Lightspeed Black Box SDK

Overview Document

Version 1.02
August 26, 2010

To obtain additional copies of this document, contact:

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

About this Document

This document describes Lightspeed's Black Box Software Development Kit (SDK). It is intended to give the reader a high level understanding of the following:

- Lightspeed's Black Box Trading Environment

- How to develop a Black Box Trading System that works within Lightspeed's Black Box Trading Environment.

- Lightspeed's Black Box Software Development Kit (SDK).

Who Should Read this Document

This document is intended for anyone who is interested in developing a Black Box Trading System.

Terminology

Throughout this document the following terms are used:

Term	Definition
SDK	Software Developers Kit. A set of tool that allow Lightspeed customers to quickly develop Black Box Trading Systems.
SDK Library	A compiled form of the SDK software components common to all Black Box Trading Systems.
Source Code	The mechanism used to specify the actions to be performed by a computer. Source code is in human readable form and must be compiled before it can run on a computer
Console	A software application used to configure, monitor and control a Black Box Trading System.
Remote Console	A console that is running on a computer that is physically located at a different location than the computer the Black Box Trading System application is running on.
Lightspeed's Black Box Trading Environment	A set of services provided by Lightspeed that allow customers to develop Black Box Trading Systems. These services include: market data service, Order entry service, risk management, short availability service, etc..

Black Box Trading System	A software application developed for the purpose of trading for a profit with little human intervention. Throughout this document the term "Black Box Trading System" is used to refer to an application that is created by compiling the customer's software and liking it with the SDK Library.
Trading Strategy Logic	A component of a Black Box Trading System that will analyze market conditions, and when conditions are met, will send buy and sell orders to the market.
Position Management Logic	A component of a Black Box Trading System that will determine when positions will be covered.
Alternative Trading System	A SEC approved, non-exchange trading venue. Refer to Rule 300 (a) of the SEC's Regulation ATS for the legal definition.
API	Application Program Interface. A set of routine, protocols and tools for building software applications.
Lightspeed Gateway	Lightspeed's Order Entry Server. Black Box Trading Systems connect to the Lightspeed Gateway to send orders to the market and receive status about orders form the market.
Short Availability	Information describing whether a security is eligible for short sale, and the number of shares that can be sold short.
Function Call	Refers to the customer's software calling a function provided by the SDK Library.
Call-Back Function	Refers to the SDK library calling a function provided by the customer's software.
Customer's software	Refers to software developed by the customer that is compiled with the SDK Library to create a Black Box Trading System.
Console operator or Operator	The person responsible for starting, stopping, configuring and monitoring the Black Box Trading System.

Revision History

The table below records the revision history of this document:

Revision	Date	Changes Made
1.00	January 25 2010	Initial Document
1.01	August 3, 2010	Updated thae Hardware Requirements section.
1.02	August 26, 2010	Added reference to EDGA now that EDGA suppot was added to the SDK.

Contents

1. Introduction

This document describes Lightspeed's Black Box Software Development Kit (SDK). It is intended to give the reader a high level understanding of the following:

- Lightspeed's Black Box Trading Environment

- How to develop a Black Box Trading System that works within Lightspeed's Black Box Trading Environment.

- Lightspeed's Black Box Software Development Kit (SDK). .

Section 1 is the introduction.

Section 2 provides an overview of Lightspeed's Black Box Trading Environment. It describes the services provided by Lightspeed to allow customer to build Black Box Trading Systems.

Section 3 provides an overview of a Black Box Trading System. It discusses the issues that developers face when building a Black Box Trading System.

Section 4 describes Lightspeed's Black Box Software Development Kit (SDK). The SDK provides a set of tools that allow customers to quickly build a Black Box Trading System that works within Lightspeed's Black Box Trading Environment.

Section 5 discusses the hardware and software required to deploy a Black Box Trading System.

2. Lightspeed's Black Box Trading Environment

Lightspeed's Black Box Trading Environment provides all the services needed by a customer to develop a Black Box Trading System. Lightspeed's Black Box Trading Environment provides the following services:

- Market Data Service. The Market Data Service provides access to market data from all US equity markets, ECN's, and Alternative Trading Systems.

- Order Entry Service. The Order Entry Service provides the ability to route orders to all US equity markets, ECN's, and Alternative Trading Systems. Lightspeed's Order Entry Server is referred to as the Lightspeed Gateway.

- Risk management Service. The Lightspeed Gateway will monitor risk associated with buying power, maximum order size, maximum position size, P&L loss threshold, short checking and odd lot allowance.

- Short Availability Service. The Short Availability Service provides access to information describing which securities are eligible for short sale.

- Clearing Service. The Clearing Service provides an internally managed database that operates in conjunction with the clearing firm to perform trade-level bookkeeping and generate all required regulatory reports (i.e. OATS).

The Market Data Service, Order Entry Service, and Short Availability Service are implemented as software components that provide the service. These software components are implemented and maintained by Lightspeed. These software components provide an API to allow customer developed Black Box Trading Systems to access these services. The API is a message based protocol, where messages are exchanged between the service provide and the customer's Black Box Trading System using a TCP/IP connection.

Lightspeed provides documentation that describes the messages and the method of communication, TCP/IP. From these documents the customer can develop software to access the services.

The following figure illustrates the relationship between the customer's Black Box Trading System and the services that make up Lightspeed's Black Box Trading Environment.
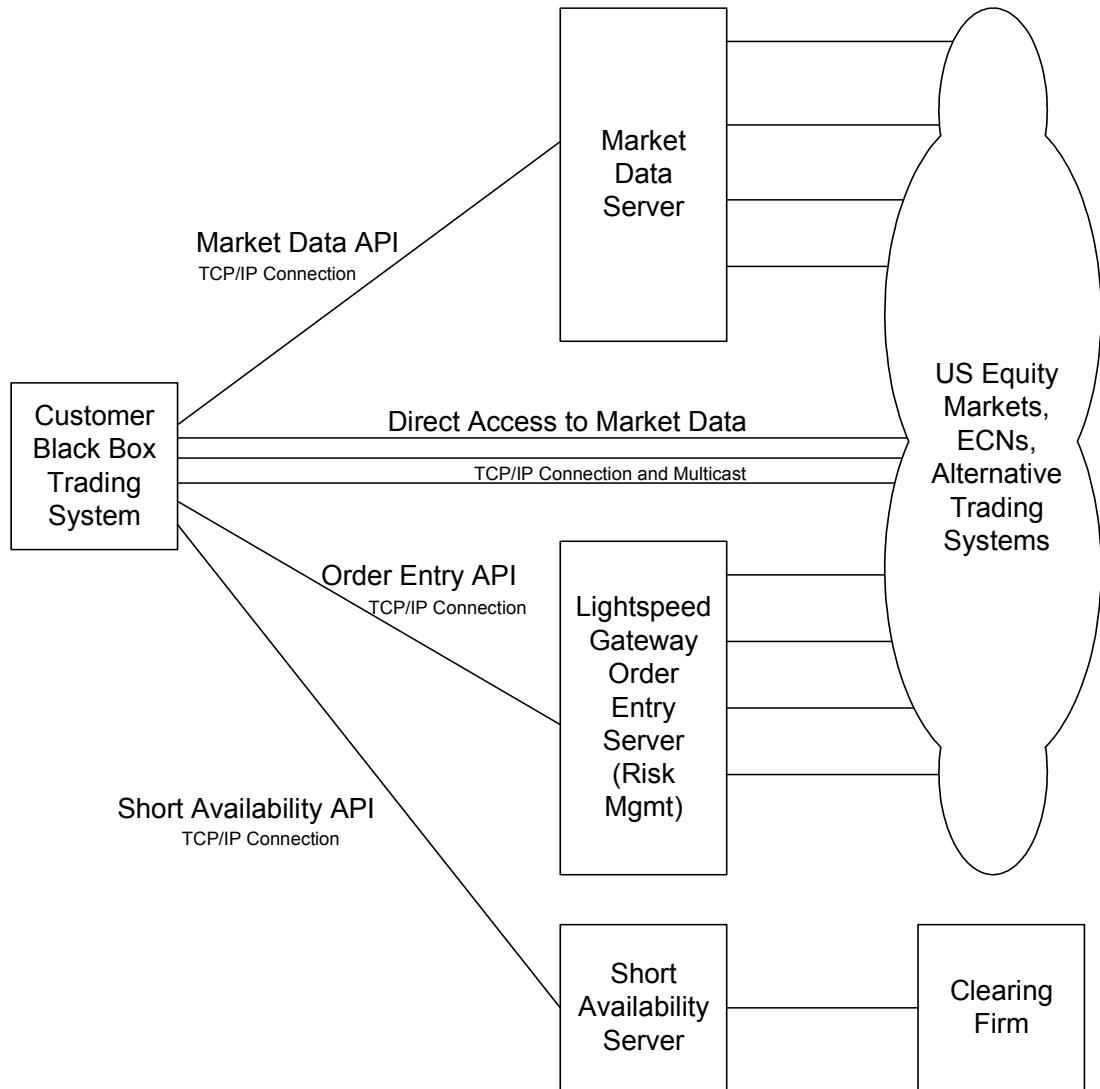
Figure 1: Lightspeed Services

As shown in figure 1, the customer's Black Box Trading System will make TCP/IP connections to Lightspeed servers to obtain service. The US equity markets, ECNs, and Alternative Trading Systems are collectively shown as a cloud in the figure. The customer's Black Box Trading Systems can get market data directly from the data providers. This is illustrated as a line from the customer's Black Box Trading Systems to the cloud and labeled "Direct Access to Market Data". Lightspeed make the market data available to customer on the network. The data is in the format provided by the data source. In some cases the customer's Black Box Trading Systems must make a TCP/IP connection to the data source. In other cases the data provider will disseminate the data via multicast, in which case the customer's Black Box Trading Systems must join the multicast group.

3. Black Box Trading System

A Black Box Trading System is a software application developed by a Lightspeed customer for the purpose of trading for a profit with little human intervention. The Black Box Trading System will analysis market data and when the proper conditions are met, it will send buy or sell orders to the market. A typical Black Box Trading System will contain the following components:

- Code to connect to market data sources and obtain market data.

- Code to build an aggregated view of the market data.

- Code to connect to the Lightspeed Gateway and enter orders and receive status about orders.

- Code to analysis market conditions and determine when buy and sell orders should be sent to the market. This component is often referred to as the "Trading Strategy".

- Code to determine when to cover a position and take a profit or loss. This component is often referred to as "Position Management".

- Code to configure and manage the Black Box Trading System.

- Code to determine which stocks are eligible for short sale.

- Code to maintain order state, save order state in a recovery file, and rebuild order state from the recovery file when the Black Box Trading System is restarted.

- Code to interface to a console. A console is used to configure, control and monitor the Black Box Trading System.

The following figure illustrates the components of a typical Black Box Trading System.
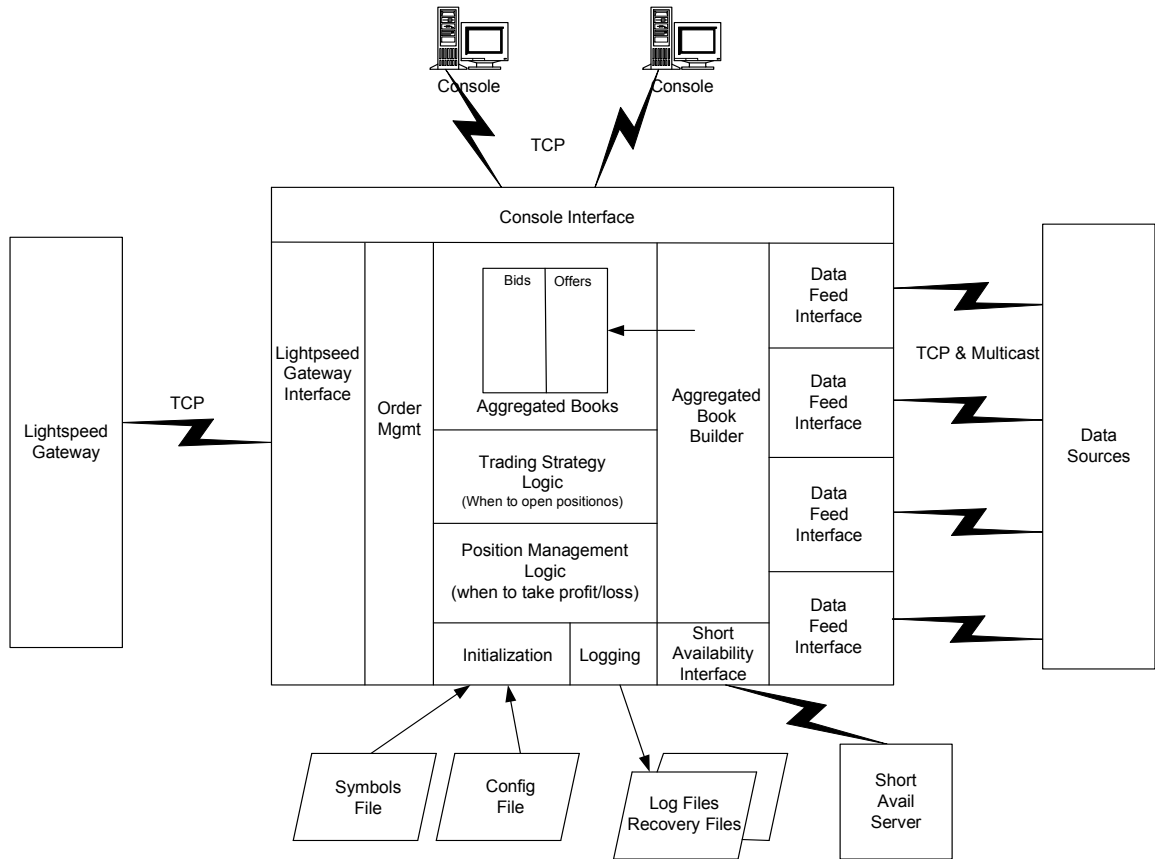
Figure 2: Black Box Trading System Components

The components of a typical Black Box Trading System are shown in the box in figure 2. Note, a Black Box Trading System can be designed many ways, but the components described here are typical of most Black Box Trading Systems.

Data Interface. The Data Interface components shown on the right side of figure 2 are responsible for obtaining the market data from the data source. For example, the Black Box Trading System may want market data from INET, ARCA, BATS, EDGX, and EDGA. The market data from these sources may be in a different formats, and the Data Interface will implement the data exchange protocol specific to a data source. In figure 2 there are four data sources, thus four Data Interface components are required.

Aggregated Book Builder: A typical Black Box Trading System will want to combine the data from multiple sources to create an aggregated view of the market. The Aggregated Book Builder will take the data from all data sources and combine the data into an aggregated book.

Trading Strategy Logic: The Trading Strategy Logic will analysis the market data and when conditions are met, it will send orders to buy and sell stock. An example of a very simple Trading

Strategy would be to compute a 20 minute moving average, and when the price moves above the 20 minute moving average send a buy order to the market.

Position Management Logic: The Position Management Logic will determine when positions will be covered. An example of a very simple position management strategy is to cover the position when a predetermined profit has been achieved, or when the market have moved against the position and should be covered at a loss.

Order Management: The Order Management component will manage the state for all order. An order can be in a number of possible states. The following is a list of the possible states:

- The order has been set the to Lightspeed Gateway and waiting for acknowledgement

- The order has been acknowledged and has not been filled (live order).

- The order has been partially filled and the remainder is live (live order with partial fill).

- The order has been partially filled and the remainder has been canceled (canceled order with partial fill).

- The order has been canceled and was not filled (canceled).

- A cancel request has been sent to the Lightspeed Gateway and waiting for a response (cancel pending).

Another important role of Order Management component is to rebuild order state when the Black Box Trading System is shutdown and restarted. The Order Management component must save information in a log file (often referred as a recovery file), and rebuild order state from the information in the recovery file upon restart.

Lightspeed Gateway Interface: The Lightspeed Gateway interface component implements the protocol specified by the Lightspeed Gateway.

Short Availability Interface: The Short Availability Interface component provides access to information describing which securities are eligible for short sale.

Initialization: The Initialization module is responsible for performing all initialization function required by the Black Box Trading System. This may include reading configuration from a configuration file.

Logging: The Logging component is responsible for logging all information required by the Black Box Trading System. As mentioned above, order information should be logged so that order state can be rebuilt upon restart.

Console Interface: The Console Interface component provides the interface for interacting with a remote console. A remote console is a separate software application. The console communicates with the Black Box Trading System via a TCP connection and will exchange messages with the Black Box Trading System. These messages may contain commands to instruct the Black Box

Trading System to perform a task, or they may contain status information to be displayed on the console.

3.1. Connectivity Model

There are many ways to configure a Black Box Trading System with regard to where the computers are located. However, a typical set up is to have one computer running the Black Box Trading System application located in Lightspeed's data center, and a second computer running a remote console located at the customer's office.   This model has the advantage that the Black Box Trading System located in Lightspeed's data center will have access to the Data servers, Order server, Short Availability Server over the local area network. A remote console can be located anywhere, at the customer's office or the customer's home, and can communicate with the Black Box Trading System over the Internet.

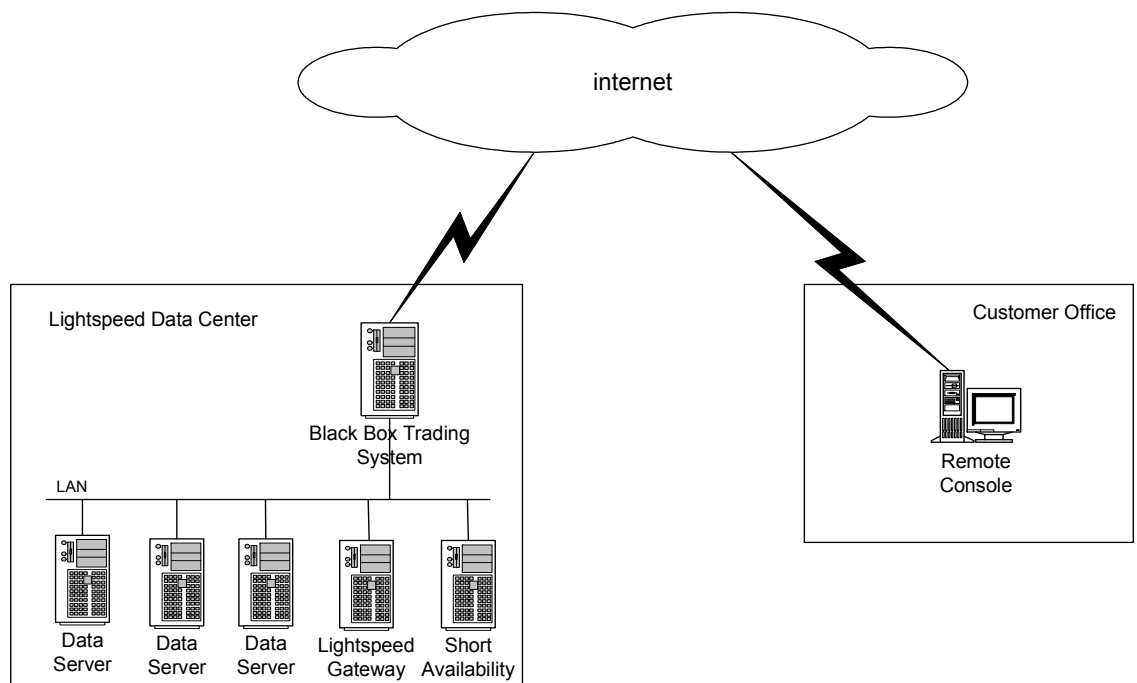The following figure illustrates this connectivity model.

Figure 3: Connectivity Model – Remote Console

4. Black Box SDK

4.1. Purpose of the SDK

Many of the functions that a Black Box Trading System must perform are common to all Black Box Trading Systems. In fact, all the components shown in figure 2 are required in every Black Box Trading System to some degree. Many of the components are identical in every Black Box Trading System. For example, every Black Box Trading System requires market data, the ability to enter order, the ability to maintain order state, the ability to initialize, the ability to log information, and a way to be controlled and managed.

The two components that differ in function are the Trading Strategy Logic and the Position Management logic. The SDK provides the common components, thus allowing the customer to focus on the Trading Strategy and Position Management. This has the following advantages for the customer:

- Allows the customer to focus on the trading strategy and position management.

- Reduces the time required to develop a Black Box Trading System because many of the components are provided by the SDK.

- Reduce development cost.

4.2. SDK Components

Lightseed's SDK consists of the following components:

- Overview Document (this document)

- SDK Library

- Customer Source Code Template (Sample Trading Application)

- SDK Library API Document

- Console source code

- How-To documents and Frequently Asked Questions

Overview Document: The Overview Document provides an introduction to Lightspeed's SDK. This should be the first document read and will give the reader basic information about Black Box Trading Systems, and an introduction to Lightspeed's SDK. The SDK is intended to help the customer develop their Black Box Trading System.

SDK Library: The library contains the executable code that provides the functions that are common to all Black Box Trading Systems. Figure 4 shows the Black Box Trading System components again, but in this figure the functions provided by the SDK library are shown with a gray background.
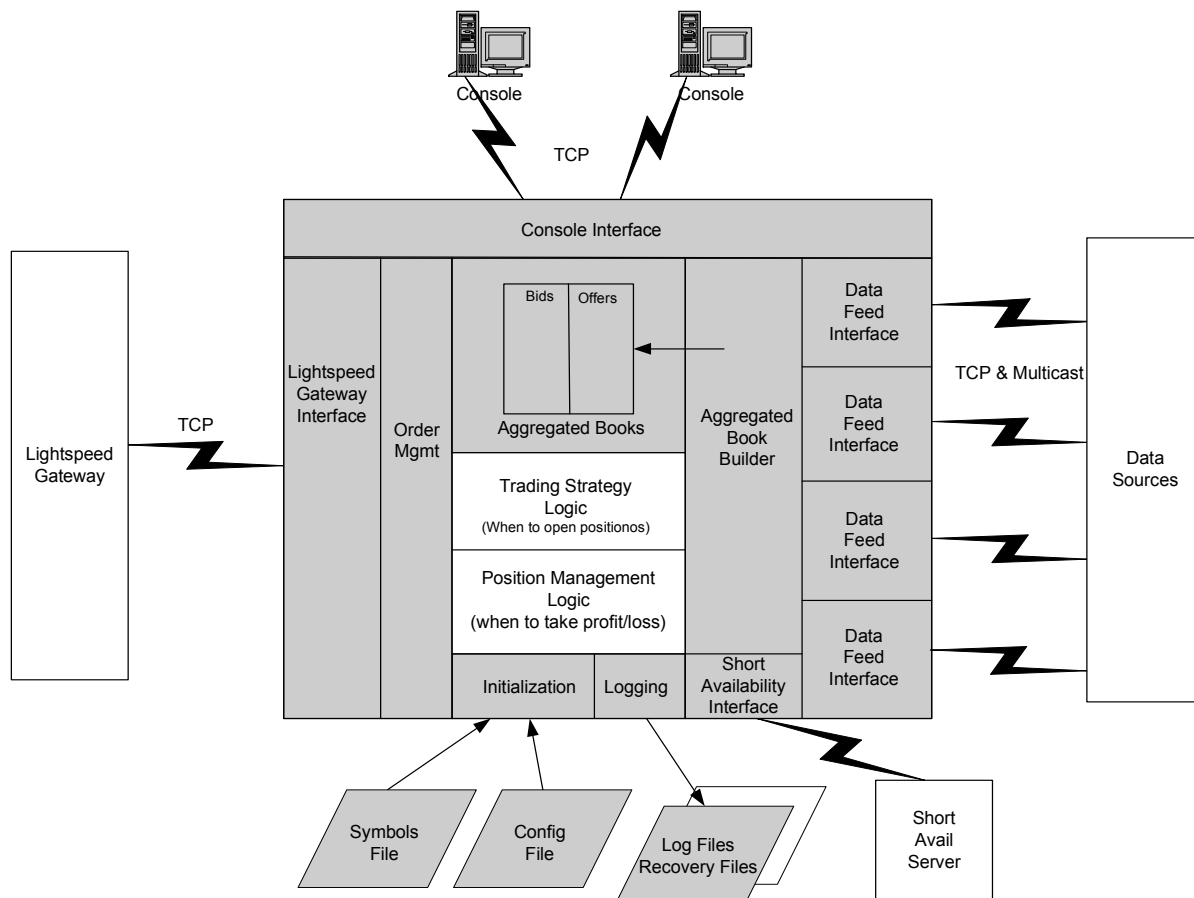

Figure 4: SDK Library components

The SDK provides template symbol files and Configuration files. The SDK Library also contains code to create the Recovery file, and to rebuild order state from the Recovery file when the Black Box Trading System is restarted. These files are shown in gray, but they are not part of the Library executable. They are shown in gray because the Library either uses the information in the file (eg, Config file) or creates the file (eg, Recovery File).

All the customer needs to do is write the Trading Strategy code and the Position Management code, shown with a white background in figure 4.

Customer Source Code Template: The Customer Source Code Template is a source code file that is used by the customer to develop their Trading Strategy code and the Position Management code.

The Customer Source Code Template provide with the SDK contains a sample trading application. The sample trading application is intended to help the customer quickly learn how to use the SDK Library. When the sample trading application is compiled and linked with the SDK Library a fully functional Black Box Trading System is produced. NOTE: The sample trading application is very simple and would lose money most days if traded. The purpose of the sample trading application is to demonstrate how to build a Black Box Trading System using the SDK, and to aid the customer in learning how to use the SDK Library.

SDK Library API Document: The SDK Library API Document describes the SDK Library interface. The interface consists of function calls and call-back functions. The Customer Source Code Template uses the SDK Library API and contains examples of many of the function calls and call-back functions. The intended audience for this document is the software developer who will be writing the Trading Strategy code and the Position Management code.

Console Source Code: The source code for a console is provided as part of the SDK. The console is a simple Windows application with very limited Windows functionality. The idea is to provide a basic console that can be used to control and manage the Black Box Trading System, and can be modified to add customer specific features by a non-Windows programmer. Experienced Windows programmers may want to write their own console application if they desire a more sophisticated Windows application.

How-To Documents: Deploying a functioning Black Box Trading System requires more than writing the Trading Strategy and the Position Management code. Many things must be considered. For example, what computers to purchase, how to configure the computers, what operating system to use, creating secure tunnels between the remote console and the Black Box Trading System, setting up accounts, etc… How-To Documents are provided to help the customer deal with these items.

5. Required Hardware and Software

The recommended configuration is to run the Black Box Trading System application on a computer located in Lightspeed's data center, and to monitor and control trading using a remote console located at the customers site.

The Black Box Trading System (based on the SDK) can be run on a Linux system or a Windows system located at Lightspeed's data center.

The minimum recommended hardware configuration for the Black Box Trading System is shown below:

- Rack mountable 1U PC

- 3 to 5 100Mpbs Ethernet NICs

- Minimum of 2 Giga Bytes of memory

- Minimum of 50 Giga Bytes Hard drive

- Minimum of 2.5 GHz processor

The console is a windows application and will run on most Windows operating systems (eg, XP, Vista, etc..).

To allow the console to connect to a Linux Black Box Trading System over the Internet, a secure tunnel must be created between the console system and the Linux system running the Black Box Trading System application. SSH is used to create the tunnel. Refer to the SSH How-To document for instructions for installing the SSH software and creating the tunnel.

Remote Desktop Connection is used to access a Windows system located in Lightspeed's data center. Refer to Microsoft's Remote Desktop Connection documentation.