

Beginner's Guide



Software

# Software Development Kit

---

Beginner's Guide

**Revision: 04/2020**  
Copyright © 2002 – 2020  
Campbell Scientific, Inc.



# ***Warranty***

---

Warranties of individual software products are found in their respective manuals.

# Table of Contents

---

PDF viewers: These page numbers refer to the printed version of this document. Use the PDF reader bookmarks tab for links to specific sections.

<b>1. Introduction.....</b>	<b>1</b>
1.1 Background.....	1
1.2 Advantages of Using an SDK .....	1
1.3 Overview of Current SDK Products.....	2
1.3.1 Current SDK Products .....	2
1.3.1.1 LoggerNet SDK.....	2
1.3.1.2 LNServer SDK .....	2
1.3.1.3 BMP5 Direct SDK.....	3
1.3.1.4 Java PakBus SDK.....	3
1.3.2 SDK Comparison Table.....	4
<b>2. LoggerNet and LNServer SDK .....</b>	<b>4</b>
2.1 <i>LoggerNet</i> Overview .....	4
2.2 LoggerNet and LNServer SDK Overview .....	5
2.2.1 LoggerNet Server DLL .....	5
2.2.1.1 Coralib3d.dll.....	5
2.2.1.2 Coralib3.dll.....	5
2.2.2 LoggerNet SDK Controls.....	5
2.2.3 LNServer SDK Controls .....	6
2.2.4 Registering the SDK Controls.....	7
2.3 Software Requirements .....	7
2.3.1 Required Campbell Scientific, Inc. Software.....	7
2.3.2 Development Tool Requirements .....	7
<b>3. BMP5 Direct SDK.....</b>	<b>7</b>
3.1 BMP5 Direct SDK Overview.....	7
3.2 SimplePB.dll Wrapper .....	8
<b>4. Java PakBus SDK.....</b>	<b>9</b>
4.1 Java PakBus SDK Overview .....	9
4.2 Installing the Java PakBus SDK Components.....	9
4.2.1 The <i>jpakbus.jar</i> File .....	9
4.2.2 The <i>DOC</i> Folder .....	9
4.2.3 The <i>SOURCE</i> Folder.....	9
4.3 Development Tool Requirement .....	10

## Figure

1-1. Campbell Scientific SDK products functional overview .....	2
---	---

## Tables

2-1. LoggerNet SDK Controls and Uses .....	6
2-2. LNServer SDK Controls and Uses.....	6

# ***Campbell Scientific Software Development Kit Beginner's Guide***

---

## **1. Introduction**

### **1.1 Background**

Campbell Scientific data loggers process measurements made with a wide variety of sensors and store summaries and statistics of these measurements as data. Data loggers carry out these operations based on instructions in a data logger program. Users create these data logger programs with Campbell Scientific software such as *Short Cut*, *Edlog*, and *CRBasic Editor*. Programs can be sent to data loggers via a variety of communication channels (for example, phone modem or RF).

A network of data loggers may contain anywhere from one to several hundred data loggers, each with its own set of sensors, communication links, and schedule for making measurements. Management of data logger networks is an intricate task. Campbell Scientific software such as *LoggerNet* handles the task of network management by:

- Sending user-created programs to a data logger.
- Checking computer and data logger times and synchronizing if necessary.
- Managing the schedule of data collection from each data logger.
- Storing values in the LoggerNet server data cache in tables that correspond to a given data logger and data logger program.
- Keeping log files for all communications, errors, etc.

Campbell Scientific customers have a wide variety of measurement interests—agriculture, weather, water resources, and vehicle testing, to name a few. Users often need applications that can be customized to suit their needs. The *LoggerNet* software product offers a number of general purpose clients that work with the LoggerNet server to perform the preceding tasks. However, there are times when users may need to write their own custom applications. Therefore, Campbell Scientific offers different Software Development Kits (SDK) that help developers create applications that extend or replace standard Campbell Scientific software products.

### **1.2 Advantages of Using an SDK**

When developers need to write custom applications, SDK controls play an intermediary role (FIGURE 1-1) in performing desired tasks, such as sending a program to a specific data logger or retrieving data from a data logger. One can, in principle, achieve the same results without SDK controls by directly interacting with the LoggerNet server or the data loggers through a “messaging” process, but this can be a very complex process. SDK controls

greatly simplify the communication process and provide a layer of insulation from future changes of the messaging protocol to the data logger.

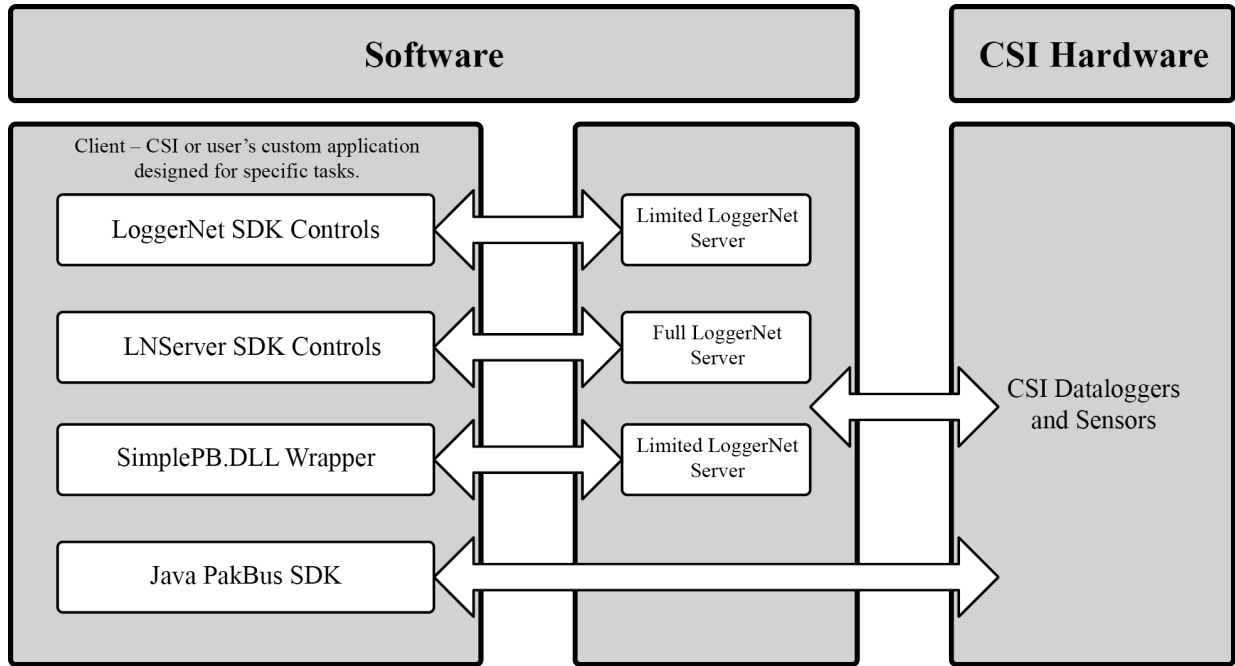


FIGURE 1-1. Campbell Scientific SDK products functional overview

SDK products simplify the task of application development tremendously by encapsulating complexities inherent in client-server communications.

### 1.3 Overview of Current SDK Products

Campbell Scientific offers several different SDK products. Each SDK product provides different advantages depending on the specific needs of the developer.

#### 1.3.1 Current SDK Products

##### 1.3.1.1 LoggerNet SDK

This software development kit contains six ActiveX® controls and a limited version of the LoggerNet Server DLL. The included LoggerNet Server DLL (coralib3d.dll) can only be loaded and accessed locally and will only communicate with a single data logger through a direct connection using RS-232 or an IP Port. However, a developer can also use this SDK to create applications that access an existing LoggerNet server installation.

The installation of this SDK includes a *Programmer's Reference* and the source code for example applications written in Visual C++® (VS2019-MFC, Microsoft Visual Studio 2019 with MFC library support), C#, and VB.NET.

##### 1.3.1.2 LNServer SDK

This software development kit contains six ActiveX controls and the unlimited version of the LoggerNet server. The included LoggerNet Server DLL (coralib3.dll) can only be loaded and accessed locally. However, a developer

can also use this SDK to create applications that access an existing LoggerNet server installation.

The installation of this SDK includes a *Programmer's Reference* and the source code for example applications written in Visual C++ (VS2019-MFC), C#, and VB.NET.

### **1.3.1.3 BMP5 Direct SDK**

This software development kit is for developers that want a simple way to create an application that communicates with any PakBus® data logger using TCP/IP Sockets or a direct RS-232 link. Intermediate communication devices that are transparent to the connection method, such as RF401 radios, are permissible. The application created using the BMP5 Direct SDK will only be able to communicate with one data logger at a time.

Starting with version 2.0, the BMP5 Direct SDK provides the developer access to a communications engine, coralib3d.dll, through the SimplePB.dll wrapper. The BMP5 Direct SDK install includes examples with source code along with documentation describing how to use the SimplePB.dll.

Previous versions of the BMP5 Direct SDK used the PakBusDLL.dll as the communication engine. The SimplePB.dll commands remain the same. Only the communication engine used by the SimplePB.dll has changed in version 2.0 of this SDK.

### **1.3.1.4 Java PakBus SDK**

This software development kit is for developers wishing to target an application host other than the Microsoft® Windows® platform. Applicable only to PakBus data loggers, this SDK requires no LoggerNet server for data logger communication and supports concurrent data logger communications. The application will function as leaf node in the PakBus network; not a PakBus router. TCP/IP Sockets provide the simplest and most common method of data logger communication with this SDK.

### 1.3.2 SDK Comparison Table

SDK Name	Installed Items	Applications	Limitations
LoggerNet SDK	<ul style="list-style-type: none"> <li>Limited LoggerNet Server (coralib3d.dll)</li> <li>Six ActiveX Control DLLs</li> <li>Examples</li> <li>Documentation</li> </ul>	Best used for applications that will be connecting to an existing LoggerNet server or for standalone applications that don't require concurrent or complex communication links. Works with all data logger types.	Communicates with only a single data logger at a time. Supports only direct TCP/IP or RS-232 connections.
LNServer SDK	<ul style="list-style-type: none"> <li>Full LoggerNet Server (coralib3.dll)</li> <li>Six ActiveX Control DLLs</li> <li>Examples</li> <li>Documentation</li> </ul>	Best used for standalone applications that require the robust communication capabilities and functionality of the full LoggerNet server. Works with all data logger types.	
BMP5 Direct SDK	<ul style="list-style-type: none"> <li>SimplePB.dll</li> <li>Limited LoggerNet Server</li> <li>Examples</li> <li>Documentation</li> </ul>	Best used for simple applications that communicate directly with PakBus data loggers only.	Communicates with only a single PakBus data logger at a time. Supports only direct TCP/IP or RS-232 connections. Limited functionality.
Java PakBus SDK	<ul style="list-style-type: none"> <li>JPakBus.jar</li> <li>Documentation</li> <li>Examples</li> </ul>	Best used for standalone applications that are required to run on platforms other than Microsoft Windows.	PakBus data loggers only. Not a PakBus router.

## 2. LoggerNet and LNServer SDK

### 2.1 LoggerNet Overview

*LoggerNet* is developed around client-server architecture. *LoggerNet's* client-server technology is based on a server that communicates with a network of data loggers via various communications technologies. The server listens for client requests, accepts the requests, and acknowledges to the client that a request has been received. The server fulfils this request and returns information to the client.

Often a client makes several requests. The server, however, processes only one request at time and in the order it was received. Once a client submits a request to the server, the client is free to do something else, knowing that its request will be processed. In other words, client requests may not get an immediate response from the server nor do the clients have to stop doing something else while waiting for an answer. This is called asynchronous communication.

## 2.2 LoggerNet and LNServer SDK Overview

The LoggerNet SDK and the LNServer SDK products differ only in the functionality of the LoggerNet Server DLL supplied with each, and the unique ActiveX® control required to start and stop the respective server. The remainder of the supplied components are common to each product and are identical in their function, operation, and use.

### 2.2.1 LoggerNet Server DLL

Fundamentally, the LoggerNet Server DLL provides the core functions of data logger communication, data collection, and storage. Additionally, it functions as the 'server' component of a client-server architecture by exposing an API for client applications. It is this interface that is abstracted by the SDK.

The LoggerNet Server DLL does not need to be registered but must be placed in the application folder, in the PATH environmental variable, or in the Windows® system directory.

Depending on the product installed, one of two versions of the LoggerNet Server DLL is included with the SDK. The versions differ in the types of telecommunication devices supported.

#### 2.2.1.1 Coralib3d.dll

The Coralib3d server is installed with the LoggerNet SDK. This limited function LoggerNet server supports only direct communications with the data logger via RS-232, USB, or TCP/IP connections. The *CsiServerDirect* control is used to start and stop this server.

#### 2.2.1.2 Coralib3.dll

The Coralib3 server is installed with the LNServer SDK. This LoggerNet Server DLL supports the same communications functionality available with the *LoggerNet* software. The *CsiServer* control is used to start and stop this server.

### 2.2.2 LoggerNet SDK Controls

The six ActiveX controls installed with the LoggerNet SDK are summarized in TABLE 2-1.



TABLE 2-1. LoggerNet SDK Controls and Uses	
LoggerNet SDK Control	Uses
CsiBrokerMap.dll	Display names of data loggers in the LoggerNet server network map Display names of tables and columns of data from data loggers
CsiDataLogger.dll	Establish a connection to a data logger via the LoggerNet server Send/Receive data logger programs Check data logger time, synchronize time with computer Retrieve data from a connected data logger
CsiDataSource.dll	Monitor data collected from a data logger
CsiCoraScript.dll	Execute <i>CoraScript</i> commands on the LoggerNet server
CsiLogMonitor.dll	Monitor LoggerNet server transaction and communication log files
CsiServerDirect.dll	Start and Stop the coralib3d.dll

### 2.2.3 LNServer SDK Controls

The six ActiveX controls installed with the LNServer SDK are summarized in TABLE 2-2.

TABLE 2-2. LNServer SDK Controls and Uses	
LNServer SDK Control	Uses
CsiBrokerMap.dll	Display names of data loggers in the LoggerNet server network map Display names of tables and columns of data from data loggers
CsiDataLogger.dll	Establish connections to data loggers via the LoggerNet server Send/Receive data logger programs Check data logger time, synchronize time with computer Retrieve data from a connected data logger
CsiDataSource.dll	Monitor data collected from a data logger
CsiCoraScript.dll	Execute <i>CoraScript</i> commands on the LoggerNet server
CsiLogMonitor.dll	Monitor LoggerNet server transaction and communication log messages
CsiServer.dll	Start and Stop the coralib3.dll

## 2.2.4 Registering the SDK Controls

The ActiveX controls must be registered on the host computer before they can be imported into a development environment or accessed by a compiled application. The controls are registered on the development host by default during the installation of the SDK. Any controls used in the development of an application must also be registered on the application host. The latter can be accomplished as part of an installation program or done manually using the *RegSvr32.exe* utility installed with the Windows® OS.

For detailed information on importing the SDK controls into a Visual Studio® development environment, refer to the *LoggerNet and LNServer SDK Programmer's Reference*.

## 2.3 Software Requirements

### 2.3.1 Required Campbell Scientific, Inc. Software

It is important to understand that it is the LoggerNet server that communicates directly with and collects data from a network of Campbell Scientific data loggers. The SDK provides the means with which an application is able to ascertain and define the structure of the network, manage the server communications with the network, access the collected data, and monitor the server operation.

A version of the LoggerNet Server DLL is included and installed with the SDK. However, an application may also be created that uses an existing installation of *LoggerNet* software. The following Campbell Scientific software products provide the necessary LoggerNet Server DLL:

- *LoggerNet* version 1.1 or higher
- *PC400* version 1.0 or higher
- *VisualWeather* 1.0 or higher

---

**NOTE** The *Allow Remote Connections* option must be enabled on the remote LoggerNet Server DLL. (Remote: meaning not residing on the application host computer.)

---

### 2.3.2 Development Tool Requirements

The SDK ActiveX controls were developed in Visual C++ MFC environment. They are best suited for the Visual C++® MFC environment. However, the SDK controls can also be used in a .NET Framework environment. Examples are included for the Visual C++ (VS2019-MFC), C#, and VB.NET programming languages.

For details on developing a .NET application using the SDK ActiveX controls, refer to the *LoggerNet and LNServer SDK Programmer's Reference*.

## 3. BMP5 Direct SDK

### 3.1 BMP5 Direct SDK Overview

The BMP5 Direct software development kit is for developers that want a simple way to create an application that communicates with any PakBus® data

logger using only RS-232 or an intermediate communication that is transparent to RS-232 such as RF400 radios. The application created using the BMP5 Direct SDK will only be able to communicate with one data logger at a time.

The BMP5 Direct SDK provides the developer with access to the communications engine, coralib3d.dll, through the SimplePB.dll wrapper. The BMP5 Direct SDK install includes examples with source code along with documentation describing how to use the SimplePB.dll.

## 3.2 SimplePB.dll Wrapper

The SimplePB.dll wrapper provides the developer with an easy interface through a communications engine to a single PakBus data logger. The following basic commands are available in the SimplePB.dll:

- OpenPort() – used to open a COM port
- ClosePort () – used to close a COM port
- OpenIPPort() – used to open an IP port
- CloseIPPort() – used to close an IP port
- GetClock() – used to query a data logger for its date and time
- SetClock() – used to set a data logger date and time
- GetValue() – used to query a data logger for a value or group of values
- SetValue() – used to set a field value in a data logger
- GetData() – used to query data from a data logger table
- GetDataHeader() – used to get just the header information for a specific table in the data logger
- GetCommaData() – used to get just the CSV (comma-separated values) data from a specific table in the data logger
- File\_Send() – used to send a program file to a data logger
- GetAddress() – used to query the PakBus address of a data logger
- GetStatus() – used to query the current status of a data logger
- GetTableNames() – used to query a data logger for current table names and numbers
- GetDLLVersion() – used to return the version of SimplePB.dll currently in use

- `GetLastResults()` – used to display the results of a previous command that currently exist in memory as a string so memory pointers do not need to be managed
- `FileControl` – used to manage programs and files on the data logger
- `SetSecurity()` – used to set the security code used to communicate with the data logger
- `GetTableRecordsCount()` – used to determine the number of records that are available for collection from the specified table

## 4. Java PakBus SDK

### 4.1 Java PakBus SDK Overview

The Java PakBus® SDK is a simple API that can be used to write Java based applications that can communicate with Campbell Scientific data loggers using PakBus protocol. An application built using this API should be able to function in any PakBus network and communicate with any data logger in that network. The application will act as a PakBus leaf node. This means that it will not send or receive routing information with the exception of messages used to confirm neighbor links.

### 4.2 Installing the Java PakBus SDK Components

There is no install program required; simply copy the SDK components to a location on the development host. The SDK comprises a *jpakubs.jar* file, a *doc* folder, and a *source* folder.

#### 4.2.1 The *jpakbus.jar* File

This Java archive file contains the compiled .CLASS files that will be accessed by the application at runtime. This file should be distributed with the application and placed in the application folder or have its location added to the CLASSPATH environment variable.

#### 4.2.2 The *DOC* Folder

This folder contains the HTML formatted documentation for the SDK. To access the documentation, simply point a browser to the *index.html* file in the root of this folder.

As well as information and examples for using the SDK, the documentation provides detail descriptions of the packages, classes and interface that comprise the SDK. The *Java PakBus Software Development Kit* is an abridged version of this documentation and can be downloaded from the Campbell Scientific website.

#### 4.2.3 The *SOURCE* Folder

This folder contains the source code files for the SDK examples as well as all of the classes and interfaces used by SDK. These files are included to aid in the understanding of the Java PakBus® SDK and cannot be redistributed, modified, or used as the basis for some other SDK/API product.

### 4.3 Development Tool Requirement

As a minimum, the *Java SE Software Development Kit 6* (JDK) or later will be required for application development. A comparable version of the *Java Virtual Machine* (JVM) must be installed on the application host.



## Global Sales & Support Network

A worldwide network to help meet your needs



### Campbell Scientific regional offices

#### Australia

**Location:** Garbutt, QLD Australia  
**Phone:** 61.7.4401.7700  
**Email:** [info@campbellsci.com.au](mailto:info@campbellsci.com.au)  
**Website:** [www.campbellsci.com.au](http://www.campbellsci.com.au)

#### Brazil

**Location:** São Paulo, SP Brazil  
**Phone:** 11.3732.3399  
**Email:** [vendas@campbellsci.com.br](mailto:vendas@campbellsci.com.br)  
**Website:** [www.campbellsci.com.br](http://www.campbellsci.com.br)

#### Canada

**Location:** Edmonton, AB Canada  
**Phone:** 780.454.2505  
**Email:** [dataloggers@campbellsci.ca](mailto:dataloggers@campbellsci.ca)  
**Website:** [www.campbellsci.ca](http://www.campbellsci.ca)

#### China

**Location:** Beijing, P. R. China  
**Phone:** 86.10.6561.0080  
**Email:** [info@campbellsci.com.cn](mailto:info@campbellsci.com.cn)  
**Website:** [www.campbellsci.com.cn](http://www.campbellsci.com.cn)

#### Costa Rica

**Location:** San Pedro, Costa Rica  
**Phone:** 506.2280.1564  
**Email:** [info@campbellsci.cc](mailto:info@campbellsci.cc)  
**Website:** [www.campbellsci.cc](http://www.campbellsci.cc)

#### France

**Location:** Vincennes, France  
**Phone:** 0033.0.1.56.45.15.20  
**Email:** [info@campbellsci.fr](mailto:info@campbellsci.fr)  
**Website:** [www.campbellsci.fr](http://www.campbellsci.fr)

#### Germany

**Location:** Bremen, Germany  
**Phone:** 49.0.421.460974.0  
**Email:** [info@campbellsci.de](mailto:info@campbellsci.de)  
**Website:** [www.campbellsci.de](http://www.campbellsci.de)

#### India

**Location:** New Delhi, DL India  
**Phone:** 91.11.46500481.482  
**Email:** [info@campbellsci.in](mailto:info@campbellsci.in)  
**Website:** [www.campbellsci.in](http://www.campbellsci.in)

#### South Africa

**Location:** Stellenbosch, South Africa  
**Phone:** 27.21.8809960  
**Email:** [sales@campbellsci.co.za](mailto:sales@campbellsci.co.za)  
**Website:** [www.campbellsci.co.za](http://www.campbellsci.co.za)

#### Spain

**Location:** Barcelona, Spain  
**Phone:** 34.93.2323938  
**Email:** [info@campbellsci.es](mailto:info@campbellsci.es)  
**Website:** [www.campbellsci.es](http://www.campbellsci.es)

#### Thailand

**Location:** Bangkok, Thailand  
**Phone:** 66.2.719.3399  
**Email:** [info@campbellsci.asia](mailto:info@campbellsci.asia)  
**Website:** [www.campbellsci.asia](http://www.campbellsci.asia)

#### UK

**Location:** Shepshed, Loughborough, UK  
**Phone:** 44.0.1509.601141  
**Email:** [sales@campbellsci.co.uk](mailto:sales@campbellsci.co.uk)  
**Website:** [www.campbellsci.co.uk](http://www.campbellsci.co.uk)

#### USA

**Location:** Logan, UT USA  
**Phone:** 435.227.9120  
**Email:** [info@campbellsci.com](mailto:info@campbellsci.com)  
**Website:** [www.campbellsci.com](http://www.campbellsci.com)