

IB Gateway Software

For CTCL FIX over the Internet

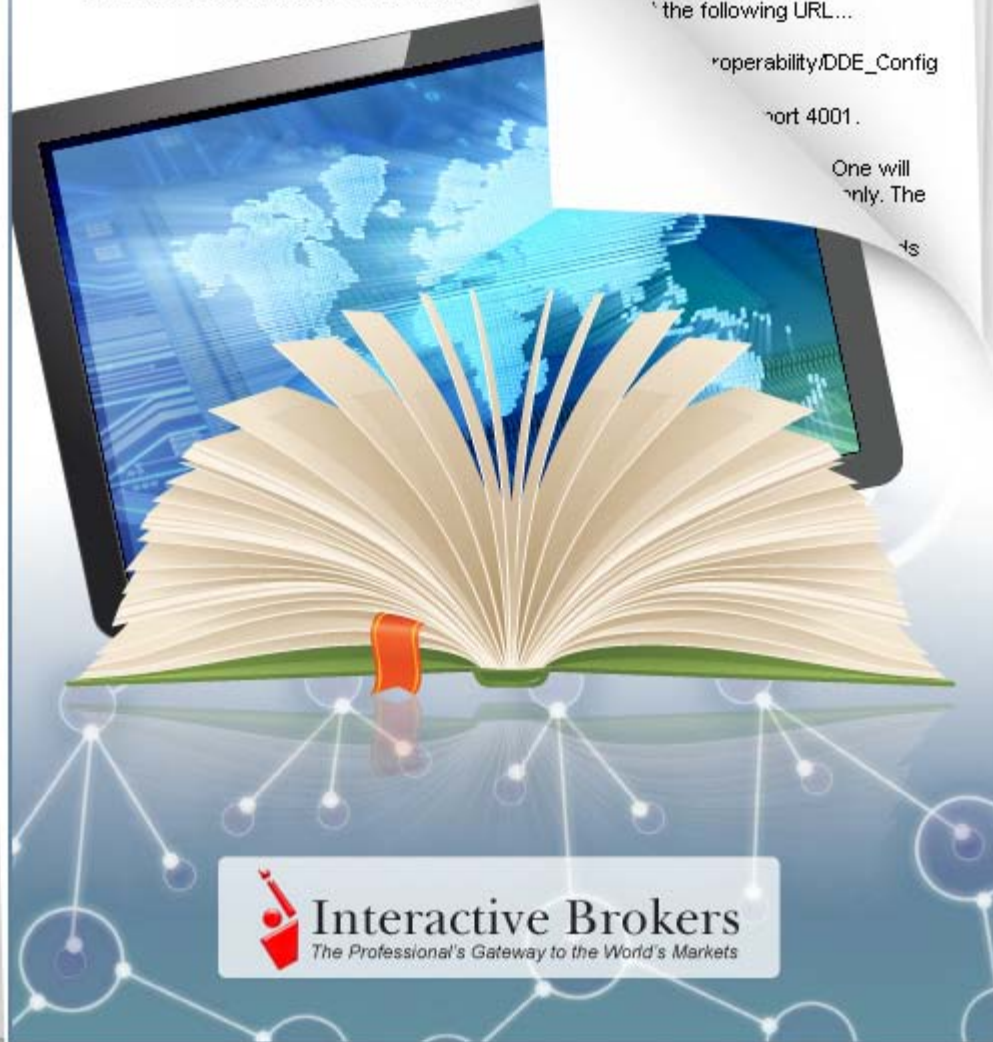
ions to IB over the Internet. It is a piece
engine and IB. The software also can
er Workstation API does. Coding
n this manual. For details on
the following URL...

operability/DDE_Config

port 4001.

One will
only. The

ts



Revision 3.1 – October 9, 2013

CONFIDENTIAL

Interactive Brokers Group LLC and its affiliates ("IB") have intellectual property right protections regarding various IB systems, methods, processes, functionalities and documentation. The information contained in this document is proprietary and confidential, and may not be disclosed to any person or entity except authorized employees and agents of the recipient organization who have a need to know such information. The information contained in this document is current as of the publication date, and is subject to change without notice. IB will not accept responsibility for damages, direct or indirect, caused by any error or omission in this document.

Contents

About the IB Gateway 3

 Using the Gateway Client..... 3

Logging In to the IB Gateway 4

 API Messages Supported by the IB Gateway Client 8

About the IB Gateway

The IB Gateway acts as a bridge between your FIX engine and the IB trading system, and is required for FIX CTCL connections over the Internet. The software can also be used to deliver market data, using the same protocol as the IB API. A list of the API market data functions supported through the IB Gateway is included on the last page of this document.

NOTE: When you use the API functions, the IB Gateway takes the place of TWS, consequently you do not need to have a TWS session running.

The gateway client communicates over the internet using **SSL on TCP port 4001**. Firewall permissions may need to be made for that port.

It is important to note that separate accounts are required for **FIX order routing** and **API market data requests**, and the IB Gateway Login box provides two separate login areas.

There are “throughput limits” at the gateway, and the client will be warned if the predefined limits are exceeded. The limits and warnings are as follows:

- **Limit:** If more than **50 messages per second** or 33000 bytes per second are sent...
- **Warning:** Initially, user defined warning messages will be sent via FIX (35=U, 58=Text String describing the warning, and 6040=34 for the first two warnings). If the “throughput limits” are **exceeded a third time, the socket will be broken.**

Using the Gateway Client

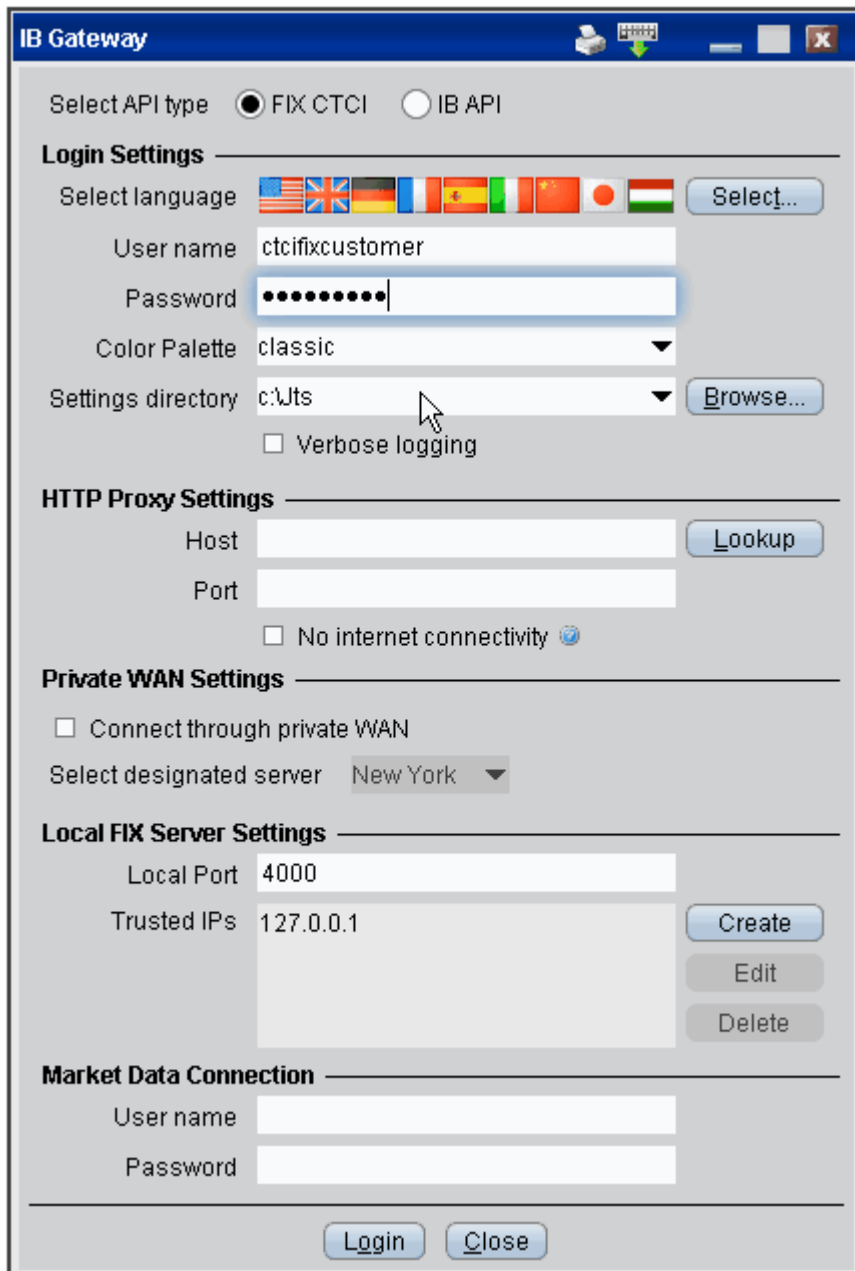
The gateway client is a java program, allowing it to run on any machine that has the Java Runtime Environment installed on it. IB recommends using JRE 6 or higher, which can be downloaded directly from Sun Microsystems at www.java.com.

You can log into the IB Gateway from the Login menu of the IB web site, or download the standalone version from the [FIX CTCL Connectivity](#) page.

Logging In to the IB Gateway

The login box provides radio button selections for the FIX CTCL or IB API connection type. Ensure that the FIX CTCL radio button is selected. Note that the FIX CTCL login box can accept two sets of login credentials: **Login Settings** is used to connect to a FIX session (and requires a FIX username and password issued by the Client Integration group) and the **Market Data Connection** fields are used to get market data via the API. You can log into just the FIX, just the Market Data, or both simultaneously, depending on your requirements. See the last section for a list of API messages supported by the IB Gateway.

NOTE: If you only want market data, you can also select the IB API radio button and log into the IB Gateway for API.




The screenshot shows the 'IB Gateway' application window. At the top, there's a title bar with standard window controls. Below it, the 'Select API type' section has two radio buttons: 'FIX CTCL' (selected) and 'IB API'. The 'Login Settings' section includes a 'Select language' dropdown with flags, a 'User name' field with 'ctclfixcustomer', a 'Password' field with masked characters, a 'Color Palette' dropdown set to 'classic', a 'Settings directory' field with 'c:\Ibs' and a 'Browse...' button, and a 'Verbose logging' checkbox. The 'HTTP Proxy Settings' section has 'Host' and 'Port' fields with a 'Lookup' button, and a 'No internet connectivity' checkbox. The 'Private WAN Settings' section has a 'Connect through private WAN' checkbox and a 'Select designated server' dropdown set to 'New York'. The 'Local FIX Server Settings' section has a 'Local Port' field set to '4000', a 'Trusted IPs' field with '127.0.0.1', and 'Create', 'Edit', and 'Delete' buttons. The 'Market Data Connection' section has 'User name' and 'Password' fields. At the bottom, there are 'Login' and 'Close' buttons.

IB Gateway

Select API type ☒ FIX CTCL ☐ IB API

Login Settings

Select language  Select...

User name ctcifixcustomer

Password

Color Palette classic

Settings directory c:\Ibs Browse...

☐ Verbose logging

HTTP Proxy Settings

Host Lookup

Port

☐ No internet connectivity

Private WAN Settings

☐ Connect through private WAN

Select designated server New York

Local FIX Server Settings

Local Port 4000

Trusted IPs 127.0.0.1 Create Edit Delete

Market Data Connection

User name

Password

Login Close

After you log into the FIX session, you can connect your FIX engine to the IB Gateway client. You will see a screen that looks like the one below. Status indicators show the status of the connections to and from the IB Gateway client.

Note that for FIX CTCL order routing, the client will connect and then disconnect until a FIX engine is connected to it on the user’s side.

Interactive Brokers Gateway. Order Routing Account: icdemo20. API Account: pspiro12.

Connection Status

Purpose	Status
Interactive Brokers Order Routing Server	inactive
Interactive Brokers API Server	connected
Order Routing Client	disconnected
API Client	disconnected
Market Data Farm	ON: useoption
Historical Data Farm	disconnected

Gateway Activity

16:42:35 JTS-Listener-10: received data and reset ping with: 5000
16:42:35 156 JTS-Listener-10: start ping with 5000 for: MD-Ping-BA-15
16:42:35 156 JTS-Listener-10: received data and reset ping with: 5000
16:42:35 156 JTS-Listener-10: start ping with 5000 for: MD-Ping-BA-16
16:42:35 156 JTS-Listener-10: received data and reset ping with: 5000
16:42:35 156 JTS-Listener-10: start ping with 5000 for: MD-Ping-BA-17
16:42:35 171 MD-Ping-BA-13: Terminating
16:42:35 171 JTS-MDDispatcher-11: setReady for itrmis100:4020:usoption:null true
16:42:35 171 JTS-MDDispatcher-11: authen ready and reset ping to 60000
16:42:35 171 JTS-MDDispatcher-11: start ping with 30000 for: MD-Ping-AA-18
16:42:35 171 JTS-MDDispatcher-11: Connection is ready for ibrmis100:4020:usoption:null
16:42:35 187 MD-Ping-BA-14: Starting
16:42:35 203 MD-Ping-BA-14: Terminating
16:42:35 218 MD-Ping-BA-15: Starting
16:42:35 218 MD-Ping-BA-15: Terminating
16:42:35 218 MD-Ping-BA-16: Starting
16:42:35 218 MD-Ping-BA-16: Terminating
16:42:35 218 MD-Ping-BA-17: Starting
16:42:35 218 MD-Ping-BA-17: Terminating
16:42:35 218 MD-Ping-AA-18: Starting
16:42:35 343 JTS-MDDispatcher-11: Received MD route table

If you log into both, once the FIX engine and/or API client software have successfully connected to the client software, your screen will look like this:

Interactive Brokers Gateway. Order Routing Account: icdemo20. API Account: pspiro12.

Connection Status

Purpose	Status
Interactive Brokers Order Routing Server	connected
Interactive Brokers API Server	connected
Order Routing Client	connected
API Client	1 connected
Market Data Farm	ON: usoption
Historical Data Farm	Inactive: ushmds2

Gateway Activity

```

16:57:25:625 JTS-Listener-31: Error reading from socket - java.net.SocketException: socket closed
16:57:25:625 MD-Ping-AA-37: onDisconnected ibctsrv1
16:57:25:625 JTS-MDDispatcher-32: Terminating
16:57:25:625 MD-Ping-AA-37: setReady for ibctsrv1:4021:ushmds2:null false
16:57:25:625 JTS-Listener-31: range = 10000 random: 2098 m_nextConnectTime: 1129755443723
16:57:25:625 JTS-Listener-31: Terminating
16:57:25:625 MD-Ping-AA-37: Terminating
16:57:35:421 MD-Ping-AA-18: sendPingMessage for ibmis100:4020:usoption:null
16:58:13:375 JTS-SocketListener: eServersChanged: 1
16:58:13:375 JTS-EServerSocket2: [0:0:23:1:0:0:0:SYS] Starting new conversation client(0) on 127.0.0.1/127.0.0.1
16:58:13:390 JTS-EServerSocket2: [0:22:23:1:0:0:0:SYS] TWS socket server version is 23
16:58:13:390 JTS-EServerSocket2: [0:22:23:1:0:0:0:SYS] Client version is 22
16:58:13:421 JTS-EServerSocket2: [0:22:23:1:0:0:0:SYS] Client ID is 0
16:58:35:453 MD-Ping-AA-18: sendPingMessage for ibmis100:4020:usoption:null
16:58:49:375 AWT-EventQueue-0: Received connect request: Socket[addr=/127.0.0.1,port=1246,localport=4000]
16:58:49:375 AWT-EventQueue-0: Contacting 192.9.217.41:34837 as user icdemo20
16:58:49:375 AWT-EventQueue-0: Contacting 192.9.217.41:34837 as user icdemo20
16:58:49:375 AWT-EventQueue-0: Local mkt svr already started at 4001
16:58:49:578 AWT-EventQueue-0: Connected to remote: Socket[addr=ibsrv22/192.9.217.41,port=34837,localport=1247]
16:58:49:578 AWT-EventQueue-0: Circuit established: Socket[addr=/127.0.0.1,port=1246,localport=4000] <=> Socket[addr=ibsrv22/192.9.217.41,port=34837,localport=1247]
  
```

The Historical Data Farm indicator will only show green when a historical market data request or scanner subscription has been made, as is shown in the following screenshot:

Interactive Brokers Gateway. Order Routing Account: icdemo20. API Account: pspiro12.

Connection Status

Purpose	Status
Interactive Brokers Order Routing Server	connected
Interactive Brokers API Server	connected
Order Routing Client	connected
API Client	1 connected
Market Data Farm	ON: usoption
Historical Data Farm	ON: ushmds2

Gateway Activity

```

16:54:26:750 AWT-EventQueue-0: The EBuffer buffer has grown to 17094 bytes
16:54:26:750 AWT-EventQueue-0: The EBuffer buffer has grown to 21232 bytes
16:54:26:875 AWT-EventQueue-0: Generating request for: QQQQ@SMART
16:54:26:875 AWT-EventQueue-0: HMDS Query attempt occurring
16:54:35:375 MD-Ping-AA-18: sendPingMessage for ibmis100:4020:usoption:null
16:54:38:171 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:54:46:856 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:54:57:218 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:55:05:765 AWT-EventQueue-0: Received connect request: Socket[addr=/127.0.0.1,port=1238,localport=4000]
16:55:05:765 AWT-EventQueue-0: Contacting 192.9.217.41:34837 as user icdemo20
16:55:05:765 AWT-EventQueue-0: Contacting 192.9.217.41:34837 as user icdemo20
16:55:05:765 AWT-EventQueue-0: Local mkt svr already started at 4001
16:55:06:000 AWT-EventQueue-0: Connected to remote: Socket[addr=ibsrv22/192.9.217.41,port=34837,localport=1239]
16:55:06:000 AWT-EventQueue-0: Circuit established: Socket[addr=/127.0.0.1,port=1238,localport=4000] <=> Socket[addr=ibsrv22/192.9.217.41,port=34837,localport=1239]
16:55:08:015 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:55:19:625 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:55:25:578 MD-Ping-AA-37: sendPingMessage for ibctsrv1:4021:ushmds2:null
16:55:30:937 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:55:35:390 MD-Ping-AA-18: sendPingMessage for ibmis100:4020:usoption:null
16:55:42:156 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
  
```

The screenshot below shows that the local FIX engine and API client have disconnected:

Interactive Brokers Gateway. Order Routing Account: icdemo20. API Account: pspiro12.

Connection Status

Purpose	Status
Interactive Brokers Order Routing Server	inactive
Interactive Brokers API Server	connected
Order Routing Client	disconnected
API Client	disconnected
Market Data Farm	ON: usoption
Historical Data Farm	Inactive: ushmds2

Gateway Activity

```

16:56:48:078 JTS-EServerSocket1: [0:22:23:1:0:0:0:SYS] Ending conversation with client{0} at 127.0.0.1/127.0.0.1
16:56:48:078 JTS-EServerSocket1: eServersChanged: 0
16:56:48:078 JTS-EServerSocket1: checking market data and cleaning maps
16:56:48:078 JTS-EServerSocket1: eServersChanged: 0
16:56:49:500 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:57:00:703 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:57:12:000 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:57:23:343 JTS-MDDispatcher-32: sendMessage for ibctsrv1:4021:ushmds2:null
16:57:25:609 MD-Ping-AA-37: Closing hmds dormant connection ushmds2
16:57:25:625 MD-Ping-AA-37: m_disconnectTime 1129755445625
16:57:25:625 MD-Ping-AA-37: Disconnecting
16:57:25:625 JTS-Listener-31: Error reading from socket - java.net.SocketException: socket closed
16:57:25:625 MD-Ping-AA-37: onDisconnected ibctsrv1
16:57:25:625 JTS-MDDispatcher-32: Terminating
16:57:25:625 MD-Ping-AA-37: setReady for ibctsrv1:4021:ushmds2:null false
16:57:25:625 JTS-Listener-31: range = 10000 random: 2098 m_nextConnectTime: 1129755443723
16:57:25:625 JTS-Listener-31: Terminating
16:57:25:625 MD-Ping-AA-37: Terminating
16:57:35:421 MD-Ping-AA-18: sendPingMessage for ibmis100:4020:usoption:null

```

API Messages Supported by the IB Gateway Client

At this time, the following API market data functions are supported by the IB Gateway client:

- Request Contract Details
- Request Market Data
- Cancel Market Data
- Request Market Depth
- Cancel Market Depth
- Request Historical Data
- Cancel Historical Data
- Request Scanner Subscription
- Cancel Scanner Subscription
- Request Scanner Parameters
- Request Account Updates

To find details on using the above methods, please refer to the [API Reference Guide](#).