Lightspeed Blackbox Developers Kit (BDK)

Getting Started Document

Version 1.02
July 8, 2011

To obtain additional copies of this document, contact:

Lightspeed Financial, Inc.
148 Madison Avenue, 9th Floor
New York, NY 10016
646-393-4815

Revision History

The table below records the revision history of this document:

Revision	Date	Changes Made
1.00	June 18, 2011	Initial Document
1.01	July 1, 2011	Fixed a typo in the IP address of the market data simulator and Gateway simulator. Added text to discuss the example configuration parameters.
1.02	July 8, 2011	Added a section to discuss ARCA data servers and how to configure the BDK to use only one ARCA data server.

Contents

1. Introduction

This document is intended for software developers that will be using Lightspeed's Blackbox Developers Kit (BDK) to develop their automated trading system (blackbox).

This document provides a process that can be used to start development using the BDK.

NOTE: this document is intended to provide helpful information for getting started. The customer is not required to follow this process.

The following is discussed in this document.

- Accessing the BDK documentation
- Obtaining the software components
- Setting up the development environment
- Required directory structure
- Running the blackbox for the first time
- Making a connection from the console to the blackbox
- Accessing the Market Data simulators and Gateway simulator
- Registering for market data and connecting to the Gateway
- Console authentication and changing the password
- Running multiple instances of the blackbox
- Learning the BDK
- ARCA Market Data Change

1.1. Accessing the BDK Documents

The BDK documents can be downloaded from Lightspeed's website. The customer must first login to the Lightspeed website. Login credentials (username and password) can be obtained from Lightspeed's support team.

To access the BDK documents, first start a web browser and bring up Lightspeed's home page (www.lightspeed.com)

Click the Login Button in the upper right corner of the window and enter your username and password when the logon window appears.

Onced logged in, the "My Account" window will appear. On the left side of the window is a list of links. Click the "Blackbox Develoers Kit" link. A web page will appear that contains a link for each of the BDK documents. The following documents can be downloaded:

- BDK Overview
- BDK API specification
- Console Users Guide
- Configuration Guide
- How-To Build the Linux Blackbox
- How-To Build the Windows Blackbox
- How-To Build the Console
- How-To Install and Use SSH
- Sample Trading Strategy Design Document
- Getting Started

1.2. Obtaining the Software Components

Contact the BDK support team to obtain the BDK software components. The BDK contains two types of software components:

- The software components needed to develop the blackbox
- The console software

The blackbox software components consist of the BDK library, the source code for a sample trading strategy, and sample configuration files. The console software components consist of a console application and source code files for the console application.

When you contact the BDK support team to obtain the BDK software components, you will be asked what development platform you will be using. This is necessary so that the support team can send you the BDK library for your platform. The following platforms are supported:

- Linux 64 bit
- Linux 32 bit
- Windows 32 bit compiled with Visual C++ 6.0
- Windows 32 bit compiled with Visual Studio 2008
- Windows 64 bit compiled with Visual Studio 2008

The BDK support team will send you 3 emails.

1.2.1. Email 1:

The first email will contain the BDK library, the source code files for a sample trading strategy, and sample configuration files. The following files will be attached to this email:

File name	Description
Liblssdk.a (Linux) lssdk.lib.x (Windows)	This file is the BDK library. Customers using the Windows platform will have to change the name of the file. For security reasons, many email systems will not allow a .lib file to be attached to an email. Windows customers will need to rename the file to lssdk.lib.
customer_sample.c	Source file for the sample trading strategy
customer_sample.h	Header file for the sample trading strategy
platform.h	Header file that defines which platform to build (linux or windows)
sdk_proto.h	Contains prototypes for all the BDK defined function calls
sdk_cfg.1	Configuration file for the BDK Library. The BDK document entitled "Configuration Guide" describes the contents of this file. The document is available on the Lightspeed website.
Symbol.list.1	List of symbols the BDK Library can trade.
Symbol.conf.1	List of symbols and trading parameters for each symbol that the sample trading strategy will trade.
Makefile	Linux customers only. The makefile used to build the Black Box Trading System.

1.2.2. Email 2:

The second email will contain the console application and a file called password_file. A password is used to make sure other customer consoles cannot connect to your black box. The following files will be attached to this email:

File name	Description
Botcon.ex	This file is the console application. For security reasons, many email systems will not accept an attachment with the .exe extension.

	Customers will need to rename the file to botcon.exe.
password_file	This file contains the customer's password. Refer to the section below that discusses console authentication and changing the password.

1.2.3. Email 3:

The third email will contain the console source code files. Most customers do not modify the console and don't need the source code. The standard console application contains a command line interface that allows customers to send commands to their blackbox. This is typically done to change trading parameters. However, some customers would like to build a better user interface for changing their trading parameters, so the console source code is provided. There is a document on Lightspeed's website that describes how to build (compile and link) the console using Visual C++ 6.0 and Visual Studio 2008. The following files will be attached to this email:

File name	Description
botcon.cpp	Console source code file
botcon.h	Console source code file
botcon.rc	Console source code file
cursor1.cur	Console source code file
cursor2r.cur	Console source code file
resource.h	Console source code file
soundalert.cpp	Console source code file
StdAfx.h	Console source code file
tradewin.cpp	Console source code file
tradewin.h	Console source code file
winscroll.cpp	Console source code file
winscroll.h	Console source code file

1.3. Setting up the development environment

Once the customer has obtained the software components, they should set up their development tools and ensure they can build (compile and link) the blackbox using the BDK library and the sample trading strategy source code. When the sample trading strategy source code is compiled and linked with the BDK library, the result is a fully functioning blackbox.

Instructions for setting up the development environment and building the blackbox are available on Lightspeed's website.

Linux customers should read the "How-To Build the Linux Blackbox"

Windows customer should read the "How-To Build the Windows Blackbox"

1.4. Required Directory Structure

After building the blackbox, the customer should make sure the blackbox will run. Before running the blackbox, the proper directories (folders) must be set up and configuration files must be placed in the correct directory.

For this example we will assume the blackbox is to be run from the following directory:

Linux: /usr/BlackboxDev/bbox

Windows: C:\ BlackboxDev \bbox

A directory called "data" must be created at the same level as the directory where the blackbox will be run. For this example, the following directory is required.

Linux: /usr/BlackboxDev/data

Windows: C:\ BlackboxDev \data

The data directory is used to store the BDK log files. The BDK library creates two log files: One is referred to as the log file and the other is referred to as the raw file. BDK log files start with the date in YYYYMMDD format, followed by .log or .raw. For example,

20110618.log.1

20110618.raw.1

The reason for the .1 at the end of the log files is discussed later in this document in the section that discusses running multiple instances of the blackbox.

1.4.1. Configuration Files

The following example configuration file provides with the BDK.

Configuration File Name	Description
sdk_cfg.1	Main configuration file for the BDK Library. Refer to the Configuration Guide for a detailed description of each configuration parameter.
symbol.list.1	List of symbols the BDK Library can trade.
symbol.conf.1	List of symbols and trading parameters for each symbol that the sample trading application will trade.

These three configuration files must be placed in the same directory as the blackbox application.

1.5. Running the Blackbox

To run the blackbox from a Linux or Windows command prompt window, simply enter the name of the blackbox application followed by 1. For example, assume the name of the blackbox is bot.

Linux: bot 1

Windows: bot.exe 1

Note that a 1 is passed as a command line parameter. The reason for the parameter is discussed later in this document in the section that discusses running multiple instances of the blackbox.

To run the blackbox in the backgourd add & to the end of the command as follows:

Linux; bot 1 &

Windows: bot.exe 1 &

1.6. Making a Connection from the Console to the Blackbox

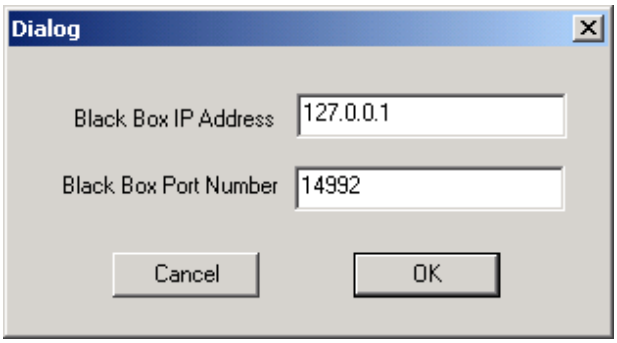
Once the blackbox is running, the next step is to make sure the console can connect to the blackbox.

The console application is a Windows program so it must be run on a windows computer.

Create a directory were the console application will be stored. Copy the console application and password_file to this directory. Note: the password_file must be in the same directory as the console application.

The console can be run form a command prompt window or by double clicking it in Explorer or MyComputer. There a many ways to start a windows application and all will work to start the console.

When the console is started, the main console window will appear. To make a connection to the blackbox, click the button on the main console window called “Connect to BBOX”. The following dialog box appears:



Enter the IP address of the computer that is running the blackbox. If you are a windows customer and are running the console on the same computer as the blackbox, then enter the IP address 127.0.0.1 (local computer's IP address).

Next, enter the port number that the blackbox will accept console connections on. The port number that the blackbox will accept console connections on is defined in the blackbox configuration file (sdk_cfg.1). There is a parameter in the blackbox configuration file called "console-port". For example, the configuration file distributed with the BDK will contain the following:

console-port = 14992

Once the IP address and port number have been entered, click OK. The console will attempt to connect to the blackbox.

When the console successfully makes a connection to the blackbox, the connection status on the main console window will look as follows:

Connection to BBOX: Up

1.7. Accessing the Market Data Simulator and Gateway Simulator

Lightspeed provides market data simulators and a Gateway simulator to allow the customer to test their blackbox before co-locating a computer in Lightspeeds data center. The simulators are located in Lightspeed's data center so a VPN connection will need to be established between the customer's computer and Lightspeed's data center.

Contact the BDK support team to request a VPN account. The BDK support team will create the VPN account and provide the customer with their VPN username and password as well as instructions for installing and running the VPN client software on their computer.

The customer can use ping to determine if their VPN is working and that they have connectivity to the simulator. Run ping as follows to test connectivity.

ping 192.168.25.240

1.7.1. Market Data Simulators

The blackbox configuration file (sdk_cfg.1) specifies the IP address and port number for all market data servers. The configuration file distributed with the BDK is set up to access the simulators. The following IP addresses and port numbers are specified in the configuration file to access the market data simulators.

Simulator	IP/port number
Quote Server	192.168.25.240/21001

INET Book Server	192.168.25.240/11001
ARCA Book Server	192.168.25.240/11002
BATS Book Server	192.168.25.240/11003
EDGX Book Server	192.168.25.240/11006
EDGA Book Server	192.168.25.240/11007

When the blackbox is started the BDK library will attempt to connect to all servers. If the VPN is working correctly, the blackbox should connect to the market data simulators.

Customers may notice that the blackbox cannot connect to the Futures and Indices server and the Short Availability server. This is because simulators do not exist for these servers. The customer will not be able to connect to these serves until they are co-located. The blackbox will attempt to connect and fail every few seconds. To stop the BDK library from attempting to connect to the Futures and Indices server and the Short Availability server, comment out the following parameter in the configuration file. Making the first character of the record a # will make the record a comment.

```
#indices-mds-ip              = 10.210.101.241
#indices-mds-port            = 11122
#sa-svr-ip                   = 10.210.101.171
#sa-svr-port                 = 3000
```

1.7.2. Gateway Simulator

Lightspeed's order processing server is referred to as the "Gateway". The Gateway simulator allows customers to test their software before connection to a "live" Gateway. The Gateway simulator will process orders differently based on the first letter of the symbol. For example, stock symbols that start with the letter 'A' will always fill regardless of the price. Stock symbols that start with the letter 'B' will never fill and can only be canceled. Other letters will cause the Gateway simulator to fill an order with two execution reports, or fill half and cancel the remainder. Refer to the Gateway Simulator Users Guide for a description of all behaviors.

The blackbox configuration file (sdk_cfg.1) specifies the IP address and port number for the Gateway. The configuration file distributed with the BDK is set up to access the Gateway simulator. The following IP address and port number are specified in the configuration file to access the Gateway simulator.

Simulator	IP/port number
Gateway Simulator	192.168.25.240/31001

1.8. Registering for Market Data

When the BDK library is started it will connect to the Market Data servers. However, it will not register to receive data. Registering to receive data must be done manually using the console. The console displays the status of all Market Data Servers (MDS). Each market data source is represented as a single letter or two letters. The following letters are used:

Q – Quote Server
I – INET Book Server
T – BATS Book Server
G – Direct Edge (EDGX) Book Server
H – Direct Edge (EDGA) Book Server
Ao – Arca OTC Book Server
Al – ARCA Listed Book Server (not currently used)
Ae – ARCA ETF Book Server(not currently used)
S – Short Availability Server
X – Futures Market Data Server (Futures data and Indices)
M – Imbalance data

The letters that represent data sources are color coded to convey the status of the data source. The following colors are used:

Red – The blackbox is not connected to the data source.
Blue – The blackbox is connected to the data source, but has not registered to receive data.
Green – The blackbox is connected to the data source and is registered to receive data.

1.8.1. Registering for Market Data

Do the following to register to receive market data:

- Click the “Market Data Configuration” button on the console window.
- The market data configuration dialog box will appear. Click the “Register/Deregister Data” button.
- A dialog box will appear that has a button for each data venue. Click the venue that you want to receive data from, or click the “Register All” button to receive data from all venues.
- Check to see that the letter that represents the status for venue is green.

1.9. Connecting to the Gateway

When the BDK library is started it will NOT automatically connect to the Gateway. Connecting to the Gateway must be done manually using the console. The Order Type Status displayed on the

console is used to inform the operator of the status of each Gateway order type. Each order type is represented as a single letter. The following letters are used

I – INET
A– ARCA
R – RASH
Y - NYSE
E – AMEX
T – BATS
H– EDGA
G – EDGX
O – RASH
J – Jefferies
P – BATY

The letters that represent the order types are color coded to convey the status of the order type. The following colors are used:

Red – The order type is disabled.
Green – The order type is enabled.

If the blackbox has not connected to the Gateway, then all Order Type Status indicators will be red. After connecting to the Gateway, the Order Type Status indicators will be green.

NOTE: The Gateway simulator does not support all order types. After successfully connecting to the Gateway simulator the Order Type Status will appear as follows:

Order Type Status: I A R Y E T H G O J P

1.9.1. Connecting to the Gateway

Do the following to connect to the Gateway:

- Click the "Connect to Gateway" button on the main console window.
- A confirmation dialog box will appear. Click "Yes" to connect to the Gateway.
- Check to see that the Order Type Status is displayed as expected. When connecting to the Gateway simulator, the Order Type Status should appear as shown above.

1.10. Console Authentication and Changing the Password

Many customers use the BDK library and the console application. It is important to ensure that one customer cannot connect to another customer's blackbox using the console application. A password is used to ensure that only authorized consoles can connect to a blackbox.

When the console makes a connection to the blackbox, it will send a console login message as the first message. If the blackbox does not receive a console login message as the first message it will drop the console connection.

The login message contains a password. The console will obtain the password from the file called password_file. For example, the password_file distributed with the BDK contains the string "abc123def".

When the blackbox receives the login message from the console, it will compare the password in the login message with the password that it expects from valid consoles. The password the blackbox expects in defined in the BDK configuration file (sdk_cfg.1). The parameter "authentication-string" contains the expected password. For example, the configuration file distributed with the BDK contains the following:

Authentication-string = abc123def

1.10.1. Changing the Password

Do the following to change the password:

- choose a password
- Edit the file password_file on the computer that runs the console and replace "abc123def" with the new password.
- Edit the BDK configuration file (sdk_cfg.1) on the computer that runs the blackbox and change the authentication-string parameter to the new password.
- The console and blackbox must be restarted for the new password to take effect.

1.10.2. Changing the Console Port Number

To increase security, the port number that the blackbox accepts console connections on can be changed. The port number that the blackbox accepts console connections on is defined in the BDK configuration file (sdk_cfg.1). There is a parameter in the configuration file called "console-port". For example, the configuration file distributed with the BDK will contain the following:

console-port = 14992

By changing both the password and port number, a hacker would have to guess the port number the blackbox accepts connects on and the password.

Note, the password carried in the console login message is encrypted to prevent a hacker with a network sniffer from learning the password.

1.11. Running Multiple Instances of the Blackbox

As mentioned above, the blackbox requires one command line parameter be passed to it when started. If the blackbox is run from a command prompt, then it would be started as follows (assume the name of the blackbox application is bot).

bot 1

Multiple instances of the blackbox can be run on the same computer. The parameter is used by a blackbox to determine which configuration files to use.

For example, lets assume two instances of the blackbox are to be run. If two instances are to be run, then 6 configuration files are needed.

sdk_cfg.1
symbol.list.1
symbol.conf.1

sdk_cfg.2
symbol.list.2
symbol.conf.2

Two instances of the blackbox are started as follows:

bot 1

bot 2

The first instance would use the following configuration files because they end in 1 (the parameter passed when starting the blackbox):

sdk_cfg.1
symbol.list.1
symbol.conf.1

The second instance would use the following configuration files because they end in 2 (the parameter passed when starting the blackbox):

sdk_cfg.2
symbol.list.2
symbol.conf.2

1.12. Learning the BDK

Once the Blackbox development tools are set up, the Blackbox and console can be run successfully, and the Blackbox can access the market data simulators and the Gateway simulator, it's time to focus on learning the BDK.

Understanding the sample trading strategy source code is a good way to learn how to use the BDK. The following documents should be read:

- Sample trading strategy design document
- BDK API specification
- Console user Guide

Once the documents have been read, examining the sample trading strategy source code is the best way to learn the BDK.

1.13. ARCA Market Data Change

How ARCA market data is disseminated to clients has changes since the BDK was first developed. Some background information will help eliminate some of the confusion.

When the BDK was first developed there was three ARCA market data servers. One server provided ARCA market data for OTC stock. A second server provided data for Listed stocks, and a third server provided data for ETFs. The BDK Library was required to make three connections to obtain all ARCA market data.

The BDK Library needed to know which stocks were OTC, Listed and ETF stocks so that it could register to receive data with the correct server. For example, MSFT (MicroSoft) is an OTC stock and the BDK Library needed to register to receive data with the ARCA OTC server, not the Listed or ETF server. Information was put in the symbol.list.x file to tell the BDK Library which ARCA server to register to received data from. The following is an example of three stock symbols in the symbol.list file.

```
MSFT,O
IBM,L
SPY,E
```

The character following the symbol was used to inform the BDK Library which ARCA server to register to received data from (O=OTC, L=Listed, E=ETF).

Because the BDK Library made connections to three ARCA market data server, three icons were displayed on the console to indicate the status of each connection. Customer would see the following displayed on the console:

MDS Status: **Q I T G H X M**
Ao Al Ae S

Each market data source is represented as either a single letter or two letters. The letters that represent data sources are color coded to convey the status of the data source. Red indicates the BDK Library is not connected to the data source. Blue indicates the BDK Library is connected but has not registered to receive data. Green indicates the BDK Library is connected and has registered to receive data.

Ao is used to indicate the status of the ARCA OTC connection. Al is used to indicate the status of the ARCA Listed connection. Ae is used to indicate the status of the ARCA ETF connection.

In late 2010 Lightspeed introduce a new market data server. The new ARCA server disseminated data for all symbols on one connection instead of three. The BDK Library still supports the ability to connect to three ARCA data servers even though it is not needed with the new ARCA data server. Support for three ARCA data servers was left in the BDK in case the amount of data increases in the future thus requiring more than one connection to disseminate all the data.

To use only one ARCA data server, the BDK Library needs to be configured to use only the OTC server, and not the Listed and ETF server. Using only one ARCA data server requires the customer to make 2 configuration changes:

1) All symbols in the symbol.list.x file should have the letter 'O' following the symbol. This will inform the BDK Library to register to receive data for all symbols with the OTC sever. The following illustrates:

   MSFT,O
   IBM,O
   SPY,O

2) The BDK configuration file (sdk_cfg.x) should specify the IP address and port number for only the ARCA OTC server. The parameters used to specify the IP addresses and port numbers for the Listed and ETF servers should be commented out or removed from the BDK configuration file. The following is an example of the BDK configuration file where the IP addresses and port numbers for the Listed and ETF servers are commented out. I # in the first character indicates a comment.

```
arca-otc-mds-ip          = 192.168.25.240
arca-otc-mds-port        = 11002
#arca-listed-mds-ip      = 192.168.25.240
#arca-listed-mds-port    = 11003
#arca-etf-mds-ip         = 192.168.25.240
#arca-etf-mds-port       = 11004
```

When the BDK Library is configure to connect to only one ARCA data server, the connection status displayed on the console should appears as follows:

MDS Status: **Q I T G H X M**
Ao Al Ae S

When the BDK Library is configured to connect to only one server (the OTC server), then the MDS Status for the Listed and ETF server should be red indicating that no connections exists.

1.14. Contact Information:

1.14.1. BDK Team Contact Information:

Phone: 916-315-9577

Email: sdean@lightspeed.com

Hours: 7:00am to 4:00pm PST

1.14.2. Technical Account Management (TAM) Team Contact Information

Phone: 888-577-2123

Email: TAM@lightspeed.com