIviFgenArbitraryWaveform
  - CNiIviFgenFm
  - CNiIviFgenOutput
  - CNiIviFgenStandardWaveform
  - CNiIviFgenTrigger
- CNiIviScopeSubObject
  - CNiIviScopeAcquisition
  - CNiIviScopeChannel
  - CNiIviScopeMeasurement
  - CNiIviScopeReferenceLevel
  - CNiIviScopeTrigger
  - CNiIviScopeTriggerAcLine
  - CNiIviScopeTriggerEdge
  - CNiIviScopeTriggerGlitch
  - CNiIviScopeTriggerRunt
  - CNiIviScopeTriggerTv
  - CNiIviScopeTriggerWidth
- CNiIviSwtchSubObject
  - CNiIviSwtchChannel
  - CNiIviSwtchCharacteristics
  - CNiIviSwtchPath
  - CNiIviSwtchScan

NI Modular Instrument Drivers

NI Modular Instrument Drivers includes a set of classes that encapsulates IVI specific drivers for National Instruments IVI class-compliant devices and third-party IVI class-compliant devices. When you want to use device-specific functionality that the IVI class driver does not provide, use these drivers instead of the IVI class drivers. Refer to *Measurement Studio for Visual C++ 6.0 and Visual C++ .NET Drivers for NI Modular Instruments* at `ni.com/devzone` to download the NI modular instrument drivers for Visual C++.

- CNiIviDriver
  - **CNiDmm**
  - **CNiFgen**
  - **CNiScope**
  - **CNiSwitch**
- CNiDmmSubObject
  - CNiDmmAc
  - CNiDmmAdvanced
  - CNiDmmMeasurement
  - CNiDmmMultiPoint
  - CNiTrigger
- CNiFgenSubObject
  - CNiFgenArbitrary
  - CNiFgenArbitrarySequence
  - CNiFgenArbitraryWaveform
  - CNiFgenFrequencyList
  - CNiFgenOutput
  - CNiFgenClock
  - CNiFgenStandardWaveform
  - CNiFgenTrigger
- CNiScopeSubObject
  - CNiScopeAcquisition
  - CNiScopeAcquisitionFetchOptions
  - CNiScopeAcquisitionStatus
  - CNiScopeAnalysis
  - CNiScopeAnalysisFilter
  - CNiScopeAnalysisReferenceLevel
  - CNiScopeAnalysisTimeHistogram
  - CNiScopeAnalysisVoltageHistogram
  - CNiScopeChannel
  - CNiScopeUtility
  - CNiScopeSynchronization
  - CNiScopeSyncOutputEvent
  - CNiScopeSyncTrigDelayCalibration
  - CNiScopeTiming
  - CNiScopeTimingRis
  - CNiScopeTrigger
  - CNiScopeTriggerDigital
  - CNiScopeTriggerEdge
  - CNiScopeTriggerHystersis
  - CNiScopeTriggerImmediate
  - CNiScopeTriggerSoftware
  - CNiScopeTriggerWindow
- CNiSwitchSubObject
  - CNiSwitchCalibration
  - CNiSwitchChannel
  - CNiSwitchCharacteristics
  - CNiSwitchPath
  - CNiSwitchScan
  - CNiSwitchScanAdvanced
  - CNiSwitchScanTrigger
  - CNiSwitchSingleSwitch

NI-DAQmx

NI-DAQmx includes a set of classes that you can use to communicate with and control an NI data acquisition (DAQ) device.

- CNiDAQmxAdvanceTrigger
- CNiDAQmxAIChannels
- CNiDAQmxAnalogEdgeReferenceTrigger
- CNiDAQmxAnalogEdgeStartTrigger
- CNiDAQmxAnalogLevelPauseTrigger
- **CNiDAQmxAnalogMultiChannelReader**
- **CNiDAQmxAnalogMultiChannelWriter**
- **CNiDAQmxAnalogSingleChannelReader**
- **CNiDAQmxAnalogSingleChannelWriter**
- CNiDAQmxAnalogUnscaledReader
- CNiDAQmxAnalogUnscaledWriter
- CNiDAQmxAnalogWindowPauseTrigger
- CNiDAQmxAnalogWindowReferenceTrigger
- CNiDAQmxAnalogWindowStartTrigger
- CNiDAQmxAOChannels
- CNiDAQmxAsyncOperation
- CNiDAQmxChannel
  - CNiDAQmxAIChannel
  - CNiDAQmxAOChannel
  - CNiDAQmxCIChannel
  - CNiDAQmxCOChannel
  - CNiDAQmxDIChannel
  - CNiDAQmxDOChannel
- CNiDAQmxCIChannels
- CNiDAQmxCOChannels
- **CNiDAQmxCounterReader**
- CNiDAQmxDevice
- CNiDAQmxDIChannels
- CNiDAQmxDigitalEdgeAdvanceTrigger
- CNiDAQmxDigitalEdgeReferenceTrigger
- CNiDAQmxDigitalEdgeStartTrigger
- CNiDAQmxDigitalLevelPauseTrigger
- **CNiDAQmxDigitalMultiChannelReader**
- **CNiDAQmxDigitalMultiChannelWriter**
- **CNiDAQmxDigitalSingleChannelReader**
- **CNiDAQmxDigitalSingleChannelWriter**
- CNiDAQmxDOChannels
- CNiDAQmxEventHandlerID
- CNiDAQmxExportSignals
- CNiDAQmxExternalCalibrationSession
- CNiDAQmxPauseTrigger
- CNiDAQmxReferenceTrigger
- CNiDAQmxScale
  - CNiDAQmxLinearScale
  - CNiDAQmxPolynomialScale
  - CNiDAQmxRangeMapScale
  - CNiDAQmxTableScale
- CNiDAQmxStartTrigger
- CNiDAQmxStream
- CNiDAQmxSwitch
- CNiDAQmxSwitchChannel
- CNiDAQmxSwitchScan
- **CNiDAQmxSystem**
- **CNiDAQmxTask**
- CNiDAQmxTiming
- CNiDAQmxTriggers

DataSocket™, FieldPoint™, IVI™, LabVIEW™, Measurement Studio™, National Instruments™, NI™, ni.com™, NI-488.2™, NI-DAQ™, and NI-VISA™ are trademarks or trade names of National Instruments Corporation. Product and company names mentioned herein are trademarks or trade names of their respective companies. For patents covering National Instruments products, refer to the appropriate location: **Help>Patents** in your software, the `patents.txt` file on your CD, or `ni.com/patents`.

© 2000–2003 National Instruments Corporation. All rights reserved. Printed in Ireland.

NATIONAL INSTRUMENTS™