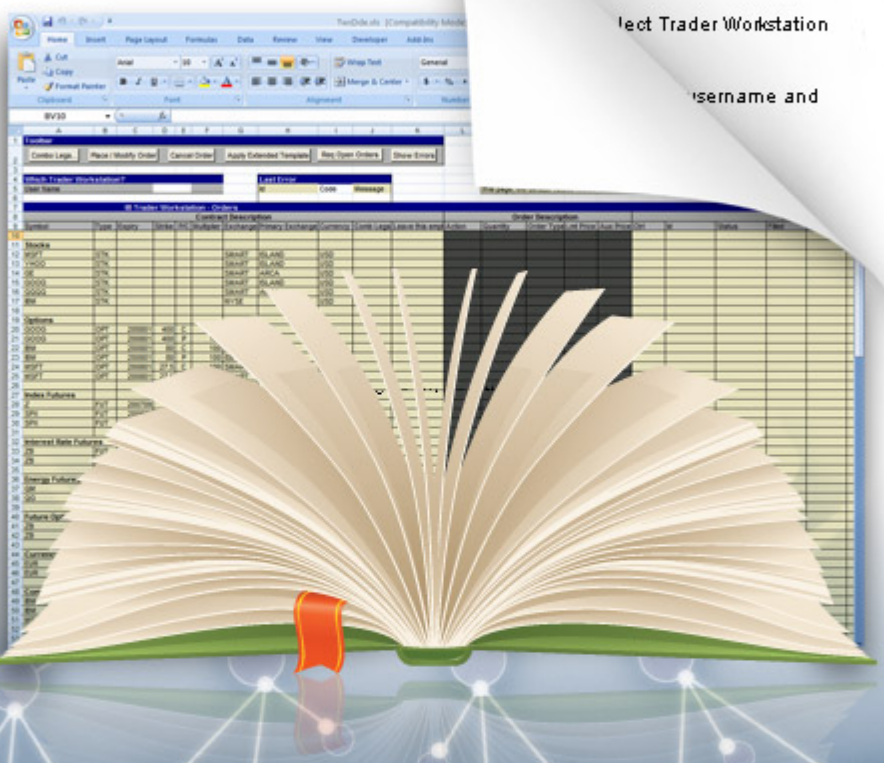
Getting Started with the TWS DDE for Excel API



Interactive Brokers

The Professional's Gateway to the World's Markets

Getting Started with the TWS DDE for Excel API

January 2014
Supports TWS API Release 9.69

Contents

1

Introduction

You might be looking at this book for any number of reasons, including:

- You love IB's TWS, and are interested in seeing how using its API can enhance your trading.
- You use another online trading application that doesn't provide the functionality of TWS, and you want to find out more about TWS and its API capabilities.

Or maybe you have a reason of your own. However you traveled this happy path, you now hold in your hands a unique and potentially priceless tome of information. Well, maybe that's a tiny bit of an exaggeration. The information in this book will, however, show you how to use the trading functionality of IB's Trader Workstation and its API. Keep reading to find out how easy it can be to augment and enhance your trading via the TWS API.

How to Use this Book

Before you get started, you should read this section to learn how this book is organized, and see which graphical conventions are used throughout.

This book is designed as a reference book, not a lesson-by-lesson textbook, so feel free to skip around and read what interests you at the time. That being said, you'll see that the *Pages* chapters in section III are ordered as the pages appear on the Excel spreadsheet, but that's just to make it easier for you (and us!) to follow.

To get the most of out of this guide, you should have at least basic PC skills as well as some basic familiarity with Microsoft Excel.

Throughout this book, we use the acronym "TWS" in place of "Trader Workstation." So when you see "TWS" anywhere, you'll know we're talking about Trader Workstation.

Organization

We've divided this book into five major sections, each of which comprises a number of smaller subsections, and each of **those** have even smaller groupings of paragraphs and figures…well, you get the picture. Here's how we've broken things down:

Part 1: Introduction to this Guide

This section serves as a brief introduction to the rest of the guide.

Part 2: Introducing the DDE for Excel API

The chapters in this section help you answer those important questions you need to ask before you can proceed - questions like: "What can TWS do for me?" and "Why would I use an API?" and "If I WERE to use an API, what does the Excel platform have to offer me?" and even "What other API choices do I have?"

If you already know you want to learn about the TWS API, just skip on ahead.

Part 3: Preparing to Use the DDE for Excel API

Section II walks you through the different things you'll need to do before your API application can effectively communicate with TWS. We'll help you configure TWS, download the API software, get the API sample spreadsheet up and running, and hopefully become more comfortable saving and opening files.

Part 4: Getting to Know the DDE for Excel API Spreadsheet

If trading really is your cup of tea, you'll want to know all about this section. We'll examine each page of your new API spreadsheet, and discuss the cool things you can do with all of the "action buttons" that we have thoughtfully included to help improve your efficiency.

Part 5: Where to Go from Here

After filling your head with boatfuls of API knowledge, we wouldn't dream of sending you off empty-handed! Part V tells you where to look to keep abreast of new API releases (which of course means new features you can incorporate into your trading), how to navigate the Interactive Brokers website to find support and information, where to find some helpful resources outside the realm of IB support, (such as Visual Basic and other programming help), and how to find answers to Excel-specific questions.

Footnotes and References

[1]Any symbols displayed are for illustrative purposes only and are not intended to portray a recommendation.

Icons

TWS-Related

When you see this guy, you know that there is something that relates specifically to TWS: a new feature to watch for, or maybe something you're familiar with in TWS and are looking for in the API.

Excel Tip

The Excel tips are specific to Microsoft Excel, and we don't include too many of those, but when you see it you should check it out - it may save you some time.

Important!

This shows you where there is a particularly useful or important point being made.

Take a Peek!

You may want to take a peek, but it isn't the end of the world if you don't.

Go Outside!

This icon denotes references outside of this book that we think may help you with the current topic, including links to the internet or IB site, or a book title.

Document Conventions

Here's a list of document conventions used in the text throughout this book.

Convention	Description	Examples
Bold	Indicates: • menus • screens • windows • dialogs • buttons • tabs • keys you press • names of classes and methods	On the **Tickers** page, select a row by clicking the row number in the far left column… Press **Ctrl+C** to copy…
Italics	Indicates: • commands in a menu • objects on the screen, such as text fields, check boxes, and drop-down lists	To access the users' guide, under the **Software** menu, select *Trader Workstation*, then click *Users' Guide*.

2

TWS and the DDE for Excel API

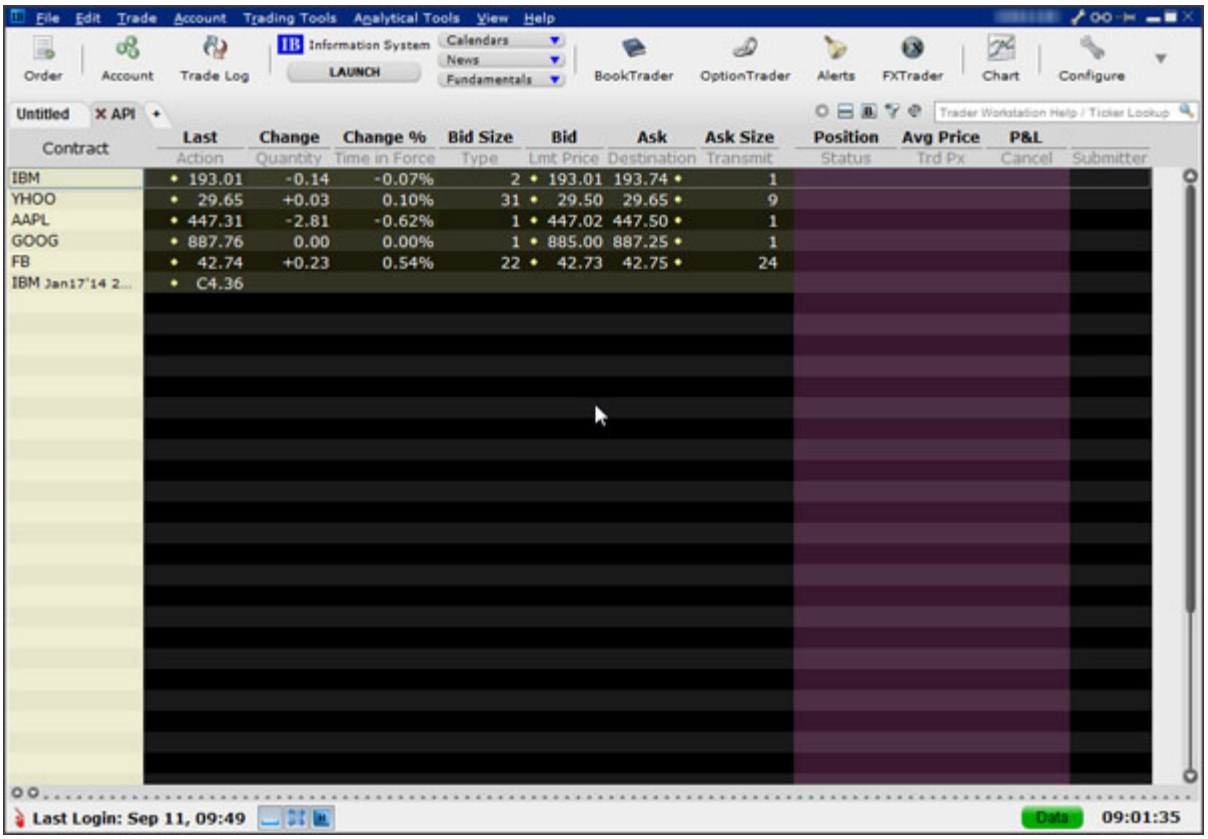
The best place to start is by getting an idea of what Trader Workstation (TWS), is all about. In this section, first we'll describe TWS and some of its major features. Then we'll explain how the API can be used to enhance and customize your trading environment. Finally, we'll give you a summary of some of the things the DDE for Excel API, (versus our other API platforms), can do for you!

Here's what you'll find in this section:

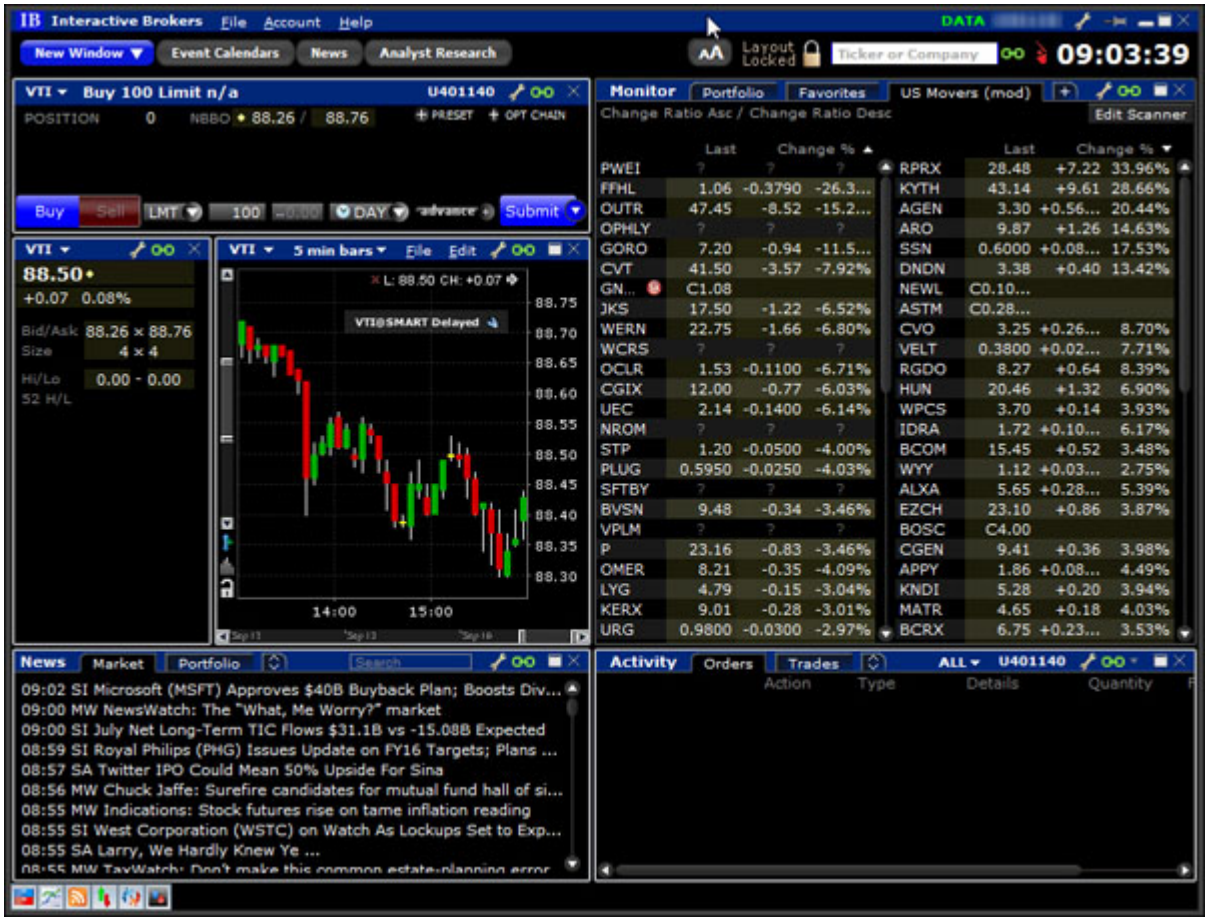
Chapter 1 - What is Trader Workstation?

Interactive Brokers' Trader Workstation, or TWS, is an online trading platform that lets you trade and manage orders for all types of financial products (including stocks, bonds, options, futures and Forex) on markets all over the world - all from your choice of two workspaces:

- The Advanced Order Management workspace, which is a single spreadsheet-like screen.



- Mosaic, a single, comprehensive and intuitive workspace which provides easy access to Trader Workstation's trading, order management and portfolio functionality.



To get a little bit of a feel for TWS, go to the IB website and try TWS demo application. Its functionality is slightly limited and it only supports a small number of symbols, but you'll definitely get the idea. Once you have an approved, funded account you'll also be able to use PaperTrader, our simulated trading tool, with paper-money funding in the amount of $1,000,000, which you can replenish at any time through TWS Account Management.

What Can You Do with TWS?

So, what can you do with TWS? For starters, you can:

- Send and manage orders for all sorts of products (all from the same screen!);
- Monitor the market through Level II, NYSE Deep Book and IB's Market Depth;
- Keep a close eye on all aspects of your account and executions;
- Use Technical, Fundamental and Price/Risk analytics tools to spot trends and analyze market movement;
- Completely customize your trading environment through your choice of modules, features, tools, fonts and colors, and user-designed workspaces.

Basically, almost anything you can think of TWS can do - or will be able to do soon. We are continually adding new features, and use the latest technology to make things faster, easier and more efficient. As a matter of fact, it was this faith in technology's ability to improve a trader's success in the markets (held by IB's founder and CEO Thomas Peterffy) that launched this successful endeavor in the first place. Since the introduction of TWS in 1995, IB has nurtured this relationship between technology and trading almost to the point of obsession!

A Quick Look at TWS

This section gives you a brief overview of the most important parts of TWS.

The TWS Quote Monitor

First is the basic TWS Quote Monitor. It's laid out like a spreadsheet with rows and columns. To add tickers to a page, you just click in the Underlying column, type in an underlying symbol and press Enter, and walk through the steps to select a product type and define the contract. Voila! You now have a live market data line on your trading window. It might be for a stock, option, futures or bond contract. You can add as many of these as you want, and you can create another window, or trading page, and put some more on that page. You can have any and all product types on a single page, maybe sorted by exchange, or you can have a page for stocks, a page for options, etc. Once you get some market data lines on a trading page, you're ready to send an order.

The Order Ticket

What? An order ticket? Sure, we have an order ticket if that's what you really want. But we thought you might find it easier to simply click on the bid or ask price and have us create a complete order line instantly, right in front of your eyes! Look it over, and if it's what you want click a button to transmit the order. You can easily change any of the order parameters right on the order line. Then just click the green Transmit guy to transmit your order! It's fast and it's easy, and you can even customize this minimal two-click procedure (by creating hotkeys and setting order defaults for example) so that you're creating and transmitting orders with just ONE click of the mouse.

Real-Time Account Monitoring

TWS also provides a host of real-time account and execution reporting tools. You can go to the Account Window at any time to see your account balance, total available funds, net liquidation and equity with loan value and more. You can also monitor this data directly from your trading window using the Trader Dashboard, a monitoring tool you can configure to display the last price for any contracts and account-related information directly on your trading window.

So - TWS is an all-inclusive, awesome powerful trading tool. You may be wondering, "Where does an API fit in with this?" Read on to discover the answer to that question.

For more information on TWS, see the TWS Users' Guide on our web site.

Chapter 2 - Why Would I Use an API?

OK! Now that you have a little bit more of a feel for TWS and what it can do, you can move on to the amazing API. If you actually read that last chapter, you might be thinking to yourself "Why would I want to use an API when TWS seems to do everything." Or you might be thinking "Hm, I wonder if TWS can… fill in the blank?" OK, if you're asking the first question, we'll explain why you might find the API useful, and if you're asking the second, it's actually the API that can fill in that blank.

TWS has the capability to do tons of different things, but it does them in a certain way, and displays results in a certain way, and it's pretty likely that our development team, as fantastic as they are, hasn't yet exhausted the number of features and way of implementing these features that all of you collectively can come up with. So it's very likely that you, with your unique way of thinking, will be or have been inspired by the power of TWS to say something like "Holy moly, I can't believe I can really do all of this with TWS! Now if I could only just …fill in the blank…, my life would be complete!"

That's where the API comes in. Now, you can …fill in the blank…! It's going to take a little work to get there, but once you see how cool it is to be able to access functionality from one application from another, completely unrelated one, you'll be hooked.

TWS and the API

In addition to allowing you pretty much free reign to create new things and piece together existing things in new ways, the API is also a great way to automate your tasks. You use the API to harness the power behind TWS - in different ways.

Here's an analogy that might help you understand the relationship between TWS and the API. Start by imagining TWS as a book (since TWS is constantly being enhanced, our analogy imagines a static snapshot of TWS at a specific point in time). It's the reference book you were looking for, filled with interesting and useful information, a book with a beginning, middle and end, which follows a certain train of logic. You could skip certain chapters, read Chapter 10 first and Chapter 2 last, but it's still a book. Now imagine, in comparison, that the API is the word processing program in which the book was created with the text of the book right there. This allows you access to everything in the book, and most importantly, it lets you continually change and update material, and automate any tasks that you'd have to perform manually using just a book, like finding an index reference or going to a specific page from the table of contents.

The API works in conjunction with TWS and with the processing functions that run behind TWS, including IB's SmartRouting, high-speed order transmission and execution, support for over 40 orders types, etc. TWS accesses this functionality in a certain way, and you can design your API to take advantage of it in other ways.

Available API Technologies

IB provides a suite of custom APIs in multiple programming languages, all to the same end. These include Java, C++, Active X for Visual Basic and .NET, ActiveX for Excel, DDE for Excel (Visual Basic for Applications, of VBA), CSharp and POSIX. This book focuses specifically on just one, the Java version. Why would you use Java over the other API technologies? The main reason might be that you are a Java expert. If you don't know Java or any other programming language, you should take a look at the Excel/DDE API, which has a much smaller learning curve. But if you know Java, this platform offers more flexibility than the DDE for Excel, is supported on Windows, MAC, and Unix/Linux (the DDE is only supported in Windows), and provides very high performance.

For more information about our APIs, see the Trading Technology > API Solutions page on our web site.

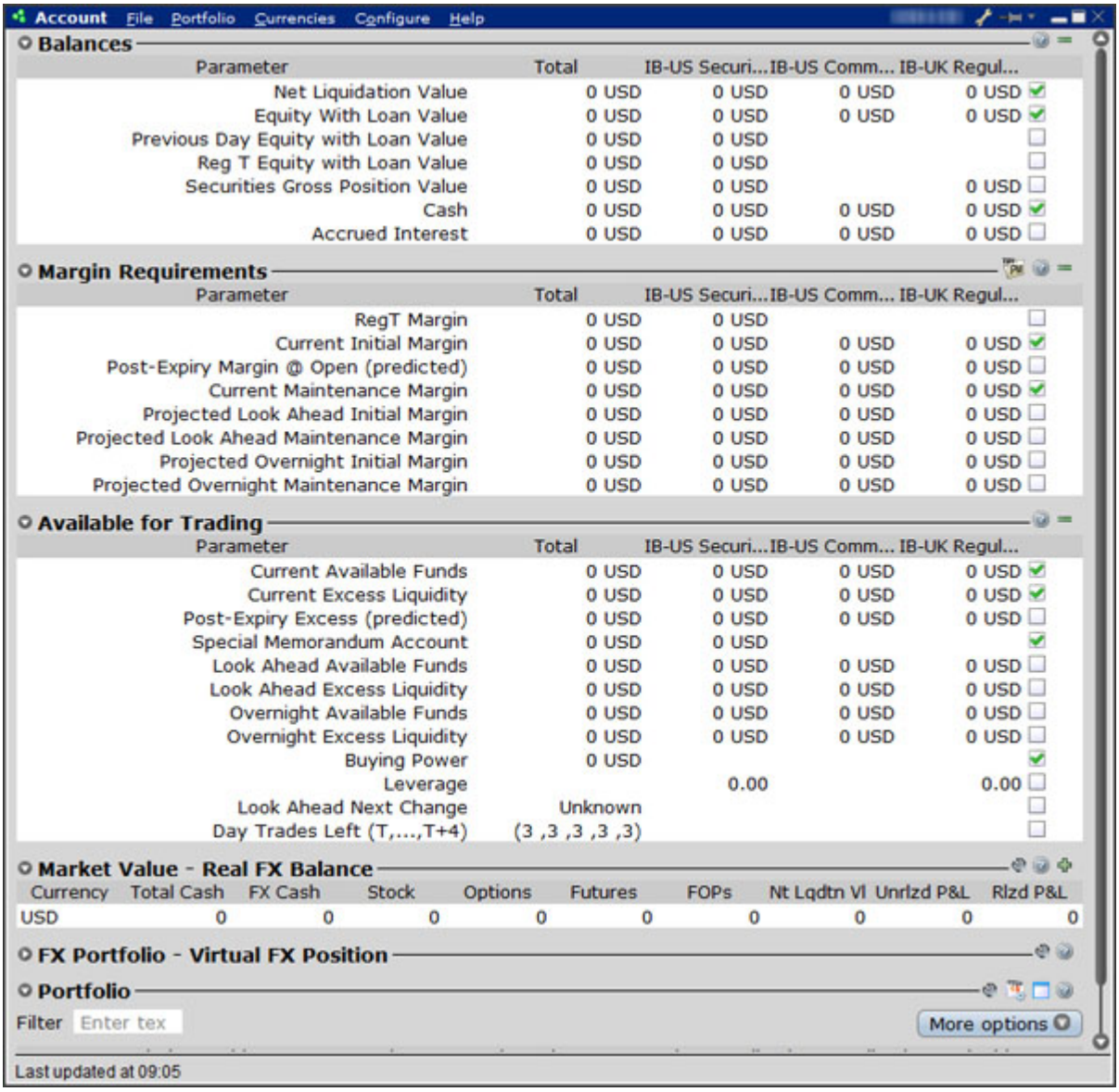
An Example

It's always easier to understand something when you have a real life example to contemplate. Following is a simple situation in which the API could be used to create a custom result.

TWS provides an optional field that shows you your position-specific P&L for the day, as either a percentage or an absolute value. Suppose you want to modify your position based on your P&L value? At this writing, the only way to do this would be to watch the market data line to see if the P&L changed, and then manually create and transmit an order if you happened to catch the value at the right point. Now, enter the API! You can instruct the API to automatically trigger an order with specific parameters (such as limit price and quantity) when the P&L hits a certain point. That's power! (See Chapter 20 for details on how to actually implement this feature).

Another Example

Another nice benefit of the API is that it gives you the ability to use the data in TWS in different ways. TWS provides an extensive Account Information window that's chock-full of everything you'll ever want to know about your account status. The thing is, it's only displayed in a TWS window, like the one below.



But, what if you wanted to do something else with this information, like have it reflected in some kind of spreadsheet where you log information for all accounts that you own, including your checking account, Interactive Brokers' account, 401K, ROIs, etc? Again - enter the API!

You can instruct the API to get any specific account or execution information and put it wherever it belongs in a spreadsheet. The information is linked to TWS, so it's easy to keep the information updated by simply linking to a running version of TWS. With a little experimenting and some help from the TWS Users' Guide, you'll be slinging data like a short-order API chef in no time!

Why Use the DDE for Excel API?

IB's API is supported in multiple programming languages including: Java, C++, and Active X, and VBA for Excel. This book focuses specifically on just one, the Excel version. Why would you use Excel with the API?

Here are some of the strongest arguments in support of using the DDE for Excel API, at least to start:

- Its interface is familiar and inspires confidence!
- Since this API uses Microsoft Excel as its development environment, most users already have at least a basic familiarity with its functionality. You may not know exactly how to write Visual Basic for Applications code, but you can see values in a cell, and with a little coaching will be able to view the sample code behind the stuff we give you and learn from that.
- It comes with a fantastic sample program that can and should be used as a starting point for anything else you want to do.
- The Excel sample application provides a series of worksheet pages populated with sample data, each geared toward helping you implement some specific action, like getting market data, sending orders or viewing execution reports.
- The learning curve is small and the time to development is fast. You're not going to need a programming course before you get started here. If you have some basic PC skills and a good idea of what you want to do, you could create and implement an API interface in a matter of days. In general, the time it will take you to implement your new API is minimal compared to our other API technologies.

In order to use the API, you need to have TWS running, and must include your login name on each function page of the Excel spreadsheet.

There are a few other things you must do before the DDE for Excel API will run. The next chapter gets you geared up and ready to go.

3

Preparing to Use the DDE for Excel API

Although the API provides great flexibility in implementing your automated trading ideas, all of its functionality runs through the Trader Workstation. This means that you must have a TWS account with IB, and that you must have your TWS running in order for the API to work. This section takes you through the minor prep work you need to complete, step by step.

Here's what you'll find in this section:

- Chapter 3 - Download the API Software
- Chapter 4 - Enable TWS to Support the API
- Chapter 5 - Using the Sample DDE for Excel API Application

The TWS DDE for Excel API runs ONLY on the Windows platform. It does not run on Mac or Unix platforms.

Chapter 3 - Download the API Software

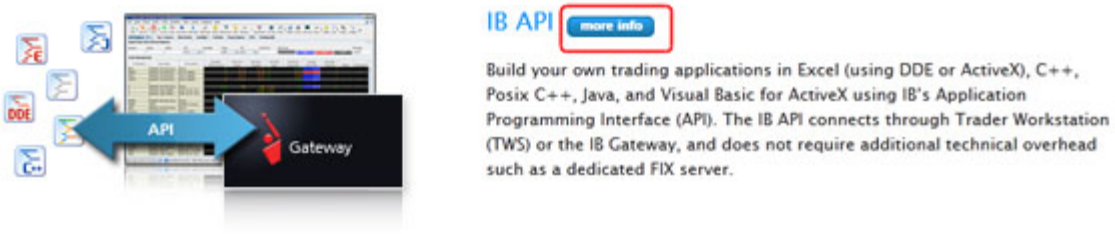
If you don't have an account with IB, you can run a lot of the following tasks using the Demo TWS system. It will have some limitations, but you'll get a feel for things. If you have an account, we recommend opening a PaperTrader version, which simulates the TWS trading environment, and gives you $100,000 in phantom cash to trade with.

Next, you need to download the API software from the IB website.

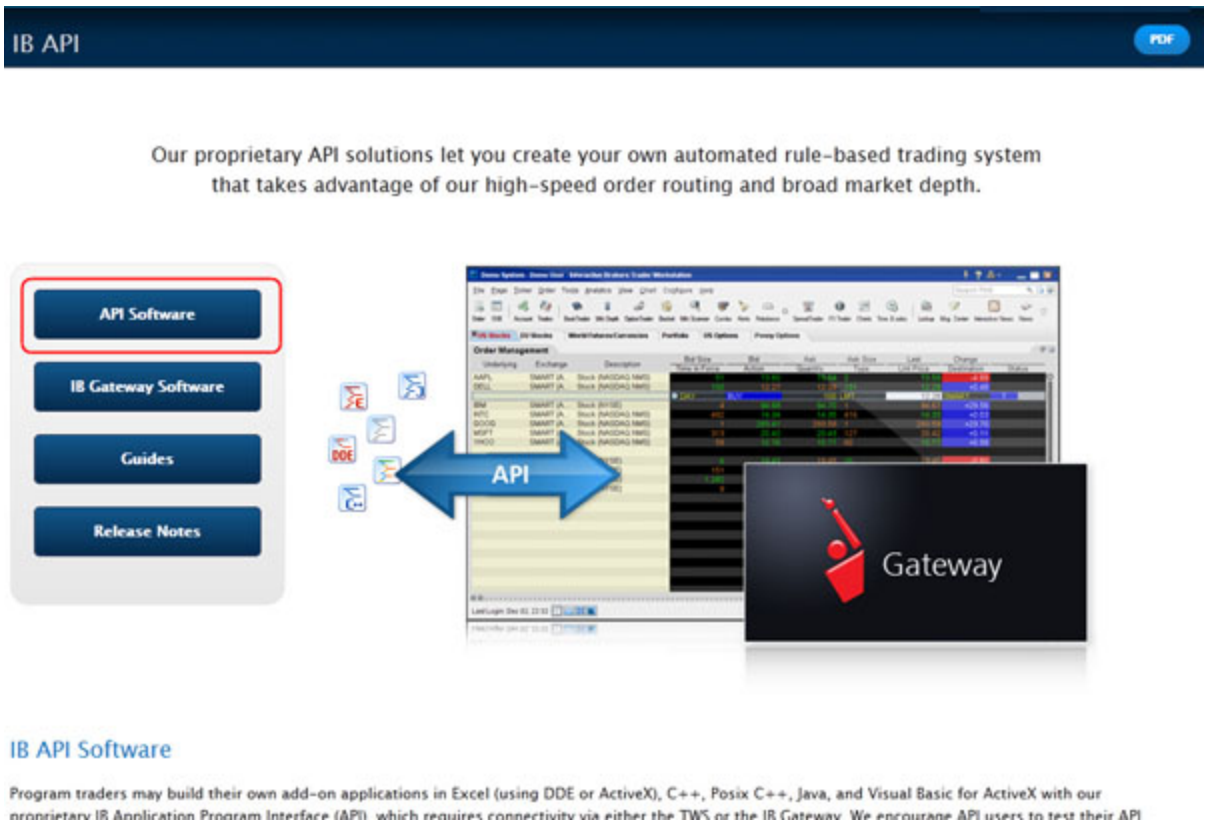
Step 1: Download the API software.

This step takes you out to the IB website at https://individuals.interactivebrokers.com/en/index.php?f=1325. The menus are along the top of the homepage. Hold your mouse pointer over the Trading Technology menu, then click *API Solutions*.

On the API Solutions page, click the **more info** button next to IB API.

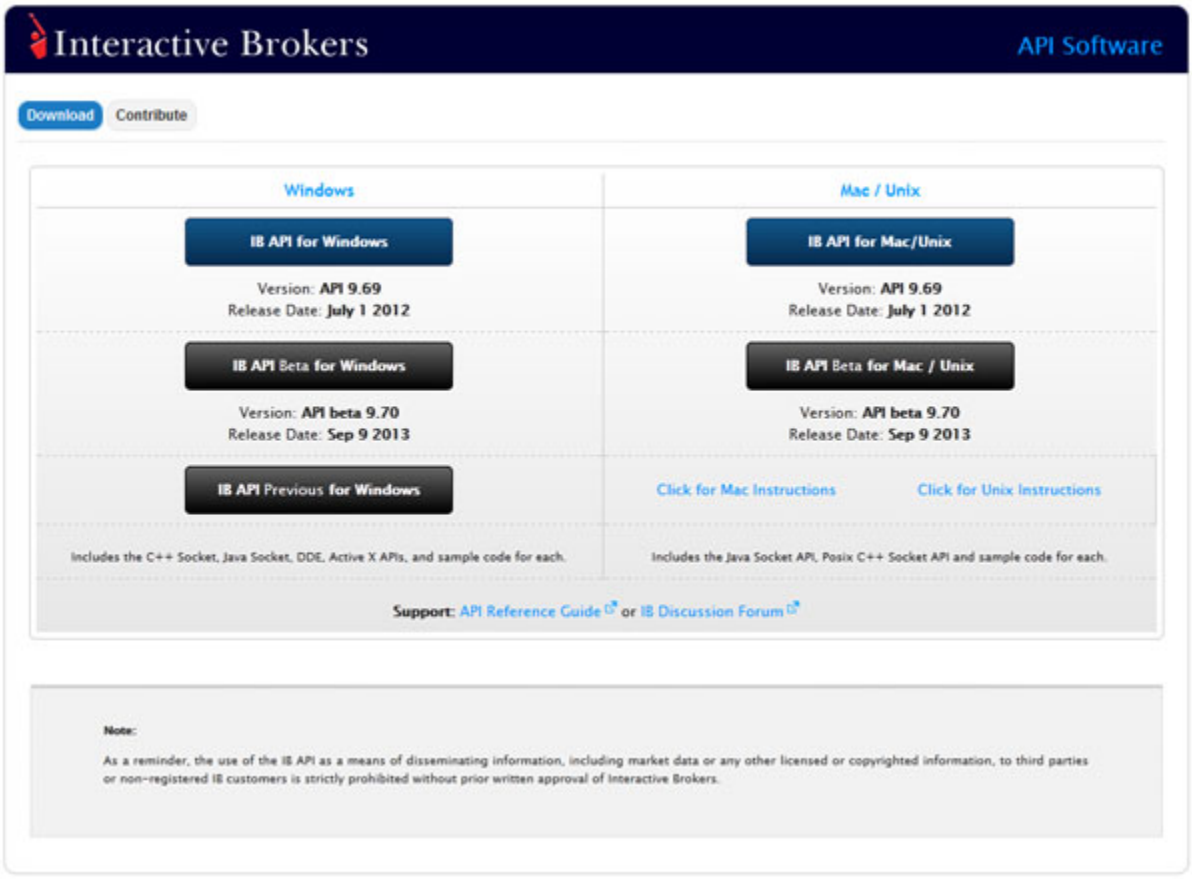


On the next page that appears, click the **API Software** button.



Click the **I Agree** button on the license agreement page to open the API software download page.

This displays the IB API page which shows a table with buttons that initiate the API software download process for Windows, MAC or Unix platforms. When available, there will also be a Windows Beta version of the software. Find the OS you need, then click the button to download the API installation program.



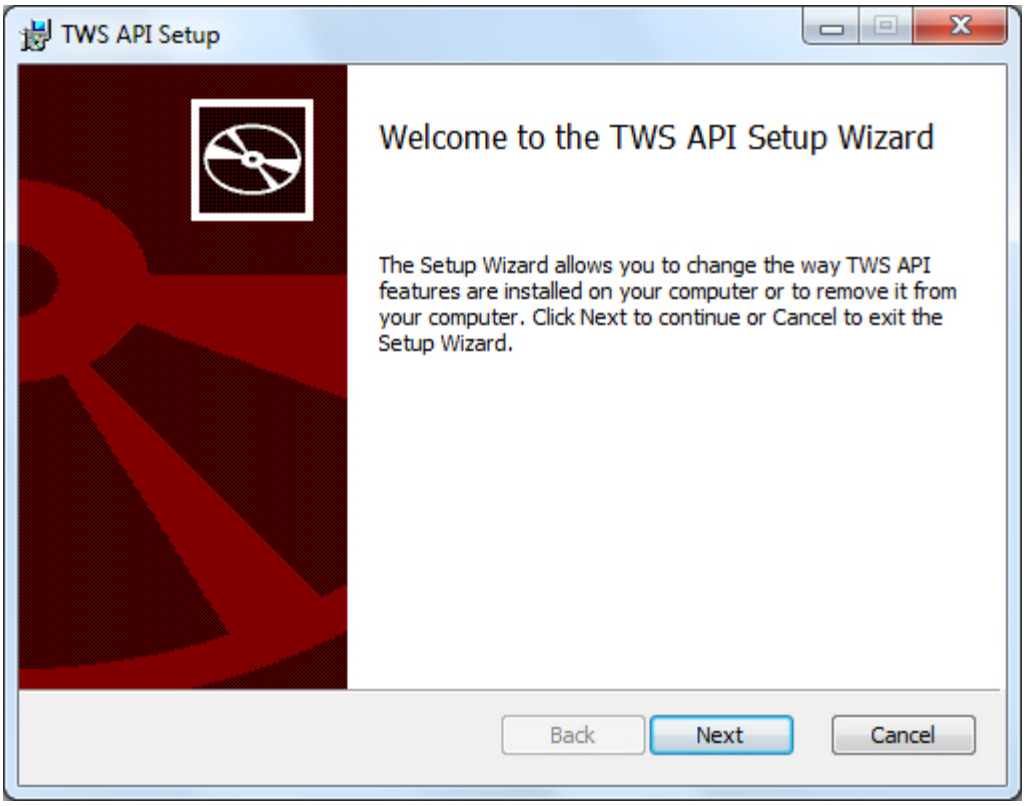
For this book, we assume that you are using Windows. If you're using a different operating system (Mac, Unix), be sure to adjust the instructions accordingly!

In the Windows column, click the **IB API for Windows** button. This opens a File Download box, where you can decide whether to save the installation file, or open it. We recommend you choose *Save* and then select a place where you can easily find it, like your desktop (you choose the path in the Save in field at the top of the Save As box that opens up). Once you've selected a good place to put it, click the **Save** button. It takes seconds to download the executable file. Note that the API installation file is named for the API version; for example, *TWS API Install 9.69.01.msi*.

*We'll usually be stressing just the opposite, but at this point, you need to make sure TWS is **not** running. If it is, you won't be able to install the API software.*

Step 2: Install the API software.

Next, go to the place where you saved the file (for example, your desktop or some other location on your computer), and double-click the API software installation file icon. This starts the installation wizard, a simple process that displays a series of dialogs with questions that you must answer.



Once you have completed the installation wizard, the sample application installs, and you're ready to open the Excel sample spreadsheet, connect to TWS, and get started using the DDE for Excel API!

Chapter 4 - Enable TWS to Support the API

Enabling TWS to support the API is probably the simplest step you'll encounter in this book. It's probably more difficult to remember to log into TWS before you run the API!

Log Into TWS

OK, first let's get TWS started. Log into your account, or even better, log into your PaperTrader account. If you don't yet have a paper trading account, go to the IB website and look at the PaperTrader page under the software menu. The PaperTrader account gives you access to everything you can get through your regular account, but you use phantom money to trade, instead of your hard-earned real money.

About Logging In

Logging into TWS is easy. You can run the TWS from your Internet browser (which is the recommended method), or download the software to your PC and launch it directly from your desktop as a standalone application.

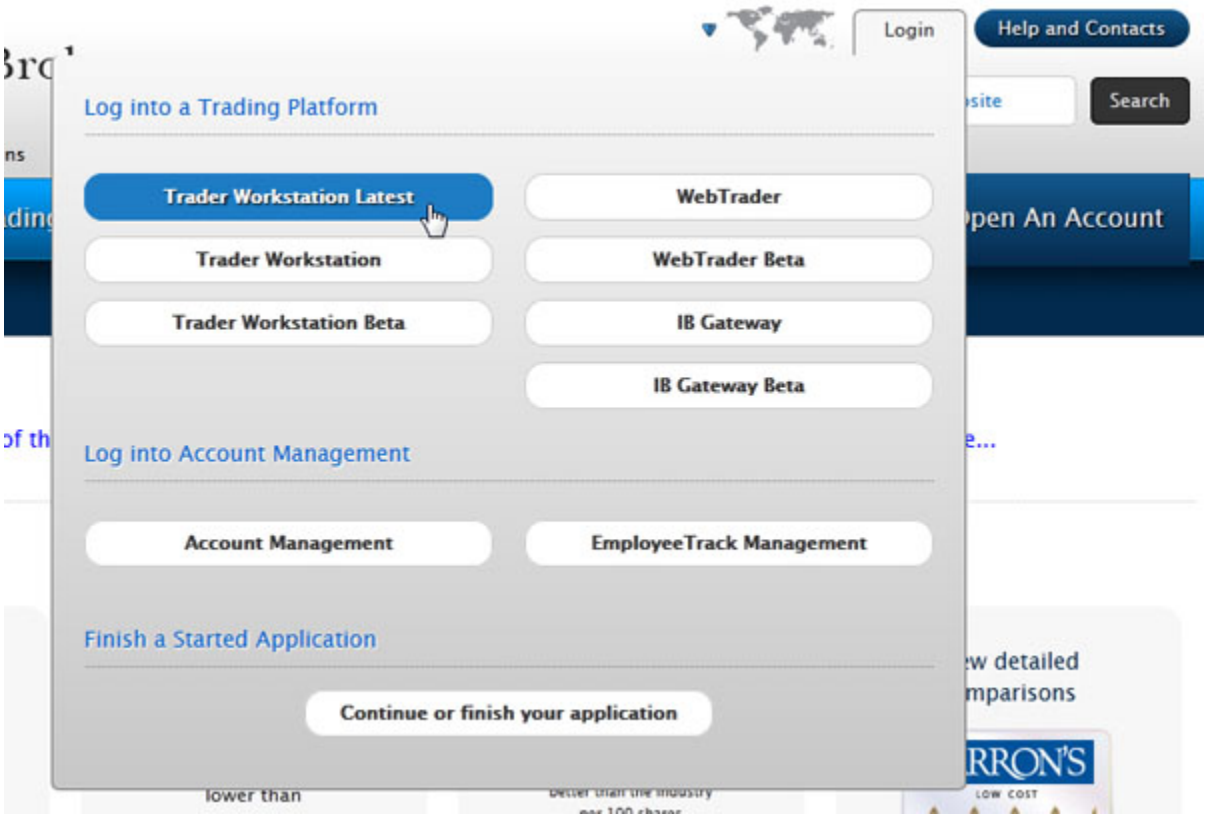
The browser-based version:

- Allows you to access your account and execute trades from any Java™-enabled internet browser.
- Is always running the latest release.
- Allows you to save your settings from your primary machine to a server so that your TWS will look exactly the same regardless of what Internet machine you use to log in.

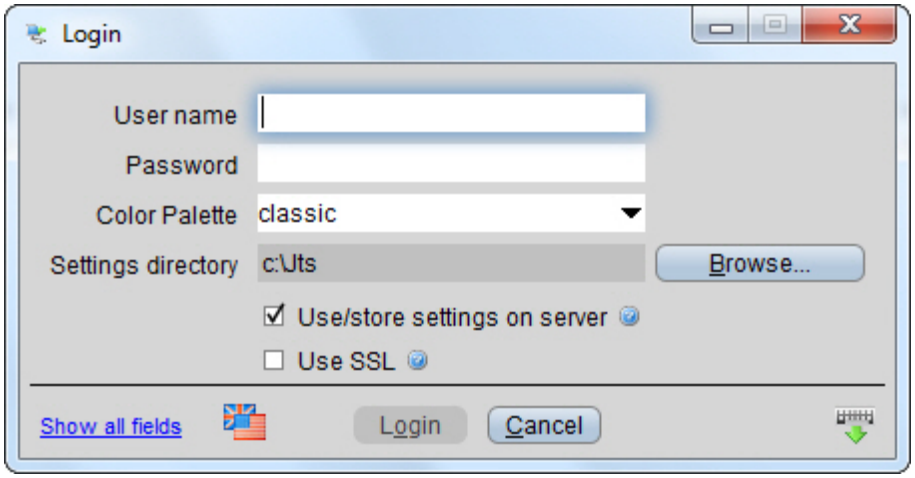
The standalone version uses less memory and may run faster, but requires you to download each release to take advantage of new features. To download to your PC, see the Installation Instructions on the website.

To log into TWS:

1 From the Login menu on our website, select **Trader Workstation Latest** from the drop-down list. You can also log into the previous version of TWS and the current TWS Beta version.



2 In the Login box, enter your username and password, and click **Login**.



Login Box Options:

- The color palette allows you to choose a new color skin for TWS. If you select a palette and want to change it once you have logged in, use the Display> Style page in Global Configuration.
- Settings Directory - By default, TWS files are saved in C:\Jts. If you would like to change the location of your settings files, use the Browse button to specify a new directory.
- Use/store settings on server - This option allows you to save your settings, including market data, pages etc., on our server instead of on your local machine. If checked, your workstation will have the same data and look regardless of your login location. If unchecked, settings are only saved on the computer you are currently using.
- Use SSL - Your identity and trading information are always protected. Checking the Use SSL option gives added privacy and integrity to your order information as it travels over the Internet. Minor performance impacts may be experienced depending on the capabilities of your PC.
- Click Show all fields to select a different language for TWS and to have the system migrate settings that you may have saved under a different user name.
- Mouse-driven login - Click the Keyboard icon in the title bar to enter your username and password using the mouse instead of your computer keyboard. Use the mouse to click the appropriate keys on the clickable keyboard display.

Enable the API Connection through TWS

To run the API through TWS, you must always have your system running and it must be configured to use any of the API components.

To enable API connection through TWS

1. On the **Edit** menu in TWS, select *Global Configuration*. Then select **API** in the left pane, then click **Settings**.

2. In the right pane, click the check box for *Enable DDE Clients*. You must have this setting enabled to connect to the API through TWS.

3. You also need to ensure that you have Microsoft Excel installed on your computer. If you don't have it, go to the Microsoft website to order it. If you're using Word or have MS Office installed, you probably already have it..

You’re all set and you can now connect the Excel sample spreadsheet to TWS and start learning about all the great features supported by the DDE for Excel API!

Chapter 5 - Using the Sample DDE for Excel API Application

OK, you've got all the pieces in place. By the end of this chapter, you'll actually be inside the sample API application, all ready to start receiving market data, create and transmit orders, and do lots of other cool stuff. The sample application is actually a complete application on its own, and provides everything you need to get started and more. As a matter of fact, your best bet is to use this file as the starting point in creating your own DDE for Excel API. We'll show you how to do that after we get you inside.

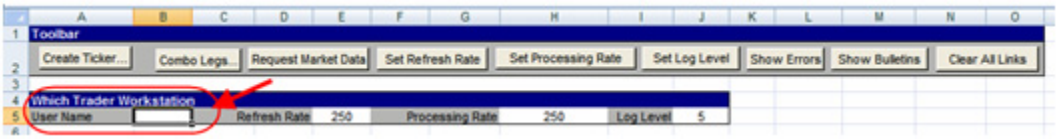
This next section shows you where to go to open the sample application and hook up the pages to TWS. Be patient with this prep work. Before you know it, you'll be in Chapter 6, where you'll finally get to use the spreadsheet pages in the DDE for Excel API application.

Step 1: Open the application and enable macros.

At this point, you should have TWS up and running. Now use your Windows explorer to find the Excel file. If you didn't change the Save location, you'll most likely find your application, which is called TwsDde.xls, under My Computer in C:\TWS API X.XX\samples\Excel\TwsDde.xls, where X.XX is the API version number. Double-click this file icon, and when you see a message about macros, be sure you click the button that says Enable Macros. If you don't enable the macros, the sample application will be useless to you!

Step 2: Connect each page to TWS.

The first page you'll see is the Tickers page. At the top, you'll see a dark blue section header that says Toolbar. This gives you all the action buttons you'll need to use the features on the page. But for now, you need to focus on the next section, or the blue header that says "Which Trader Workstation." This section has an editable field titled "User Name." This important field is the place that will connect you to TWS when you enter the username of the workstation you're running. If you enter the wrong username, or forget to enter anything, you won't get connected and nothing will work!



So, enter your login Username (not your password, just your username). Since you'll have to enter this username on most of the other pages too, we recommend taking a few minutes to copy the name from this field, use the tabs at the bottom of the worksheet to move from page to page, and paste your username in each occurrence of the User Name field.

Step 4: Save the Excel spreadsheet with another name.

We mentioned earlier that this sample spreadsheet provides the perfect starting point for developing your own DDE for Excel API. Since you have so much in here to work with, we think it makes sense to save this sample spreadsheet with a new name so that it becomes your own file.

If you were to keep the existing name in the current folder, the next time you downloaded a new version everything you'd done would be deleted and replaced with latest software. Giving it a new name allows you to keep all of your changes intact.

You've done it! You've enabled TWS, downloaded and installed the software, and opened the Excel sample application. Let's move on to Chapter 6, and start having some fun!

4

Getting to Know the DDE for Excel API Spreadsheet

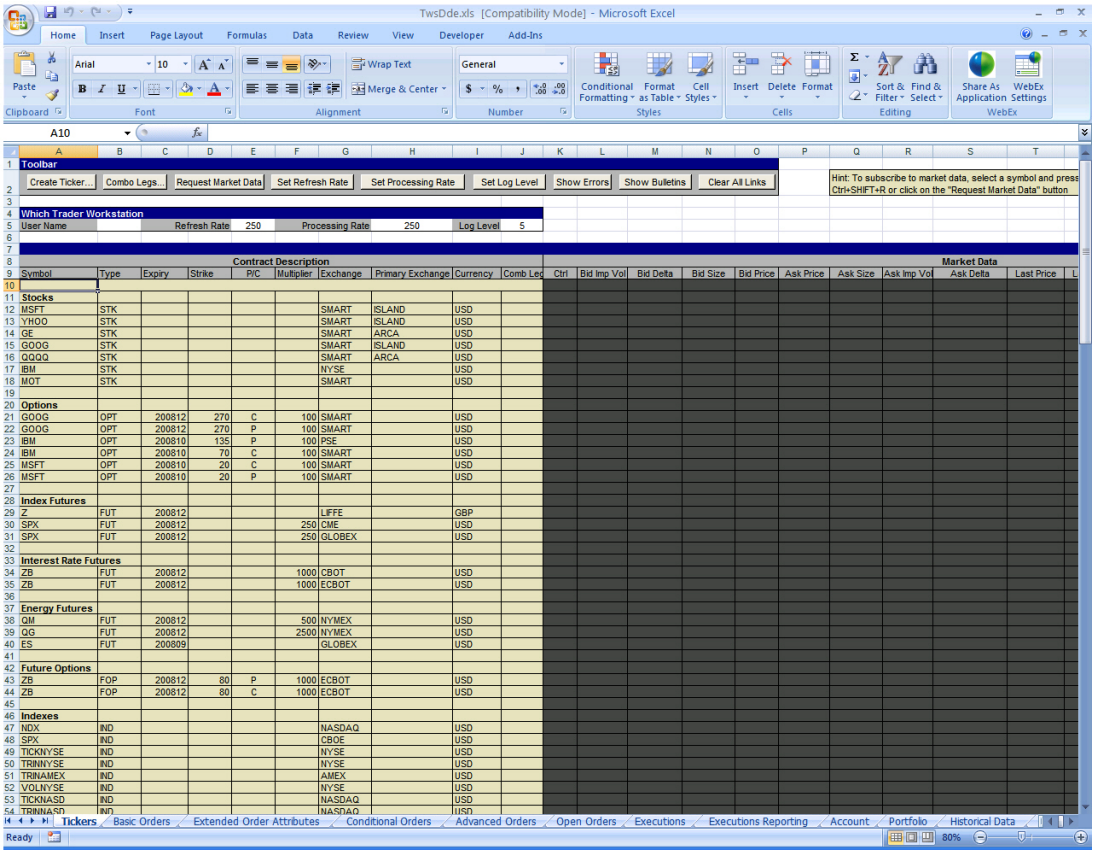
By now you should have the "working" version of the Excel sample spreadsheet up and running. Now comes the fun part - becoming familiar with all the cool features. The sample pages give you lots of functionality without making you figure out how to implement it. More importantly, you can look at the underlying Visual Basic code to see how it was done, and use some of it to get starting implementing new creative features of your own!

Here's what you'll find in this section:

The TWS DDE for Excel API runs ONLY on the Windows platform. It does not run on Mac or Unix platforms.

Chapter 6 - Introducing the Excel Pages

When you first open the sample spreadsheet, you'll find that our development team has included data on many of the pages to help you better understand how they work. You can use or delete the data as you want, and add your own. Here's what you see when you first open the spreadsheet:



You can see examples on the Tickers page of how to create tickers for stocks, options, futures, future options, indexes, currency and combination lines. In the next chapter, "The Tickers Page," we'll show you how to set up your own ticker lines and request market data.

The Pages

When you open the spreadsheet, you'll see a row of tabs along the bottom of the worksheet. Each of these tabs opens a page, and each page is dedicated to a specific function or subset of functions. Many of the titles are self-explanatory, but we've summarized them all in the following table.

DDE for Excel API Spreadsheet Pages	
Page	**Description**
Tickers	Set up ticker lines and create links to real-time market data.
Basic Orders	Send, manage, edit and cancel basic orders.
Extended Order Attributes	Includes all optional order attributes, such as a good 'til date time in force, hidden and discretionary orders, volatility orders and others. Apply them to specific orders as a template or use as a default by all orders.
Conditional Orders	Similar to the orders page, but also includes a section for conditional statements which allows you to create orders that are dependent on something else, such as the execution of another order or a change in price.
Advanced Orders	Send, manage, edit and cancel advanced orders. These are orders that require the use of extended order attributes.
Open Orders	Subscribe to all open orders, whether you created them via the API or TWS.
Executions	Displays all of your order executions.
Executions Reporting	Specify execution report criteria and display only those execution reports that meet your criteria.
Account	Displays all account information in real-time, including equity with loan value, available funds, and your current portfolio value.
Portfolio	Displays positions for all asset types using an array query (or, a subscription!).
Historical Data	Displays historical data for an instrument.
Market Scanner	Scans the desired markets to find instruments that meet your criteria and displays the results on a separate page in descending order. The data continuously updates in real time.
Contract Details	This is sort of a research page where you can request information on a specific instrument, like the Contract ID, the support order types on specific markets, the minimum ticks and the trading class.

DDE for Excel API Spreadsheet Pages	
Page	**Description**
Bond Contract Details	Similar to the contract details page, but focuses specifically on bond details, like the CUSIP, maturity, interest rate and bond ratings.
Market Depth	Displays a list of alternative bids and offers away from the inside quote, to help you better gauge market liquidity.
Advisors	Send, manage, edit and cancel Financial Advisor orders.
Old Style Executions	This is the old Executions page, which is no longer used. It has have been left intact to allow those with code connecting to these pages to revise it and work off the newer pages without losing functionality. We don't recommend using this page to start new projects!
Old Style Account-Portfolio	This is the old Account-Portfolio page, which is no longer used. It has have been left intact to allow those with code connecting to these pages to revise it and work off the newer pages without losing functionality. We don't recommend using this page to start new projects!

Each page includes a row of action buttons at the top. Clicking these buttons tells Excel to run a "macro," which is basically a piece of Visual Basic code designed to make something specific happen in your spreadsheet, like populating your ticker line with live market data. In Section IV, "Creating Your Own API Application", we'll show you how to create and use this code to completely customize your application!

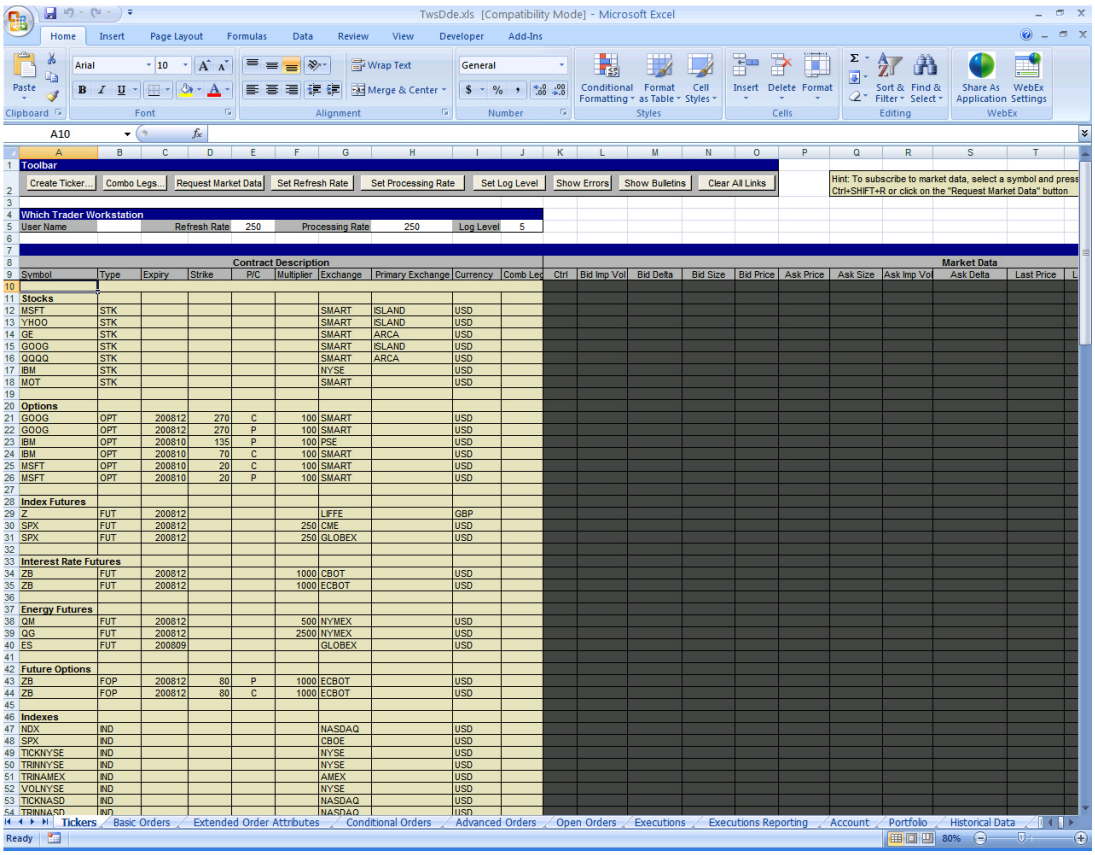
The functionality on these pages is linked to TWS. That's why you need to be sure that TWS is running and that you have entered your username on any page with a Which Trader Workstation? field at the top left.

Common Action Buttons

Although each page is geared toward accomplishing a different set of tasks, they all contain some common elements, including some of the same action buttons. The table below describes these common buttons.

Button	Description
Combo Legs	Allows you to identify individual legs to be sent as a combination order.
Set Refresh Rate	The Refresh Rate value is in milliseconds, and determines how often the DDE link to TWS is refreshed. The default refresh rate is 1000 (updates every 1 second), and the allowed range is 100 to 2000, inclusive. Note that the TWS market data updates every 300 milliseconds. This means the default "every 1 second" rate will only show 30% of the ticks. A Refresh Rate of 250 will get every tick to the spreadsheet…
Set Processing Rate	Set the TWS/DDE server message processing rate (also in milliseconds) to affect the speed at which DDE will handle requests between the spreadsheet and TWS. The allowed range is 100ms to 2000ms, inclusive.
Show Bulletins	Click this button to subscribe to IB News bulletins. Once you have selected the data source subscription (your username) you will see a red bulletin area at the top of your page.
Set Log Level	This specifies the level of log entry detail used when processing API requests. Valid values include: 1 = SYSTEM 2 = ERROR 3 = WARNING 4 = INFORMATION 5 = DETAIL
Show Errors	Jumps to the Error Code field and shows the Last Error.
Clear All Links	Removes all links to TWS. Use this button to prepare your spreadsheet for saving to disk.

Chapter 7 - The Tickers Page



The Tickers page happens to be the first page you see when you open the Excel sample application. Again, remember that you MUST have TWS running, and you must enter your TWS user name in the *Which Trader Workstation?* area before you'll be able to get data.

Define a Ticker

OK, the Tickers page is designed for two main tasks: first, to define tickers, and second, to get market data for those tickers. To get started, we'll enter a ticker for a stock, and then request market data. You can see that the TWS API development team has included a bunch of sample rows to show you where to enter your information. We'll create a new blank row. When you select any of these rows you need to select from the row number in the far left column. Otherwise, you'll only be selecting a single **field** and many of the programmed actions won't work. So, let's select any row by clicking the mouse on one of the row numbers, then use your right-click button and select Insert. Voila, you have a new blank row!

OK, let's do it!

1. First, put your mouse in the *Symbol* field and enter an underlying symbol, for example IBM[1].

2. Tab over to the next column, *Type*, and enter **Stk**.

3. Since you want market data for a stock symbol, you don't need the next four columns (Expiry, Strike, P/C and Multiplier) so tab past them and get to the *Exchange* field. This field defines where the market data comes from. For example, enter **NYSE**.

   The next field, *Primary Exchange*, is optional, and is used to help to identify the product, especially in cases where the same symbol may be used on different exchanges with different currencies, and you have used Smart as the exchange.

4. Now tab to the currency field and enter the currency in which the product trades (for example, **USD** for US dollars). This field also helps define the correct symbol.

OK, now you need to…wait a sec. You're done! You've successfully entered a ticker line for a stock product. Now, want to get some market data running against this ticker? OK, but before you do this, you should know that you can also define tickers using the Ticker box, which pops up when you click the Create Ticker button. Tab through these fields just as you did on the page row, and click OK to put the ticker row on your screen.

Request Market Data

Now, on to the market data. This is even easier than creating the ticker. First, make sure that you have TWS running, and make sure you have entered the appropriate user name in the Which Trader Workstation? User Name field.

OK, are you ready? Let's begin.

1. Click on the row number in the farthest left column to select the entire row.

2. Now, move your mouse up to the top of the page and click the button that says **Request Market Data**.

That's it. You should now be seeing a row of market data running across all of the black fields that are applicable to the type of product you entered on the line. So in this case, you won't see any data in the Implied vol or Delta fields, or any of the other fields that are used for other asset types.

Want to do it again? Let's get data for all of the sample tickers included. If you click in the first row and then click the **Request Market Data button**, each line will be filled in, and you will be moved to the next line automatically. Keep clicking the button and soon you'll have data for the whole page. That's all there is to it!

Create a Group Combination Strategy

Now, there's one other function this page allows. You can mark any of these ticker lines as belonging to a group combination strategy.

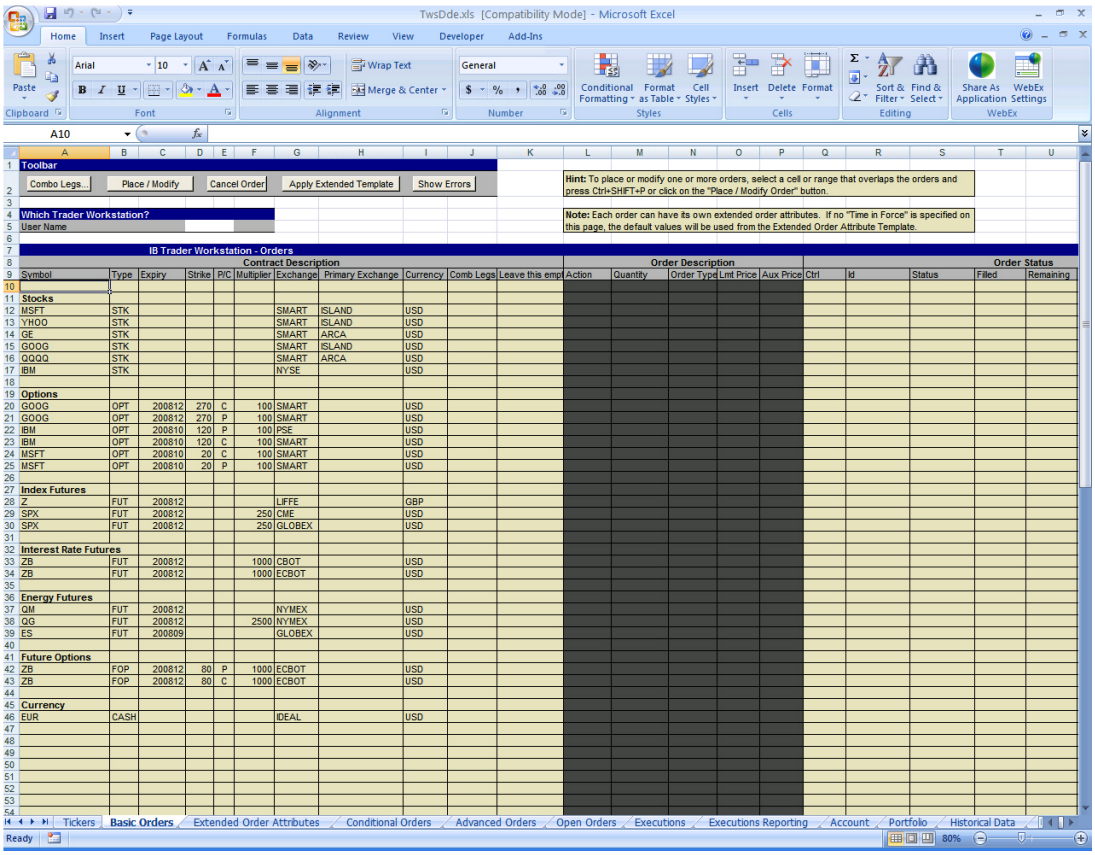
Simply select a ticker line and then click the **Combo Legs** button that sits right in front of the **Request Market Data** button. You'll get a little dialog box that helps you define the characteristics of the leg, such as the ratio of this leg to the rest, the side (buy or sell), the exchange, and whether the order will open or close a position.

After you have defined the parameters, click the **Add** button to see the leg in the Combination Legs box. Then click **OK** to close the box, and the *Combo Leg* field on that ticker row will now be populated to identify the ticker as part of a combination order.

We've now covered all of the Tickers page functionality. If you want to review the definitions of the remaining common buttons, review the table earlier in this chapter. Otherwise, let's proceed to creating orders from the Basic Orders page!

Chapter 8 - The Basic Orders Page

In TWS, you view market data and send orders from the same page, but in the DDE for Excel API you have a separate page for Basic Orders and Advanced Orders. The difference between these order pages is that advanced orders require you to use extended order attributes. But since we haven't talked about extended order attributes yet, you're not ready for advanced orders. For now, click the Basic Orders tab on the bottom of the spreadsheet, and take a look.



You see a bunch of lines that look very similar to the ticker lines you entered on the Tickers page. In fact, they are just like those entries. The difference is that with these ticker entries you'll be submitting orders instead of viewing market data. Like most pages in the sample spreadsheet, many of the rows are already populated to help you understand how the rows and fields work. You might however, prefer to have these rows populated with the same tickers you entered on the Tickers page, so that you can monitor prices and submit orders on the same symbol from two different pages.

As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

Add Your Tickers to the Basic Orders Page

To add your tickers from the Tickers page to the Basic Orders page, simply copy and paste your ticker lines from one page to another.

1. On the Tickers page, select a row by clicking the row number in the far left column, then right click and choose *Copy*. The selected row starts blinking away likes it's on a marquee. For example, copy the IBM ticker line.

2. Now, come back to the Basic Orders page, select a blank row by clicking in the number field of a blank row, right click and choose *Paste*.

Depending on which version of Excel you are using, you can also choose *Insert Copied Cells* to paste the copied ticker row a blank row; this command automatically moves the other rows down to make room for the new row.

You've just created a new potential order row. You can also enter a new ticker symbol manually just as you did on the Tickers page.

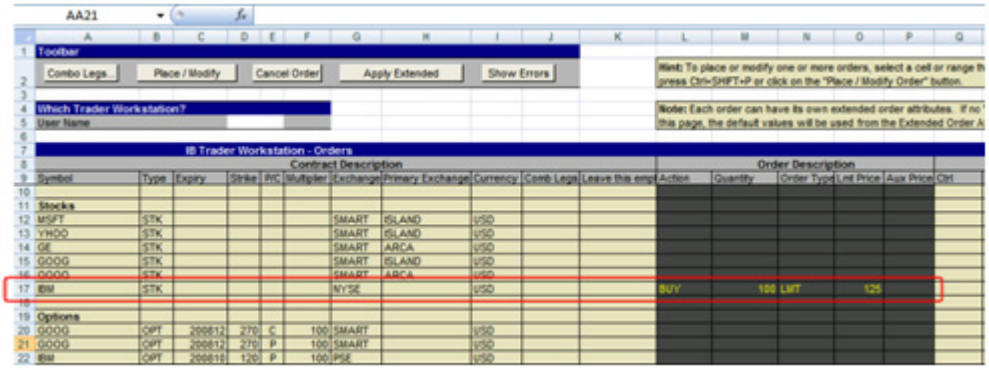
Create an Order

Now you'll create an order for the IBM[1] stock ticker, which you entered by copying and pasting its ticker row onto the Basic Orders page.

1. First, click your mouse in the *Action* field (the first of the black fields) on the IBM[1] ticker line, and enter **Buy** or **Sell**.

2. Tab through the fields, and enter the order *Quantity* (let's use 100 for this example), the *Order Type* (we'll use Limit - you can see a list of valid order types in a bulleted list on the next page.

3. Since it's a limit order, you need to enter a limit price. Take a look at the IBM[1] market data and come up with a reasonable limit price. Note the *Aux Price* field. This is only used for certain order types, such as relative orders, that need two prices. You don't need to use it for a basic limit order.



Order Types

The order types currently supported through the DDE for Excel API are:

- Limit (LMT)
- Market (MKT)
- Limit if Touched (LIT)
- Market if Touched (MIT)
- Market on Close (MOC)
- Limit on Close (LOC)
- Pegged to Market (PEGMKT)
- Relative (REL)
- Stop (STP)
- Stop Limit (STPLMT)
- Trailing Stop (TRAIL)
- Trailing Stop Limit (TRAILLIMIT)
- Volume-Weighted Average Price (VWAP)
- Volatility orders (VOL)

We recommend you take a look at the Order Info section on the Trading menu at http://individuals.interactivebrokers.com/en/p.php?f=orderTypes&o&ib_entity=lic.

Now you've entered all of the basic order information. If you transmitted your order right now, it would be valid and a Day time in force would be used. However, you have much more flexibility than that, if you care to explore it, with Extended Order Attributes. We'll go into them in detail in Chapter 9 (and in Chapter 11 when we talk about those mysterious advanced orders), since they have their own tab, but the DDE for Excel API makes these attributes available on the Basic Orders page so that you can customize them order by order. Scroll past the Order Status section and take a quick look at the attributes in the second black section. Here's where you enter a good-till-cancelled time in force, or mark orders as part of an OCA (one cancels all) group. You can skip ahead if you want to read about these now, or scroll back and look at the beige Order Status section of the Basic Orders page.

Order Status Fields

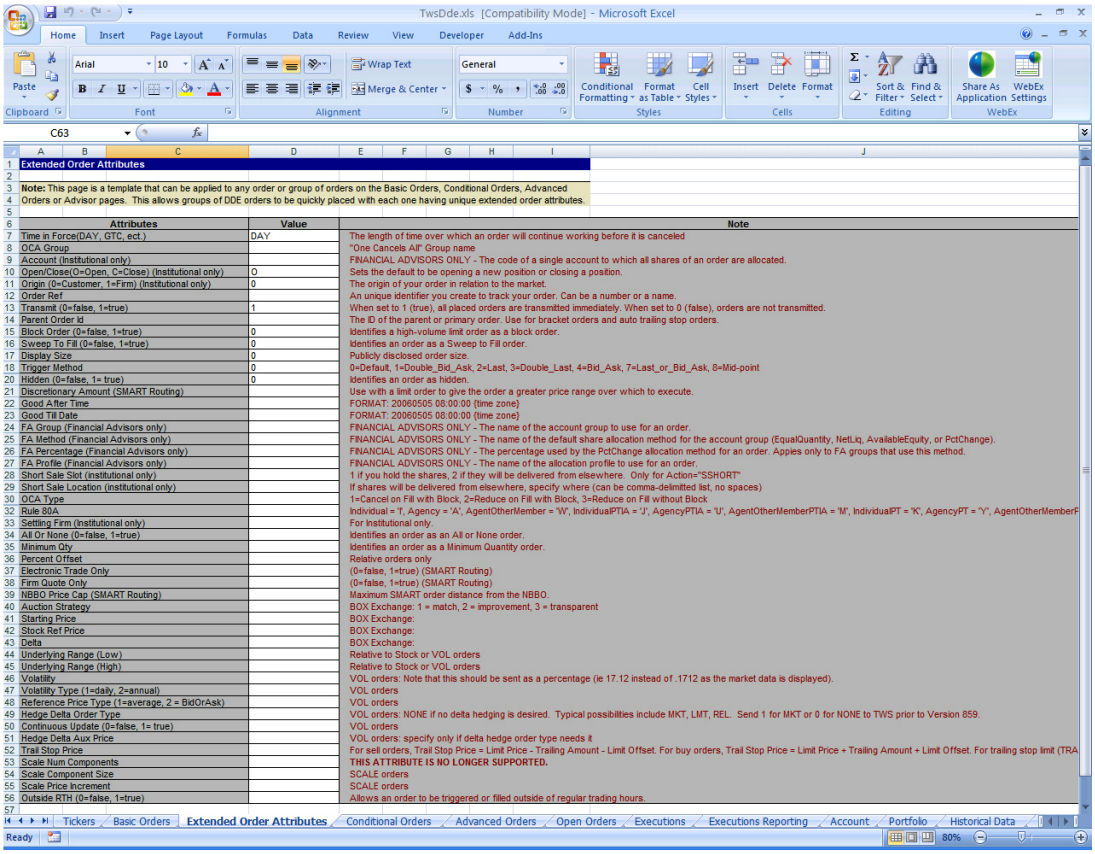
The Order Status fields let you know what's happening with your order. If you want to see this in action, change the limit order type to a market order, delete the limit price and send the order. How do you do that? You may have already figured out that you simply select the order line, make your changes and click the **Place/Modify Order** button. It may move too quickly for you to see, but if you're lucky you'll see the *Status* field display *Submitted* then change to *Filled*, and you'll note the *Filled* field go from 0 to 100. If you'd submitted a larger order that didn't get completely filled, you'd have values in both the *Filled* and the *Remaining* fields. We also show you the average price of the filled portion of the order, and the price of the last fill.

You can also modify and resubmit the order BEFORE it executes by selecting the order line, making any changes (say you want to increase the quantity, or change the order type) and then clicking the **Place/Modify Order** button again.

Once your order is filled, you can find more details on the Executions page, which we'll discuss in Chapter 13. Now let's take a look at those totally cool Extended Order Attributes!

Chapter 9 - The Extended Order Attributes Page

These attributes are completely optional, but you'll find yourself perusing and using at least some of them as you become more comfortable with the IB way of trading. Some of these attributes are only applicable to Institutional traders and others only to advisors, and these are well marked. If you click the **Extended Order Attributes** tab, you'll notice that this page looks quite different from all of the other pages. It doesn't look like the normal data page, instead it's a list with the attribute and some notes sandwiching the active *Value* field, where you enter your input to control the attributes.



Attributes	Value	Note
Time in Force(DAY, GTC, ect.)	DAY	The length of time over which an order will continue working before it is canceled
OCA Group		"One Cancels All" Group name
Account (Institutional only)		FINANCIAL ADVISORS ONLY - The code of a single account to which all shares of an order are allocated.
Open/Close(O=Open, C=Close) (Institutional only)	O	Sets the default to be opening a new position or closing a position.
Origin (0=Customer, 1=Firm) (Institutional only)	0	The origin of your order in relation to the market.
Order Ref		An unique identifier you create to track your order. Can be a number or a name.
Transmit (0=false, 1=true)	1	When set to 1 (true), all placed orders are transmitted immediately. When set to 0 (false), orders are not transmitted.
Parent Order Id		The ID of the parent or primary order. Use for bracket orders and auto trailing stop orders.
Block Order (0=false, 1=true)	0	Identifies a high-volume limit order as a block order.
Sweep To Fill (0=false, 1=true)	0	Identifies an order as a Sweep to Fill order.
Display Size	0	Publicly disclosed order size.
Trigger Method	0	0=Default, 1=Double_Bid_Ask, 2=Last, 3=Double_Last, 4=Bid_Ask, 7=Last_or_Bid_Ask, 8=Mid-point
Hidden (0=false, 1= true)	0	Identifies an order as hidden.
Discretionary Amount (SMART Routing)		Use with a limit order to give the order a greater price range over which to execute.
Good After Time		FORMAT: 20060505 08:00:00 {time zone}
Good Till Date		FORMAT: 20060505 08:00:00 {time zone}
FA Group (Financial Advisors only)		FINANCIAL ADVISORS ONLY - The name of the account group to use for an order.
FA Method (Financial Advisors only)		FINANCIAL ADVISORS ONLY - The name of the default share allocation method for the account group (EqualQuantity, NetLiq, AvailableEquity, or PctChange).
FA Percentage (Financial Advisors only)		FINANCIAL ADVISORS ONLY - The percentage used by the PctChange allocation method for an order. Appies only to FA groups that use this method.
FA Profile (Financial Advisors only)		FINANCIAL ADVISORS ONLY - The name of the allocation profile to use for an order.
Short Sale Slot (institutional only)		1 if you hold the shares, 2 if they will be delivered from elsewhere. Only for Action="SSHORT"
Short Sale Location (institutional only)		If shares will be delivered from elsewhere, specify where (can be comma-delimitted list, no spaces)
OCA Type		1=Cancel on Fill with Block, 2=Reduce on Fill with Block, 3=Reduce on Fill without Block
Rule 80A		Individual = 'I', Agency = 'A', AgentOtherMember = 'W', IndividualPTIA = 'J', AgencyPTIA = 'U', AgentOtherMemberPTIA = 'M', IndividualPT = 'K', AgencyPT = 'Y', AgentOtherMemberF
Settling Firm (Institutional only)		For Institutional only.
All Or None (0=false, 1=true)		Identifies an order as an All or None order.
Minimum Qty		Identifies an order as a Minimum Quantity order.
Percent Offset		Relative orders only
Electronic Trade Only		(0=false, 1=true) (SMART Routing)
Firm Quote Only		(0=false, 1=true) (SMART Routing)
NBBO Price Cap (SMART Routing)		Maximum SMART order distance from the NBBO.
Auction Strategy		BOX Exchange: 1 = match, 2 = improvement, 3 = transparent
Starting Price		BOX Exchange:
Stock Ref Price		BOX Exchange:
Delta		BOX Exchange:
Underlying Range (Low)		Relative to Stock or VOL orders
Underlying Range (High)		Relative to Stock or VOL orders
Volatility		VOL orders: Note that this should be sent as a percentage (ie 17.12 instead of .1712 as the market data is displayed).
Volatility Type (1=daily, 2=annual)		VOL orders
Reference Price Type (1=average, 2 = BidOrAsk)		VOL orders
Hedge Delta Order Type		VOL orders: NONE if no delta hedging is desired. Typical possibilities include MKT, LMT, REL. Send 1 for MKT or 0 for NONE to TWS prior to Version 859.
Continuous Update (0=false, 1= true)		VOL orders
Hedge Delta Aux Price		VOL orders: specify only if delta hedge order type needs it
Trail Stop Price		For sell orders, Trail Stop Price = Limit Price – Trailing Amount - Limit Offset. For buy orders, Trail Stop Price = Limit Price + Trailing Amount + Limit Offset. For trailing stop limit (TRA
Scale Num Components		THIS ATTRIBUTE IS NO LONGER SUPPORTED.
Scale Component Size		SCALE orders
Scale Price Increment		SCALE orders
Outside RTH (0=false, 1=true)		Allows an order to be triggered or filled outside of regular trading hours.

As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

The Extended Order Attributes page is a template. This means that any values you enter in the fields on this page will apply to all orders you transmit UNLESS you specify a different attribute value on that order line. Remember the last chapter, where you took a look at the second black section of fields labeled *Extended Order Attributes*? That section of the Basic Orders page lets you set attributes at the order level, versus the Extended Order Attributes page, which applies attributes at a global level, affecting all orders.

There is one small catch to all of this template business, and that is the time in force. The time in force field dictates which attributes will be applied to your order. If you leave the TIF value blank on your order line all of the values that currently reside in the Extended Order Attributes template will be applied to your order. If however you ENTER a time in force value, the EOAs (you know, Extended Order Attributes) on the EOA page will no longer be used for that order, which leaves every other EOA field blank. So, if you're thinking to yourself, "Self, I think I'll enter a different time in force for this order, but use all of the other attributes from the EOA page." well, let's just say you'd be kidding yourself!

An Example

Still don't understand? Let's look at something basic that will resonate with many traders, the time in force value of DAY. A day order is familiar to us all, it's simply an order that will be cancelled at the close of the day's market if it hasn't executed. If you notice on this Extended Order Attributes page, the very first value at the top of the page is DAY, in the time in force row. Now, if you create and transmit an order and don't enter any value in the Time in Force field that appears on your order line, then your order by default is a DAY order, since it uses whatever value is in this Extended Order Attributes template.

Now go back to that order on the Basic Orders page, but before you transmit it, scroll over to the right and enter GTC (good till cancelled) in the *Time in Force* field. Now, for JUST THAT ORDER, you have changed the time in force to good-till-cancelled, and the order will continue to work, day after day, until it either executes or gets cancelled. On the other hand, if you were to have entered GTC as the time in force on the Extended Order Attributes page, all of your orders from that day forward would have the GTC time in force, 'till death do they part. Remember that the template applies for all of the fields on this page, not just the time in force.

Extended Order Attributes

As you can see in Appendix A, the API supports many optional order attributes.

We recommend you check the API Release Notes and API Reference Guide at http://individuals.interactivebrokers.com/en/p.php?f=programInterface&p=a&ib_entity=lic to keep abreast of new attributes as they're added.

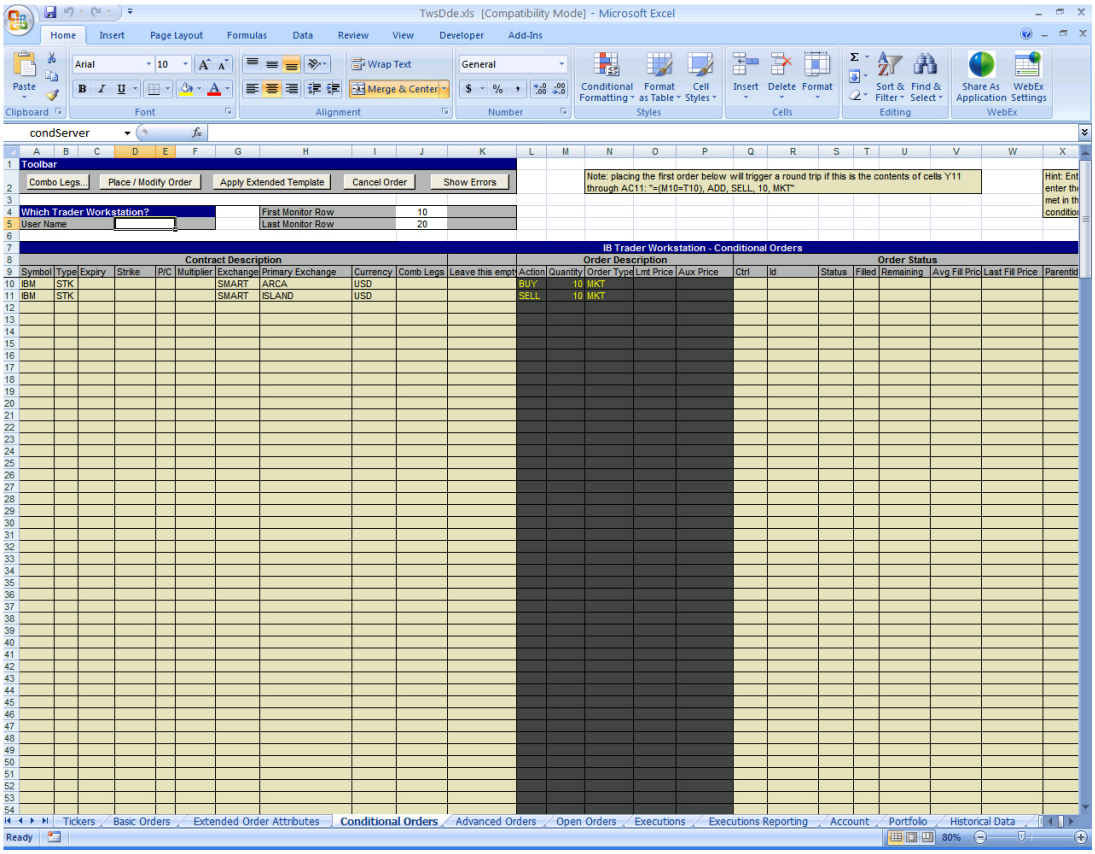
You can see that the attributes are labeled if they are specific to institutional or advisor accounts. You enter the value you want to use in the white Value column, and that value will be used for all orders unless you specify an order-level value (on the Basic Orders page) that is different.

Now you've seen the Basic Orders page, where you create your basic orders, and the lovely Extended Orders Attributes page, where you apply such characteristics as a new time in force, a display size to create an iceberg order, a discretionary amount to give more leeway to your limit orders, and even a checkbox to hide your entire order. You can do a lot of cool things with these two pages, but the one thing you can't do is set up conditional orders, which are described in the next chapter.

We've provided a second orders page, the Advanced Orders page, where you can place orders that require use of extended order attributes, but you won't learn about that page until Chapter 11!

Chapter 10: The Conditional Orders Page

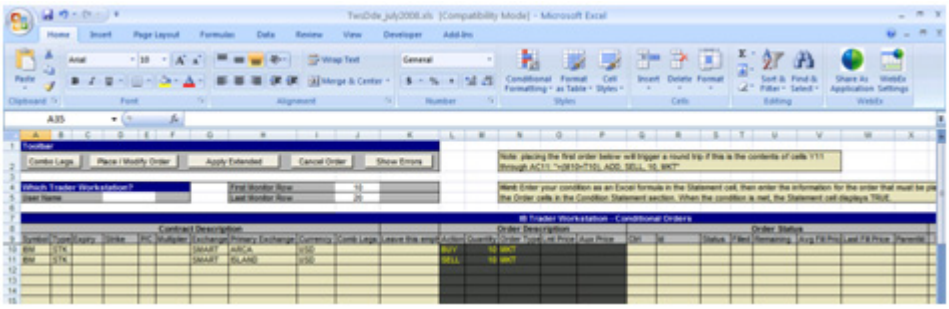
A conditional order is one that relies on something else happening, either a price changing or another order executing, before this order is submitted. Let's take a look at this useful and outstanding order.



As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

In a nutshell, a conditional order is one that waits around, all set to go, for something precipitous to happen that will trigger its coming to life. For example, you might want to submit a sell order for 100 shares of XYZ, but you only want to do it if you're first able to buy 100 shares of LMNOP at some specific price. With the conditional order all set waiting in the wings, you don't need to sit around watching and waiting to see if the first order executes. All you need to do is set up a trigger condition and let it go.

Take a look at the paired set of sample orders that are included on the Conditional Orders page in the sample spreadsheet.



The key field in a conditional order is field Y11, the *Statement* trigger field. Until some condition is met, the value in this field is always FALSE. This means nothing will happen. When your condition is met, this value becomes TRUE and that triggers your conditional order.

In the above illustration, note the value "FALSE" which appears in cell Y11, the trigger Statement field. When you put your curser in this cell, the cell's condition trigger displays on the formula line. In this case the condition says "=(M10=T10)," which in English means "the value in this field will be TRUE when the value in cell M10 is equal to the value in cell T10." Let's look closer and see what M10 and T10 really are. You see that cell **M10** is the *Quantity* field for the first order, and has a value of 10. Cell **T10** is the *Filled* field for that order and currently has no value. Since the filled field shows the filled order quantity, this means that the second order will only trigger when the full quantity of the first order has executed.

To see this order in action, highlight the order on row 10 and click the **Place/Modify Order** button. Since it's a market order it should execute quickly, so watch closely!

Set Up a Conditional Order

Now you'll set up a conditional order from scratch. Instead of submitting an order based on another order's execution, let's set one up that is based on a price change. If you want this to execute you'll have to do a little bit of research to find a price that is valid now, but let's walk through a hypothetical example where we have the API submit an order if a price drops by a certain amount. Let's say you want a conditional order that says "If the bid price for IBM[1] drops below $81.00, submit a buy limit order for 500 shares with a limit price of $80.90."

For this order, you'll need to monitor the value on another page, specifically the *Bid Price* on the Tickers page since you want to know when the bid price drops below 81.00.

1. To get this value, go to the Tickers page and put your cursor in the *Bid Price* field on the line that's displaying IBM[1] market data (if you don't have it, add it!) Now look up in the formula line (it's right under the toolbar at the top of the page, and has the equal sign to identify it) and see that it says "=username|tik!id4?bid" but with your username instead of the word "username." Copy this value using the *Copy* command or **Ctrl+C**, then scoot right back to the Conditional Orders page.

2. Now, let's set this up. Start in a blank line, and enter the underlying and other parameters in the *Contract Description* fields (IBM[1], STK, Smart and USD).

3 Scroll over and find the *Condition Statements* area just to the right of the *Order Status* section. Put your cursor in the *Statement* field, then paste the value you copied from the Tickers page into the Formula bar. To see what you've made so far, move your cursor into some other field and look at the value in the Statement field - it shows you the current bid price for IBM1! OK, now go back to the Formula bar for that statement and add this to the end: <81.00. This simply says "if the bid price is less than 81.00."

You might be wondering why you didn't just use the formula "O 12 > 81.00" or something like that, since you can pinpoint the column and row in which the participating bid price resides. The reason is that you're using two different pages, both of which have an "O 12" field. The Conditional Orders page would use the O 12 value from its own page, and that would not help you one bit. So in this case, to avoid confusion you have to use the actual field code.

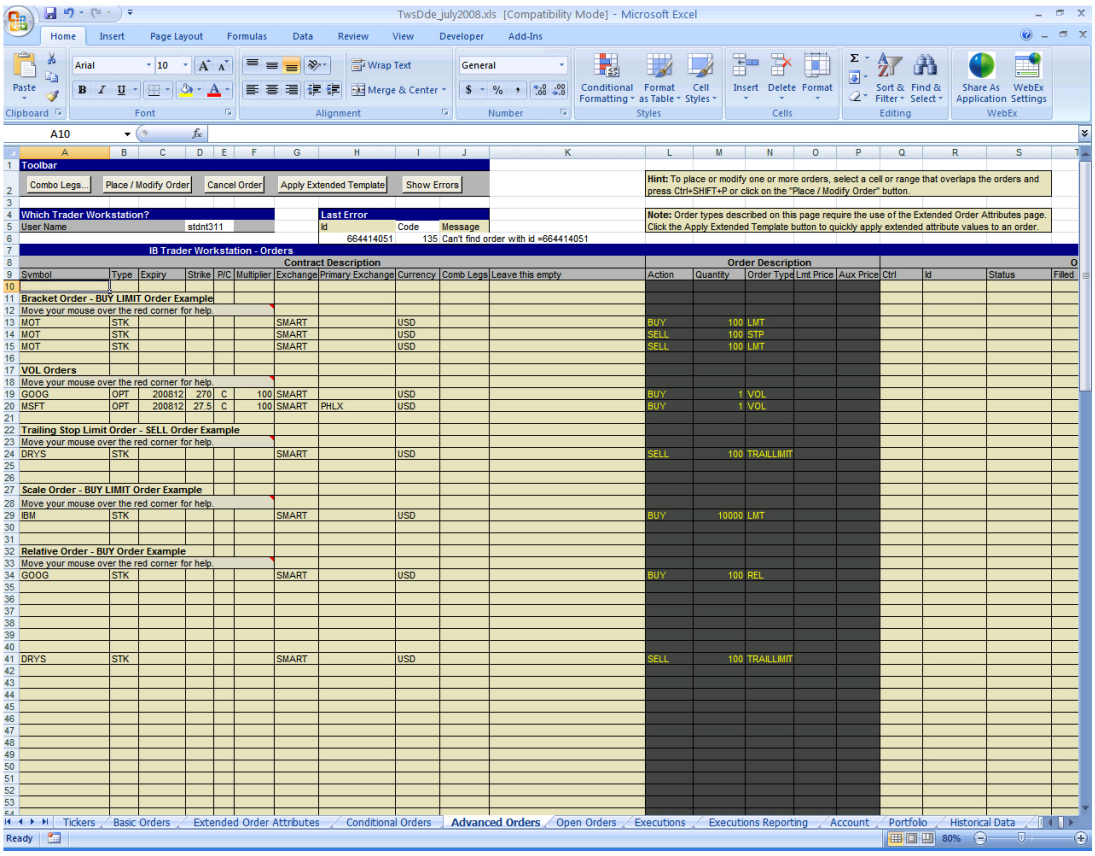
4 Again, move your cursor to some other field, and now you'll see that the value in the *Statement* field is "FALSE," which is just what you want! If the IBM[1] bid price drops below 81.00, this will become TRUE, and when the Statement value is true - it triggers the conditional order!

5 Next, you need to tell the API what you want it to do if the bid price drops below 81.00. So, in the *ADD/MOD* field, enter **ADD** (since you're not modifying an order), then **BUY** in the *Action* field, **500** in the *Quantity* field, **LMT** in the *Order Type* field and **80.90** in the *Lmt Price* field.

OK, you've done it. Will this execute for you? Who knows? But if your conditions are ever met, your new order will be submitted, that you can count on! You can do lots of interesting things with the Conditional Orders feature. We recommend you log into your PaperTrader account and experiment.

Chapter 11 - The Advanced Orders Page

The Advanced Orders page is exactly like the Basic Orders page. We created the this page to make it easy for to you set up and place orders that require the use of extended order attributes. Order types such as bracket, trailing stop limit, volatility and scale orders all require values to be entered on the extended order attributes page before you place the order. You remember extended order attributes from Chapter 9, don't you? Well don't worry, you'll be using them again in this chapter.



You might notice that there are some rows with a little red symbol on the upper right corner of the cell. If you roll your mouse over the red symbol, additional details about each type of advanced order appear in a little popup. Of course, you're always encouraged to look at our TWS Users' Guide or the DDE for Excel section of the API Reference Guide if you're interested in learning more.

Add Your Tickers to the Advanced Orders Page

To add your tickers from the Tickers page to the Orders page, simply copy and paste your ticker lines from one page to another. You might remember this process from Chapter 8, but in case you've forgotten, here it is again:

1 On the Tickers page, select a row by clicking the row number in the far left column, then right click and choose *Copy*. The selected row starts blinking away likes it's on a marquee. For example, copy the IBM ticker line.

2 Now, come back to the Advanced Orders page, select a blank row by clicking in the number field of a blank row, right click and choose *Paste*.

Depending on which version of Excel you are using, you can also choose *Insert Copied Cells* to paste the copied ticker row a blank row; this command automatically moves the other rows down to make room for the new row.

You've just created a new potential order row. You can also enter a new ticker symbol manually just as you did on the Tickers page.

Create a Trailing Stop Limit Order

To demonstrate how to use extended order attributes in setting up advanced order types, we'll show you how to create a trailing stop limit order.

A trailing stop limit order lets you create a trailing stop order that works in conjunction with a dynamically-updating limit order. When the stop order triggers, a limit order is submitted at the last calculated price (instead of a market order which would be submitted with a regular trailing stop order).

The Trailing Stop Limit order uses four components:

- stop price
- trailing amount
- limit price
- limit offset.

In TWS, you enter the stop price, trailing amount and either the limit price or the limit offset; TWS calculates the limit offset or limit price for you, depending on which value you enter. In the DDE for Excel API spreadsheet, you enter the trailing amount, stop price and limit price. There is no field or extended order attribute for the limit offset value. You must include the limit offset in the stop price (the *Trail Stop Price* extended order attribute).

To create a trailing stop limit order:

1 First, enter the ticker information in the *Contract Description* section of the Advanced Orders page. (In our sample spreadsheet, we've included a sample ticker, DRYS[1], and we've even entered the order information in the *Order Description* fields for you! Feel free to use this order row to learn how to place a trailing stop limit order, or enter your own order row.)

2 Click your mouse in the *Action* field on the DRYS[1] ticker line.

Notice that some of the Order Description fields have been filled in for you (Action is SELL, Quantity is 100, and Order Type is TRAILLIMIT.

3 Now you need to enter a limit price in the *Lmt Price* field. Take a look at the DRYS[1] market data and come up with a reasonable limit price. For this example, we've assumed a market price of $71.75[1], so enter a limit price of **71.50**.

4 Enter the trailing amount in the *Aux Price* field. For this example, enter **0.10**.

5 Now go to the Extended Order Attributes page and enter a value for the *Trail Stop Price* attribute. This is the Stop Price for the trailing stop limit order. Type this value in the Trail Stop Price *Value* field. For this example, assume a percent offset of 0.02.

The Trail Stop Price value must include the limit offset.

For a sell order:

Trail Stop Price = Limit Price - Trailing Amount - Limit Offset

For a buy order:

Trail Stop Price = Limit Price + Trailing Amount + Limit Offset

So for this example, which is a sell order, the trail stop price would be 71.38 (or 71.50 - 0.10 - 0.02, using the above sell order equation). Enter **71.38** in the *Value* field for the *Trail Stop Price* attribute (Line 52 on the Extended Order Attributes page).

6 Go back to the Advanced Orders page and highlight the order row, then click the **Apply Extended Template** button. The Trail Stop Price (the stop price for the order) that you entered on the Extended Order Attributes page is applied to your order.

7 With the row still highlighted, click the **Place/Modify Order** button. Watch the *Status* field update with the current order status.

8 Don't forget to delete the *Trail Stop Price* value on the Extended Order Attributes page after you've placed your order. If you don't delete this value, every order you place will have this value associated with it, which you probably don't want!

Now let's place another type of advanced order, a Bracket order.

Create a Bracket Order

Bracket orders are designed to limit your loss and lock in a profit by "bracketing" an order with two opposite-side orders. A BUY order is bracketed by a high-side sell limit order and a low-side sell stop (or stop-limit) order. A SELL order is bracketed by a high-side buy stop (or stop-limit) order and a low side buy limit order.

The Transmit and Parent Order ID Extended Order Attributes

Bracket orders in the DDE for Excel sample spreadsheet require the use of the extended order attributes *Transmit* and *Parent Order Id*.

Transmit is a true/false value (what programmers call a "boolean" value; false is represented by the number 0 and true is represented by the number 1). If *Transmit* is set to true, you type 1 in the *Value* field for the attribute on the Extended Order Attributes page, and every order you place in the sample spreadsheet will be transmitted as soon as you click the **Place/Modify Order** button. It it is set to false, you type 0 in in the *Value* field for the attribute , and every order you place in the spreadsheet will be "on hold" and will not be transmitted to TWS. For your convenience, we've set the *Transmit* value set to true (1 is already entered in the *Value* field on the Extended Order Attributes page).

In the DDE for Excel API sample spreadsheet, the first order in the bracket must be identified as the parent, or primary, order in the bracket. The remaining two orders in the bracket must then be associated with that parent order before you can transmit the bracket order. You do this by using the *Parent Order Id* extended attribute. For now, don't worry about how to do this; we'll show you in the detailed steps below.

Creating the Bracket Order

In the following example, we will guide you through the process of creating a BUY order.

To create a bracket order

1 Click the Advanced Orders tab at the bottom of the spreadsheet.

In our sample spreadsheet, we've included a sample BUY-LIMIT bracket order for shares of Motorola1, and we've even entered the order information in the Order Description fields for all three orders in the bracket for you! Of course, you are always free to enter your own order rows. For these steps, we assume that you are using our sample rows to learn how to place a bracket order.

2 Enter the contract descriptions and order descriptions for all three orders on three contiguous rows. Just as in any other order, enter **BUY** or **SELL** in the *Action* field, the number of shares in the *Quanty* field, and the order type in the *Order Type* field.

In our sample, we've entered the following order rows for you:

- The first order is a BUY LMT order for 100 shares of MOT[1].
- The second order is a SELL STP order for 100 shares of MOT[1].
- The third order is a SELL LMT order for 100 shares of MOT[1].

3 Click the Tickers tab and check the current market price for MOT stock by requesting market data. Our sample spreadsheet includes the MOT ticker for you already.

4 Click the Advanced Orders tab again, then type the limit price for the BUY LMT order row, the stop price for the SELL STP order row, and the limit price for the SELL LMT order row.

For this example, let's assume that the current market price for MOT1 is $9.50:

- In the *Lmt Price* field for the first order (the buy limit order), type **9.00**.
- In the *Aux Price* field for the second order (the sell stop order), type **8.50**.
- In the *Lmt Price* field for the third order (the sell limit order), type **10.00**.

5 Click the **Extended Order Attributes** tab. Change the value for *Transmit* to **0** (you can find this attribute on row 13).

- This ensures that your orders are not transmitted until you have completed the order setup.

6 Click the **Advanced Orders** tab, highlight the first order in the bracket order, then click the **Place/Modify Order** button.

Remember, you've turned off *Transmit* (set it to false), so the order is not immediately executed. However, the system does generate an Order ID, which you can see in the *Id* field in the *Order Status* section of the page.

7 Now it's time to use that *Parent Order Id* extended attribute.

a First, copy the Order ID for the first order, omitting the "id" prefix (click the *Id* field for the first order, select only the numbers of the Id in the Excel Formula Bar, then press **Ctl+c** to copy).

b Click the **Extended Order Attributes** tab and paste the Order ID into the *Value* field for *Parent Order Id*, which is on row 14 (click in the *Value* field for row 14, then press **Ctl+V** to paste the Order Id that you copied in the previous step).

This value will be applied to all subsequent orders until you remove it from the Extended Order Attributes page. And now you've identified the parent or primary order of the bracket!

8 Click the **Advanced Orders** tab, highlight the second order, then click the **Place/Modify Order** button.

Don't worry, the order is not executed yet because that *Transmit* extended order attribute is still set to **False**. But this second order is now associated with the parent, or primary, order in the bracket by means of the *Parent Order Id* extended order attribute.

9 You're almost ready to place the entire bracket order. Click the **Extended Order Attributes** tab again and change the value for *Transmit* back to **1** (row 13). The next order you place WILL be instantly transmitted.

10 Click the **Advanced Orders** tab again, highlight the third order in the bracket order, then click the **Place/Modify Order** button.

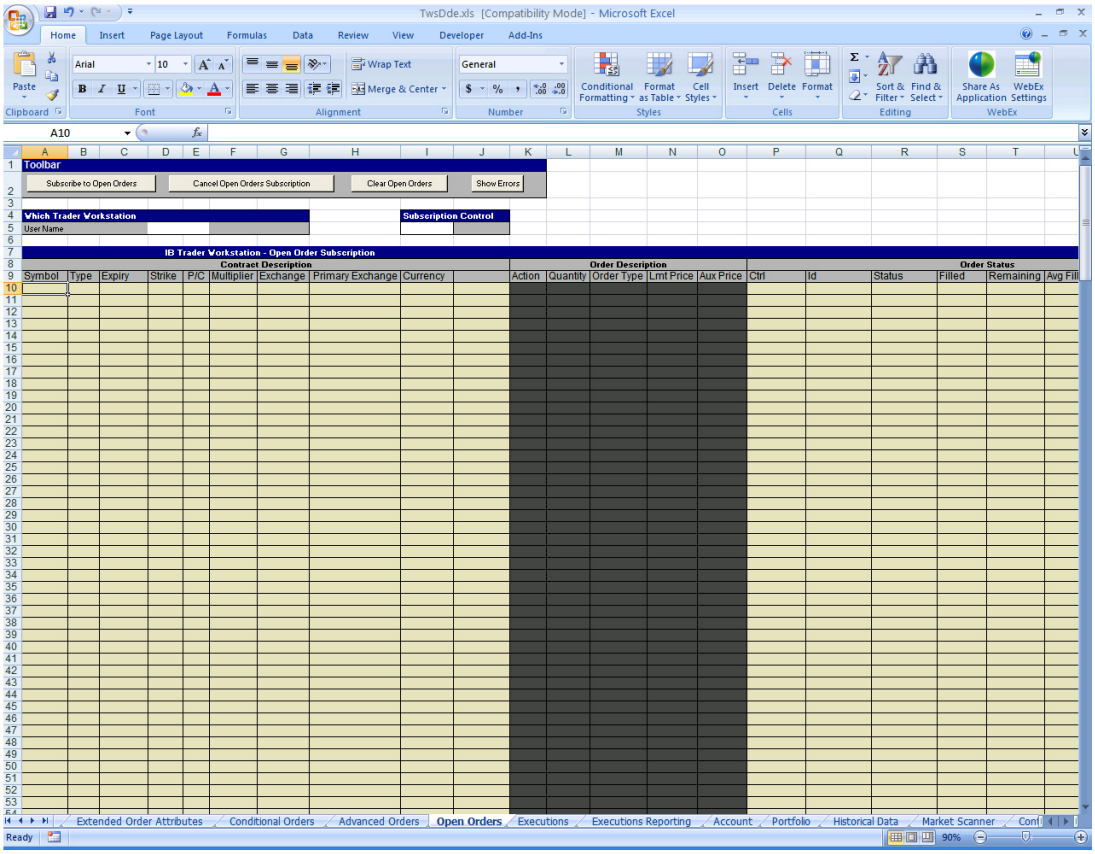
The entire bracket order is finally transmitted and you'll notice that all three orders now have a status displayed in the *Order Status* field. Also take a look at the *Parent Id* field, which is the right-most column in the *Order Status* section of the page. Notice that the second and third order have the same Parent Id, and that Parent Id is the same as the Id of the first order! This is proof that the second and third orders in the bracket order have been correctly associated with the first order.

11 When you are done placing your bracket order, go to the Extended Order Attributes page and delete the *Parent Order Id* value you entered. If you do not, this value will be applied to all subsequent orders that you place in the spreadsheet.

There are other advanced order samples on the Advanced Orders page, including sample VOL orders, scale and relative orders. All of these order types require the use of extended order attributes. You can read more about how to set up these orders in the API Reference Guide, which is available from http://individuals.interactivebrokers.com/en/p.php?f=programInterface&p=a&ib_entity=lic on our web site.

Chapter 12 - The Open Orders Page

The Open Orders page shows you all transmitted orders, including those that have been accepted by the IB system, and those that are working at an exchange.. Once you have subscribed, the page is updated each time you submit a new order, either through the API or in TWS.



As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

How Do I Subscribe?

Another way to say this would be "View your open orders." You don't subscribe in the everyday sense of signing up, paying a fee and receiving something. You simply click a button, and the VBA macros working behind the scenes quickly do all the complicated stuff, so that you can see all of your open orders. In addition, these macros ensure that this data updates in real-time, so that if an order executes you'll know it.

To do this, imply click the **Subscribe to Open Orders** button on the toolbar. All of your orders will display in the Excel worksheet as soon as they are accepted. If you don't currently have any working orders, the API will kindly tell you that you have "None at this time."

Even when you log out of TWS or close down your Excel spreadsheet, your subscription remains active until you elect to cancel it by clicking the **Cancel Open Orders Subscription** button. If you entered new orders via TWS while the API was shut down, as soon as you log back in any new open orders are displayed.

Clearing Open Orders

To clear the spreadsheet of all displayed open orders, first cancel the subscription, and then click the **Clear Open Orders** button.

Named Ranges for Results

The results are displayed in a "named range" so that they can easily be manipulated through a page-specific *Worksheet_Calculate* macro, which our API developers have written and included. A named range is a Microsoft Excel feature that lets you assign a meaningful name to a single cell or a range of cells. In the DDE for Excel API VB code, we use named ranges on every page to represent ranges of cells on that page. For example, on the Open Orders page, we assigned the named range "openSubOrders" to all of the cells in the *Order Description* section of the Open Order results to easily reference those cells in the code.

In Microsoft Excel 2007, you access the named range feature on the Formulas tab (for example, the Name Manager displays all names defined for the current worksheet). In previous versions of Microsoft Excel, you define name ranges from the Insert menu.

Using named ranges, you can tell the API to put the open orders somewhere else, like on a different spreadsheet or in a different set of rows and columns.

View the VB Code

To view or use the code in this macro:

1. Use the Excel Tools menu, mouse over the *Macro* subcommand and then choose *Macros*.

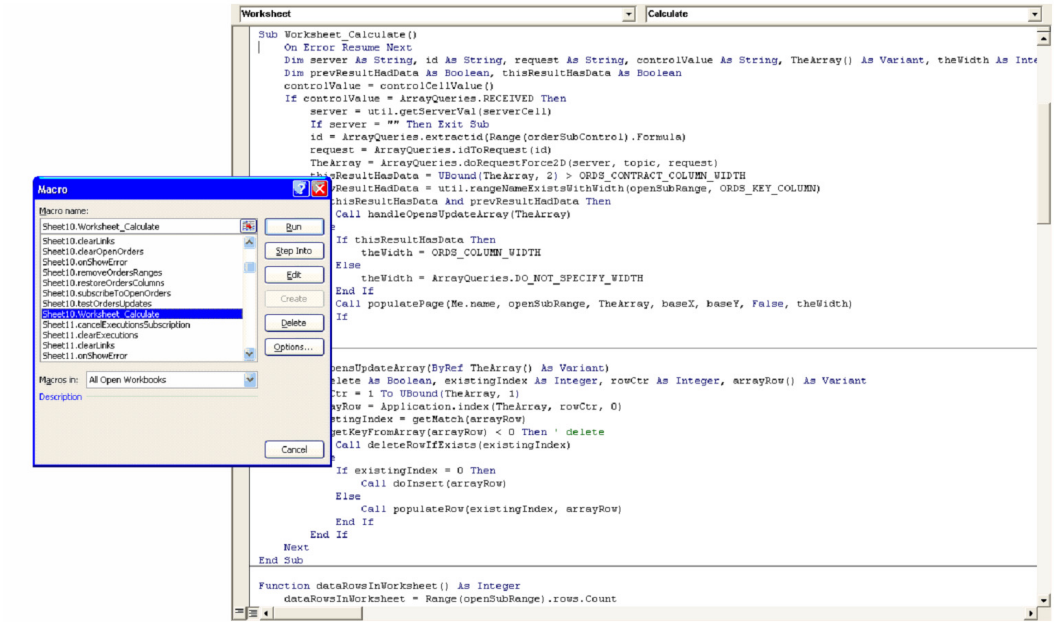
   If you are using Excel 2007, select the **View** tab at the top of the screen, then click the **Macros** button.

2. Scroll down the code window to find the *Worksheet_Calculate* macro specific to the page functionality you're interested in. For example, Sheet10 macros apply to the Open

Orders page, which you can infer by the Sheet10 macro names, including *subscribeToOpenOrders* and *cancelOpenOrderSubscription*.

3 To see the named ranges used on this page, open the code for Sheet 10 and look at the code for the macros used on this page.

In Excel 2007, you can view all of the named ranges used in the entire API spreadsheet by clicking the *Formulas* tab at the top of the window, then clicking **Name Manager**. In previous versions of Excel, you can view and define named ranges by selecting *Name* from the **Insert** menu, then selecting *Define*. There is also a Name Manager for Excel utility available from Microsoft for versions of Excel prior to Excel 2007.

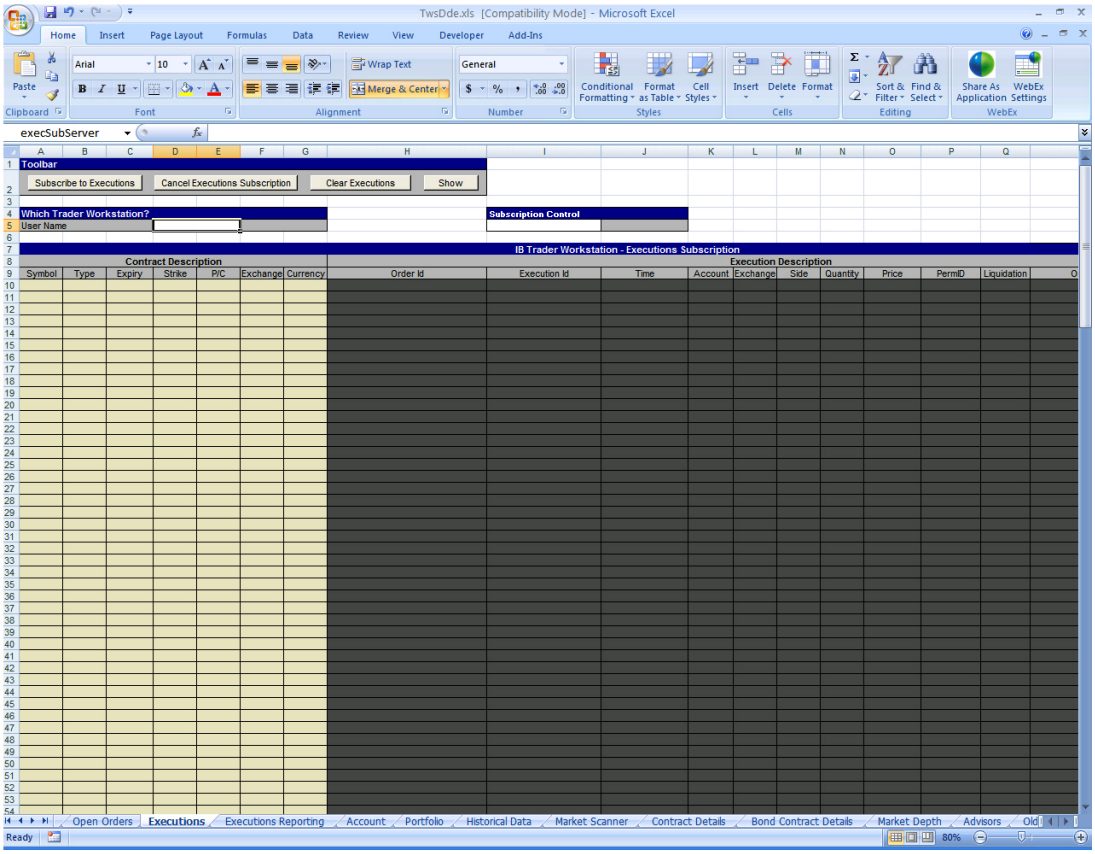


What Happens If an Open Order Executes?

Once an order executes, it remains on the Open Orders page for 30 seconds, with the *Status* value changed to *FILLED*. Then the filled order is cleared away, and you can see it on the Executions page (once you subscribe to real-time executions, of course!).

Chapter 13 - The Executions Page

The Executions page is pretty straightforward - it displays the current day's executions.



As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

How Do I Subscribe to Executions?

Another way to say this would be "View your executions." You don't actually "subscribe" to anything in the usual sense of signing up, paying a fee and receiving something. You simply click the **Subscribe to Executions** button, and the VBA macros working in the background quickly do all the complicated stuff and magically populate the screen with your current day's executions. Your "subscription" remains active until you elect to cancel it by clicking the **Cancel Executions Subscription** button. And, any orders that execute while TWS is shut down or the API is logged off will display as soon as you log back and go to the Executions page.

OK, to do all this stuff, simply click the **Subscribe to Executions** button on the toolbar. All of the day's executions will display in the Excel worksheet. If you currently do not have any execution reports, the API will kindly tell you that you have "None at this time."

Clearing Executions

To clear the spreadsheet of all displayed executions, first cancel the subscription, and then click the **Clear Executions** button.

Named Ranges for Results

The results are displayed in a "named range" so that they can easily be manipulated through a page-specific *Worksheet_Calculate* macro, which our API developers have written and included. A named range is a Microsoft Excel feature that lets you assign a meaningful name to a single cell or a range of cells. In the DDE for Excel API VB code, we use named ranges on every page to represent ranges of cells on that page. For example, on the Executions page, we assigned the named range "execSubDetails" to all of the cells in the *Execution Description* portion of the Executions page results to easily reference those cells in the code.

In Microsoft Excel 2007, you access the named range feature on the Formulas tab (for example, the Name Manager displays all names defined for the current worksheet). In previous versions of Microsoft Excel, you define name ranges from the Insert menu.

Using named ranges, you can tell the API to put this execution data somewhere else, like on a different spreadsheet or in a different set of rows and columns.

View the VB Code

To view or use the code in this macro:

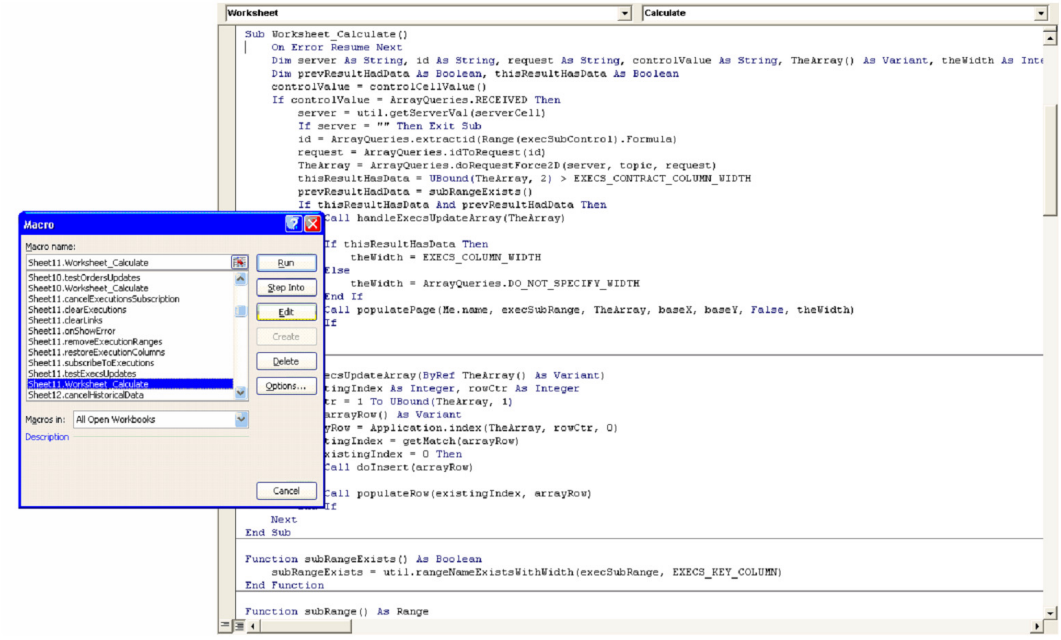
1. In the Excel **Tools** menu, mouse over the *Macro* subcommand and then choose *Macros*.

   If you are using Excel 2007, select the **View** tab at the top of the screen, then click the **Macros** button.

2. Scroll down the code window to find the *Worksheet_Calculate* macro specific to the page functionality you're interested in. For example, Sheet11 macros apply to the Executions page, which you can infer by the Sheet11 macro names, including *subscribeToExecutions* and *clearExecutions*.

3 To see the named ranges used on this page, open the code for Sheet 11 and look at the code for the macros used on this page.

In Excel 2007, you can view all of the named ranges used in the entire API spreadsheet by clicking the Formulas tab, then clicking Name Manager. In previous versions of Excel, you can view and define named ranges by selecting Name from the Insert menu, then selecting Define. There is also a Name Manager for Excel utility available from Microsoft for versions of Excel prior to Excel 2007.



The Executions Reporting Page

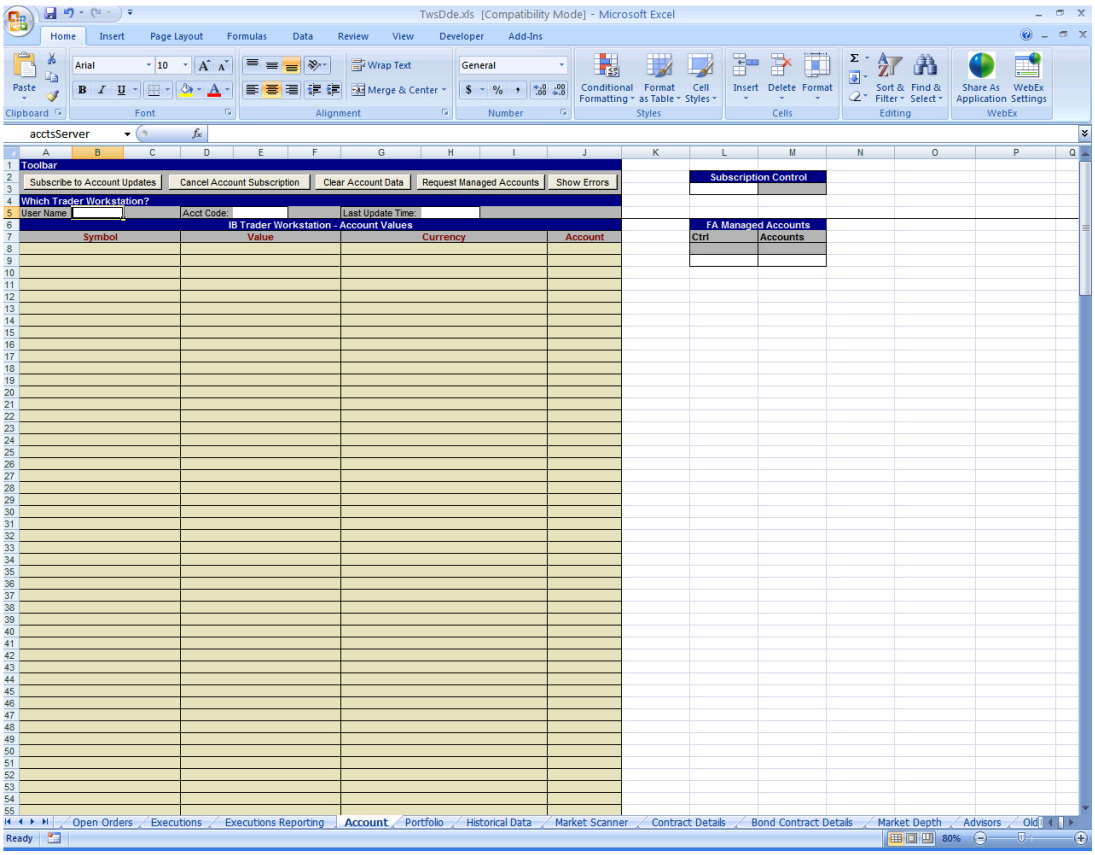
In addition to viewing all executions, you can filter your returns using the Executions Reporting page. This page is linked to the Executions page (which means you don't need to enter your username anywhere) and can be used as long as you have subscribed to the executions function on the Executions page.

You can select one of the following report types on the Executions Reporting page:

- **Order ID** - finds all executions resulting from orders with a specified PermID.
- **Order Ref** - finds all executions resulting from orders with a given order reference; for example executions from a specific basket order.
- **VOL order** - finds all executions resulting from specific volatility order, including any hedge delta executions.
- **Strategy** - in the Key field, enter a value to define the Type you selected. For example, if you selected Order ID as the type, enter a specific order ID in the *Key* field.

Chapter 14 - The Account Page

The Account page shows you everything you will ever need to know about the status of your account. Once you've entered your TWS Username, simply click the Subscribe to Account Updates button and the Account Values section will be filled with up-to-the-minute values, from your Equity with Loan Value through your margin and P&L values. (see Appendix B for details on all of the values shown on the Account page). These values are updated every 3 minutes.



As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

How Do I Subscribe to Account Updates?

As with the Open Orders and Executions pages, it might have made more sense to describe this as "View your account information" rather than "Subscribe to Account Updates." You don't actually "subscribe" to anything in the usual sense of signing up, paying a fee and receiving something. You simply click a button, and the VBA macros working in the background quickly do all the complicated stuff and magically populate the screen with your up-to-date account information. Your "subscription" remains active until you elect to cancel it by clicking the **Cancel Account Subscription** button.

Note the little field between the buttons and the field titled *Last Update Time*. This time tells you when the API last updated the values on the spreadsheet through TWS. These values are checked and updated automatically every three minutes.

To make this work, simply click the **Subscribe to Account Updates** button on the toolbar. Your latest account information instantly populates the page.

Clearing Account Values

To clear the spreadsheet of all the account values, first cancel the subscription, and then click the **Clear Account Data** button.

Named Ranges for Results

The results are displayed in a "named range" so that they can easily be manipulated through a page-specific *Worksheet_Calculate* macro, (for the Account and Portfolio pages, the macros are for Sheet.14 and Sheet.15 respectively) which our API developers have written and included. A named range is a Microsoft Excel feature that lets you assign a meaningful name to a single cell or a range of cells. In the DDE for Excel API VB code, we use named ranges on every page to represent ranges of cells on that page. For example, on the Account page, we assigned the named range "acctsDataRange1" to all of the cells in the *Symbol* column to easily reference those cells in the code.

In Microsoft Excel 2007, you access the named range feature on the Formulas tab (for example, the Name Manager displays all names defined for the current worksheet). In previous versions of Microsoft Excel, you define name ranges from the Insert menu.

Using named ranges, you can tell the API to put this execution data somewhere else, like on a different spreadsheet or in a different set of rows and columns.

View the VB Code

To view or use the code in this macro:

1. In the Excel **Tools** menu, mouse over the *Macro* subcommand and then choose *Macros*.

   If you are using Excel 2007, select the **View** tab at the top of the screen, then click the **Macros** button.

2. Scroll down the code window to find the *Worksheet_Calculate* macro specific to the page functionality you're interested in. For example, Sheet14 macros apply to the Account

page, which you can infer by the Sheet14 macro names, such as *subscribeToAccts* and *requestManagedAccts*

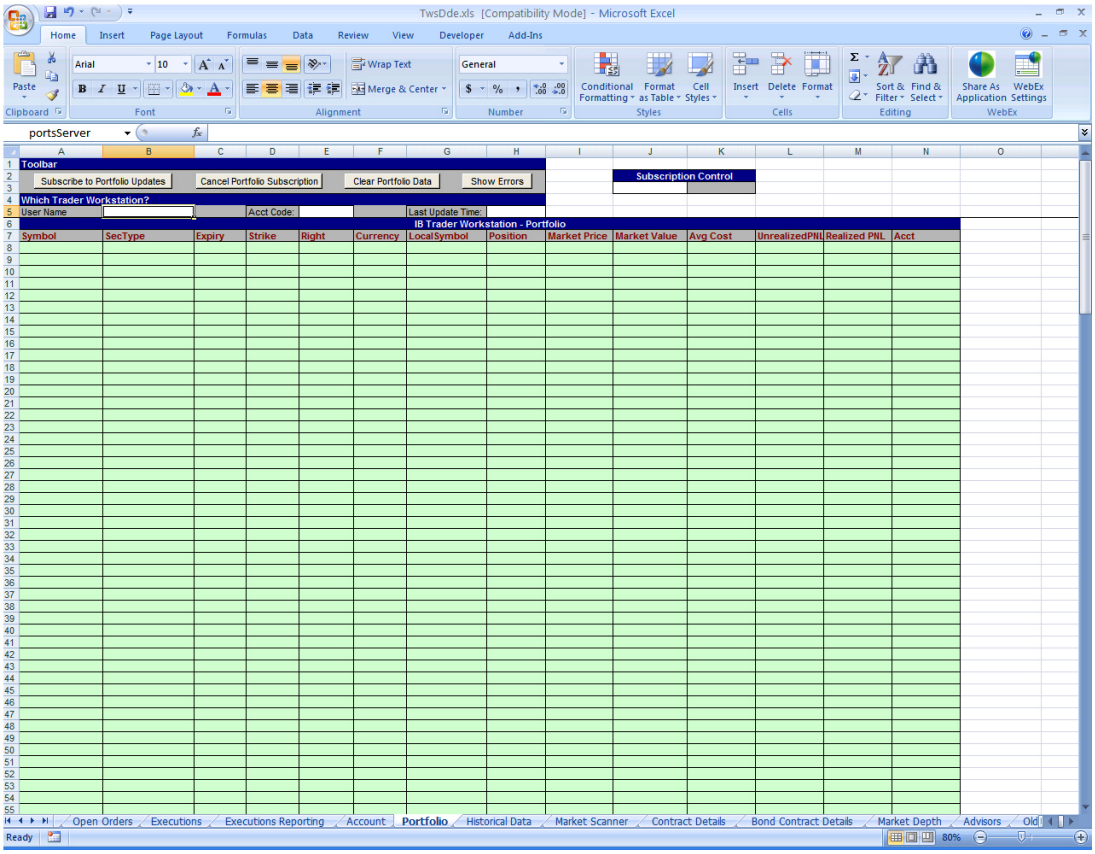
3 To see the named ranges used on this page, open the code for Sheet 11 and look at the code for the macros used on this page.

If you are using Excel 2007, select the View tab at the top of the screen, then click Macros button.

In Excel 2007, you can view all of the named ranges used in the entire API spreadsheet by clicking the Formulas tab, then clicking Name Manager. In previous versions of Excel, you can view and define named ranges by selecting Name from the Insert menu, then selecting Define. There is also a Name Manager for Excel utility available from Microsoft for versions of Excel prior to Excel 2007.

Chapter 15 - The Portfolio Page

The Portfolio page displays all of your current positions.

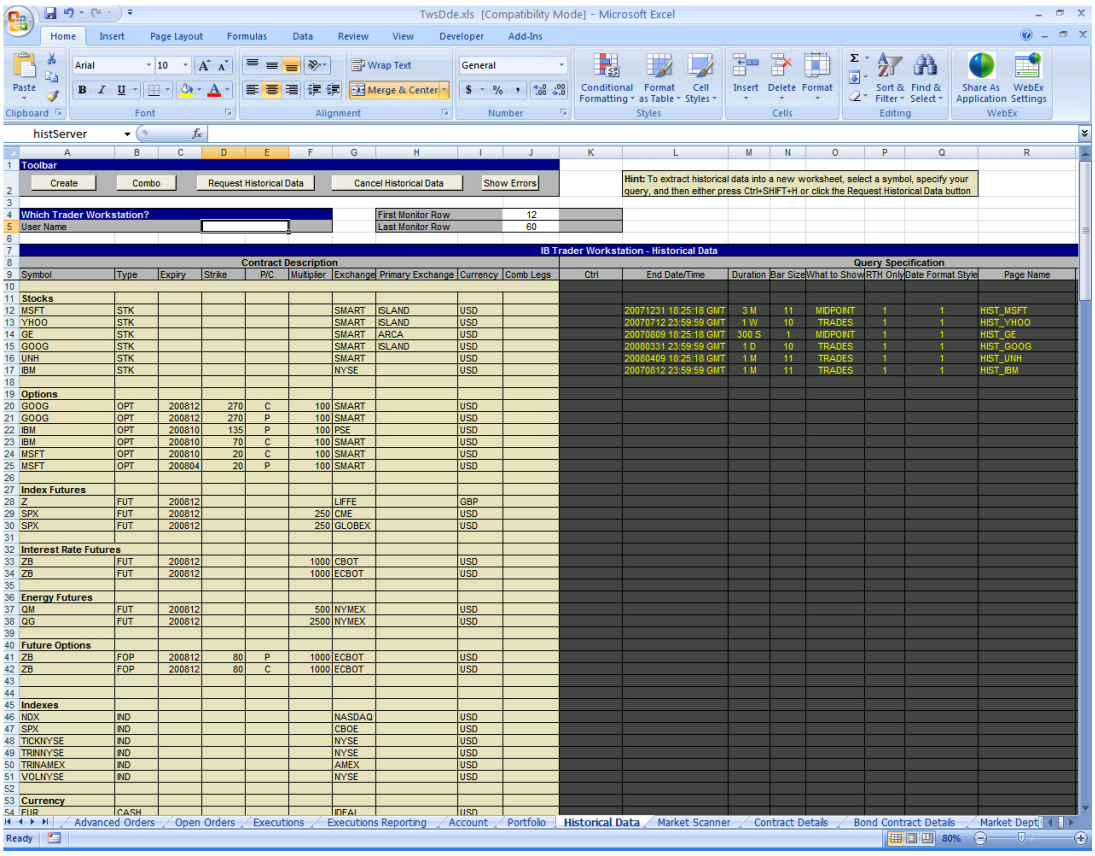


As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

To view your portfolio, simply click the **Subscribe to Portfolio Updates** button to populate the page with the current positions you hold in your portfolio. This page works the same as the Account page, and also goes out to TWS and updates the values every three minutes, which you can see in the *Last Update Time* field in the *Which Trader Workstation?* area.

Chapter 16 - The Historical Data Page

Use the Historical Data tab to request historical data for an instrument based on data you enter in a query. This information equates to charting a contract in TWS, except that the information you see is all in rows on a spreadsheet, rather than in a pretty colored line or bar chart! As a matter of fact, each row value in the spreadsheet is considered a "bar" just a like a bar in your chart. You can request historical data as far back as a year, and also for expired futures contracts.



As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

Excel shows the results on a separate page, which it creates specifically for these results. When you define the parameters, you'll include a name for your historical data returns in the *Page Name* field.

Note that since the query returns a named range of cells, you can write VBA macros to perform computations on it, and you can chart and sort the data in Excel.

How Do I Request Historical Data?

Requesting historical data is as easy as pushing a button! Well, first you need to make sure you're hooked up to TWS through your user name, and then you have to figure out which symbol you want to request data for. After you've done all THAT, well THEN it's just as easy as pushing a button!

The Historical Data page in the sample application comes pre-populated with all types of instruments as examples for you to follow when creating a ticker and then requesting historical data. To create a ticker line, it's the same simple process you used to add tickers on the Tickers page. You can either click the **Create Ticker** button and enter the description in the Ticker dialog box, or you can click your mouse in an empty *Symbol* field on the worksheet page and tab through the fields necessary to define your ticker.

Historical Data Parameters

Now that you have the ticker on the page, you need to define the parameters the application will use to find your data. The following table describes the fields that appear in the *Query Specification* section of the Historical Data page.

Parameter	Description
End Data/Time	Use the format yyyymmdd {space}hh:mm:ss{space}*tmz* where the time zone is allowed (optionally) after a space at the end.
Duration	This is the time span the request will cover, and is specified using the format *integer* {space} *unit*, where valid units are: • S (seconds) • D (days) • W (weeks) • M (months) • Y (years) This unit is currently limited to one. If no unit is specified, seconds are used.

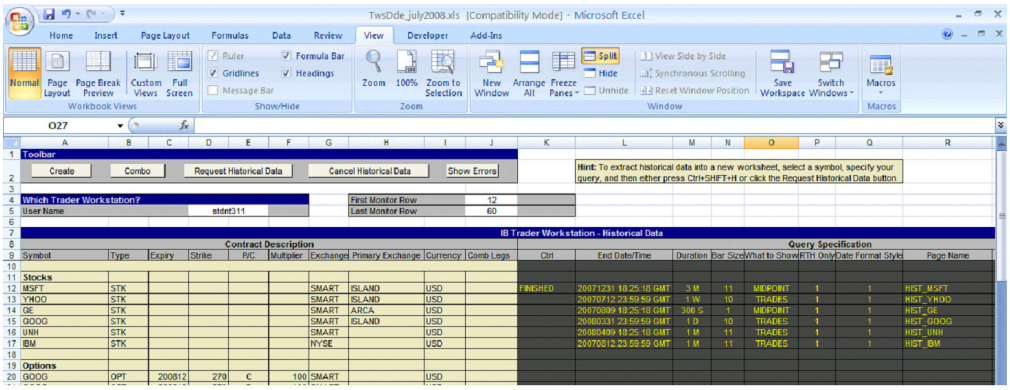
<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Bar Size</td><td>Specifies the size of the bars that will be returned. The following bar sizes may be used, and are specified using the parametric value:

Bar Size String — Integer Value
1 second — 1
5 seconds — 2
15 seconds — 3
30 seconds — 4
1 minutes — 5
2 minutes — 6
3 minutes — 16
5 minutes — 7
15 minutes — 8
30 minutes — 9
1 hour — 10
1 day — 11
1 week — 12
1 month — 13
3 months — 14
1 year — 15

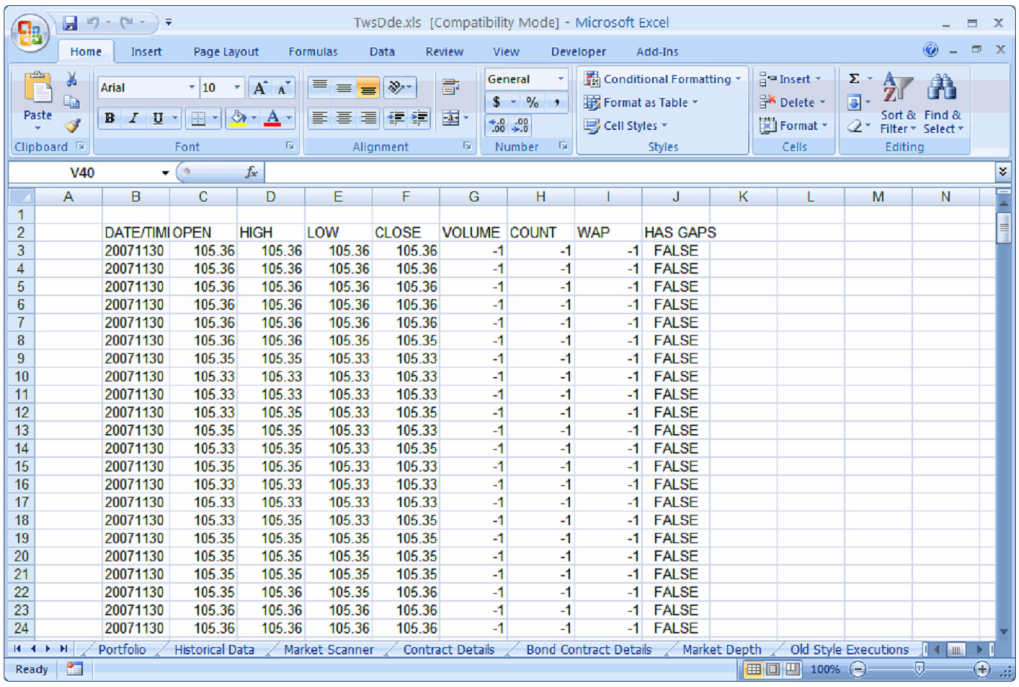
On the query return page, each "bar" is represented by a line in the spreadsheet. If you specify a duration of 300 seconds, and a bar size of "1" (one second) your return will include 300 lines, and the value in each line is equal to one second, or is a one-second bar. Note that you can use either the Integer value of the Bar Size String or the Integer Value to define the bar sizes.</td></tr>
<tr><td>What to Show</td><td>Determines the nature of the data extracted. Valid values include:
Trades
Midpoint
Bid
Ask
Bid_Ask
All but the Bid_Ask data contain the start time, open, high, low, close, volume, trade count and weighted average price during the time slice queried. For the Bid/Ask query, the open and close values are the time-weighted average bid and the time-weighted average offer, respectively. These bars are identical to the TWS charts' candlestick bars.</td></tr>
</table>

Parameter	Description
RTH Only	Regular Trading Hours only. Valid values include: • 0 - all data available during the time span requested is returned, including time intervals when the market in question was outside of regular trading hours. • 1 - only data within the regular trading hours for the product requested is returned, even if the time span falls partially or completely outside.
Date Format Style	Valid values include: • 1 - dates that apply to bars are returned in the format yyyymmdd{space}{space}hh:mm:dd (the same format used when reporting executions). • 2 - the dates are returned as an integer specifying the number of seconds since 1/1/1970 GMT.
Page Name	The name you enter will be used to name the new system-created page into which Excel puts the historical data.
Expired	Valid values include True and False. If true, the query will run on an expired contract, although it is limited to the last year of the contract's life, and is currently only valid for expired futures contracts.

OK, so you've created a ticker and then entered all the query information you'll need. Your spreadsheet page looks something like this:



Now just click the **Request Historical Data** button and after a few moments, the API will create a new page that looks something like this:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1														
2		DATE/TIMI	OPEN	HIGH	LOW	CLOSE	VOLUME	COUNT	WAP	HAS GAPS				
3		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				
4		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				
5		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				
6		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				
7		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				
8		20071130	105.36	105.36	105.35	105.35	-1	-1	-1	FALSE				
9		20071130	105.35	105.35	105.33	105.33	-1	-1	-1	FALSE				
10		20071130	105.33	105.33	105.33	105.33	-1	-1	-1	FALSE				
11		20071130	105.33	105.33	105.33	105.33	-1	-1	-1	FALSE				
12		20071130	105.33	105.35	105.33	105.35	-1	-1	-1	FALSE				
13		20071130	105.35	105.35	105.33	105.33	-1	-1	-1	FALSE				
14		20071130	105.33	105.35	105.33	105.35	-1	-1	-1	FALSE				
15		20071130	105.35	105.35	105.33	105.33	-1	-1	-1	FALSE				
16		20071130	105.33	105.33	105.33	105.33	-1	-1	-1	FALSE				
17		20071130	105.33	105.33	105.33	105.33	-1	-1	-1	FALSE				
18		20071130	105.33	105.35	105.33	105.35	-1	-1	-1	FALSE				
19		20071130	105.35	105.35	105.35	105.35	-1	-1	-1	FALSE				
20		20071130	105.35	105.35	105.35	105.35	-1	-1	-1	FALSE				
21		20071130	105.35	105.35	105.35	105.35	-1	-1	-1	FALSE				
22		20071130	105.35	105.36	105.35	105.36	-1	-1	-1	FALSE				
23		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				
24		20071130	105.36	105.36	105.36	105.36	-1	-1	-1	FALSE				

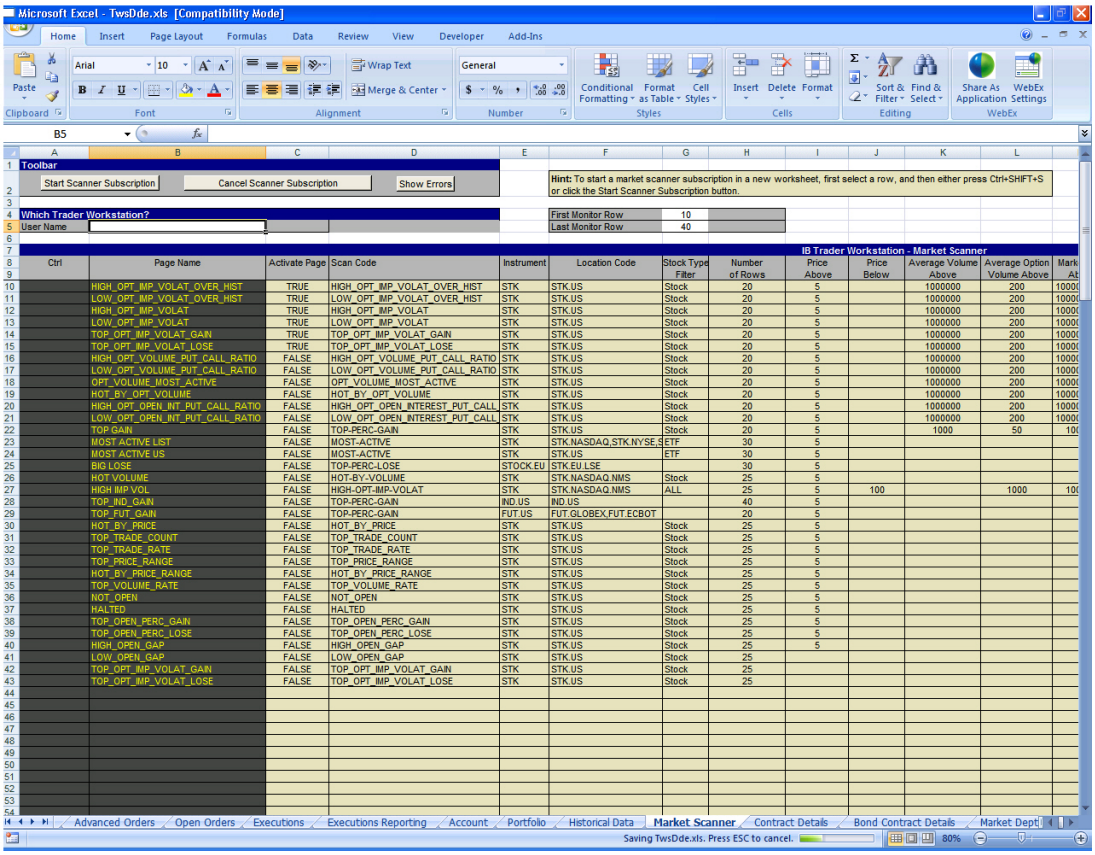
Notice that the date and time, the opening price, the high and low prices and, the closing price, all for that bar. You also see the trading volume, the number of trades that occurred (trade count) per bar, and the weighted average price.

Canceling Historical Data Requests

To cancel historical data requests which TWS cannot process, click the **Cancel Historical Data** button at the top of the Historical Data page. If you want to delete one of these pages, just right-click in the page title on the bottom tab, and select *Delete*.

Chapter 17 - The Market Scanner Page

The market scanners use the criteria you enter along with the scan code definition to return X number of the top matching contracts, all in a nice, new scanner page. For example, you might want to know the top 30 most active stocks currently trading on the NYSE. You can view these results on the above mentioned nice new scanner page, which by the way YOU named, and can even activate a feature that will make this page pop to the front of your application any time there are additions, deletions or any other changes to the list. Let's see how to do all this!



As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

How Do I Subscribe to a Market Scanner?

Look at the list of "Scan Codes" in the *Scan Code* field. This field, along with the rest of the subscription details, tells the API which data to return to you. We have included a bunch of them for you, but more could be added that you might want to see.

Just select a scan line, modify any of the filter elements if you want (they're described in the first of two tables below) and click the **Start Scanner Subscription**. Once it successfully subscribes to the scan, the value to the left of the name in the *Ctrl* column changes from *PROCESSING* to *SUBSCRIBED*.

You can find out what scan codes are available via the Java sample application. To do this, you must have a JDK (Java Development Kit) installed on your PC, which you can get from www.java.com. Use your Windows Explorer to navigate to the Jts/Java directory, under the API software directory. Open the Java file and you'll see a file called run.bat. Double-click this file and you'll be able to see the complete list of supported scan parameters.

Elements of a Market Data Scan

The following table describes the elements that make up a scanner subscription, in other words, the available fields into which you can enter the descriptive information that will get you what you want to see!

Watch out for the Activate Page field! It seems like a nice feature that gets your attention any time the results change. But what it will do is pop this scanner page to the front of the application regardless of what else is going on, such as you entering data on another worksheet or creating some complex and ingenious VBA code to perform some amazing feat. This feature doesn't care, it interrupts at will! If you find it interrupts too much, just change the value in this field to FALSE.

The following table lists the elements that make up a market data scan.

Element Name	Description
Page name	This is the name that will be given to the new page that is created with the scanner data.
Activate Page?	If set to true, the new scanner page will pop to the front of the worksheet pages every time a result returns. This could be as often as every minute.
Scan Code	Identifies the type of scan. See the *Take a Peek!* Note above for some instruction on how you can find the scan codes!
Instrument	Identifies the instrument type used in the scan. To see what instruments are available, you can look at the TWS Market Scanner instruments.

Element Name	Description
Location code	Identifies the market used (i.e. US Stocks) for the scan. To see what locations are supported, you can look at the TWS Market Scanner location codes.
Stock type filter	Allows you to specify just stocks, just etfs, or both.
Number of rows	Define the number of rows of data you want returned.
Price Above	Filters out returns with prices below the named price. Can be left empty.
Price Below	Filters out returns with prices above the named price. Can be left empty.
Average volume above	Filters out returns with an average volume below the named price. Can be left empty.
Average Option Volume Above	Filters out returns with an average option volume below the named price. Can be left empty.
Market Cap Above	Filters out returns with a market capitalization value below the named price. Can be left empty.
Market Cap Below	Filters out returns with a market capitalization value above the named price. Can be left empty.
Moody Rating Above	Filters out returns with a Moody rating below the named price. Can be left empty.
Moody Rating Below	Filters out returns with a Moody rating above the named price. Can be left empty.
S&P Rating Above	Filters out returns with an S&P rating below the named price. Can be left empty.
S&P Rating Below	Filters out returns with an S&P rating above the named price. Can be left empty.
Maturity Date Above	Filters out returns with a maturity date below the named price. Can be left empty.
Maturity Date Below	Filters out returns with a maturity date above the named price. Can be left empty.
Coupon Rate Above	Filters out returns with a coupon rate below the named price. Can be left empty.
Coupon Rate Below	Filters out returns with a coupon rate above the named price. Can be left empty.
Exclude Convertible	Filters out convertible bonds. Can be left empty.
Scanner Settings Pairs	For example "Annual/True" used on the Top Option Implied Vol% Gainers would instruct the scan to return annualized volatilities. Delimit setting pairs by slashes.

Available Market Scanners

The following table shows a list (current as of July 2008) of the available scanners.

Market Scanner (Scan Code)	Description
Low Opt Volume P/C Ratio (LOW_OPT_VOL_PUT_CALL_ RATIO)*	Put option volumes are divided by call option volumes and the top underlying symbols with the lowest ratios are displayed.
High Option Imp Vol Over Historical (HIGH_OPT_IMP_VOLAT_OVE R_HIST)*	Shows the top underlying contracts (stocks or indices) with the largest divergence between implied and historical volatilities.
Low Option Imp Vol Over Historical (LOW_OPT_IMP_VOLAT_OVE R_HIST)*	Shows the top underlying contracts (stocks or indices) with the smallest divergence between implied and historical volatilities.
Highest Option Imp Vol (HIGH_OPT_IMP_VOLAT)*	Shows the top underlying contracts (stocks or indices) with the highest vega-weighted implied volatility of near-the-money options with an expiration date in the next two months.
Top Option Imp Vol % Gainers (TOP_OPT_IMP_VOLAT_GAIN)*	Shows the top underlying contracts (stocks or indices) with the largest percent gain between current implied volatility and yesterday's closing value of the 15 minute average of implied volatility.
Top Option Imp Vol % Losers (TOP_OPT_IMP_VOLAT_LOSE)*	Shows the top underlying contracts (stocks or indices) with the largest percent loss between current implied volatility and yesterday's closing value of the 15 minute average of implied volatility.
High Opt Volume P/C Ratio (HIGH_OPT_VOLUME_PUT_C ALL_ RATIO)	Put option volumes are divided by call option volumes and the top underlying symbols with the highest ratios are displayed.
Low Opt Volume P/C Ratio (LOW_OPT_VOLUME_PUT_CA LL_ RATIO)	Put option volumes are divided by call option volumes and the top underlying symbols with the lowest ratios are displayed.
Most Active by Opt Volume (OPT_VOLUME_MOST_ACTIV E)	Displays the most active contracts sorted descending by options volume.
Hot by Option Volume (HOT_BY_OPT_VOLUME)	Shows the top underlying contracts for highest options volume over a 10-day average.
High Option Open Interest P/C Ratio (HIGH_OPT_OPEN_INTEREST _PUT_CALL_RATIO)	Returns the top 50 contracts with the **highest** put/call ratio of outstanding option contracts.

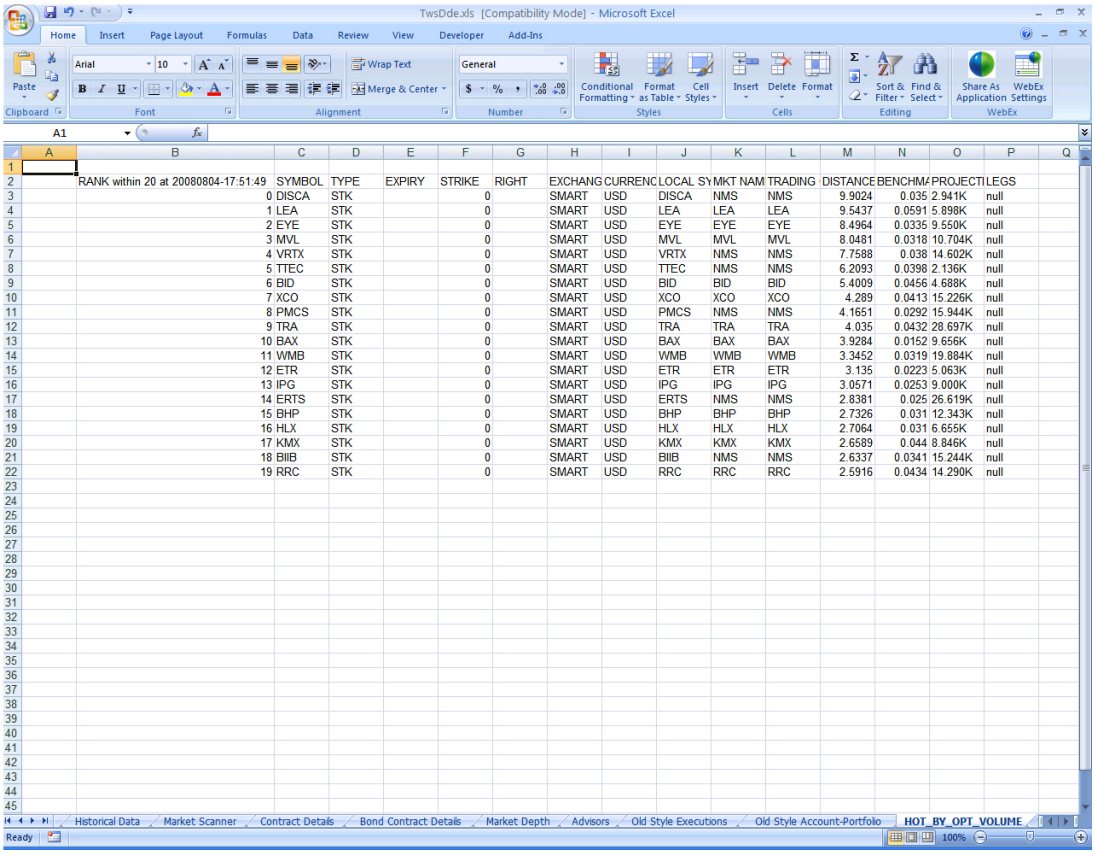
Market Scanner (Scan Code)	Description
Low Option Open Interest P/C Ratio (LOW_OPT_OPEN_INTEREST_PUT_CALL_RATIO)	Returns the top 50 contracts with the **lowest** put/call ratio of outstanding option contracts.
Top % Gainers (TOP_PERC_GAIN)	Contracts whose last trade price shows the highest percent increase from the previous night's closing price.
Most Active (MOST_ACTIVE)	Contracts with the highest trading volume today, based on units used by TWS (lots for US stocks; contract for derivatives and non-US stocks). The sample spreadsheet includes two Most Active scans: Most Active List, which displays the most active contracts in the NASDAQ, NYSE and AMEX markets, and Most Active US, which displays the most active stocks in the United States.
Top % Losers (TOP_PERC_LOSE)	Contracts whose last trade price shows the lowest percent increase from the previous night's closing price.
Hot Contracts by Volume (HOT_BY_VOLUME)	Contracts where: today's Volume/avgDailyVolume is highest. avgDailyVolume is a 30-day exponential moving average of the contract's daily volume.
Top % Futures Gainers (TOP_PERC_GAIN)	Futures whose last trade price shows the highest percent increase from the previous night's closing price.
Hot Contracts by Price (HOT_BY_PRICE)	Contracts where: (lastTradePrice-prevClose)/avgDailyChange is highest in absolute value (positive or negative). The avgDailyChange is defined as an exponential moving average of the contract's (dailyClose-dailyOpen)
Top Trade Count (TOP_TRADE_COUNT)	The top trade count during the day.
Top Trade Rate (TOP_TRADE_RATE)	Contracts with the highest number of trades in the past 60 seconds (regardless of the sizes of those trades).
Top Price Range (TOP_PRICE_RANGE)	The largest difference between today's high and low, or yesterday's close if outside of today's range.

Market Scanner (Scan Code)	Description
Hot by Price Range (HOT_BY_PRICE_RANGE)	The largest price range (from Top Price Range calculation) over the volatility.
Top Volume Rate (TOP_VOLUME_RATE)	The top volume rate per minute.
Lowest Option Imp Vol (LOW_OPT_IMP_VOLAT)	Shows the top underlying contracts (stocks or indices) with the lowest vega-weighted implied volatility of near-the-money options with an expiration date in the next two months.
Most Active by Opt Open Interest (OPT_OPEN_INTEREST_MOST_ ACTIVE)	Returns the top 50 underlying contracts with the (highest number of outstanding call contracts) + (highest number of outstanding put contracts)
Not Open (NOT_OPEN)	Contracts that have not traded today.
Halted (HALTED)	Contracts for which trading has been halted.
Top % Gainers Since Open (TOP_OPEN_PERC_GAIN)	Shows contracts with the highest percent price INCREASE between the last trade and opening prices.
Top % Losers Since Open (TOP_OPEN_PERC_LOSE)	Shows contracts with the highest percent price DECREASE between the last trade and opening prices.
Top Close-to-Open % Gainers (HIGH_OPEN_GAP)	Shows contracts with the highest percent price INCREASE between the previous close and today's opening prices.
Top Close-to-Open % Losers (LOW_OPEN_GAP)	Shows contracts with the highest percent price DECREASE between the previous close and today's opening prices.
Lowest Option Imp Vol (LOW_OPT_IMP_VOLAT)*	Shows the top underlying contracts (stocks or indices) with the lowest vega-weighted implied volatility of near-the-money options with an expiration date in the next two months.
Top Option Imp Vol % Gainers (TOP_OPT_IMP_VOLAT_GAIN)*	Shows the top underlying contracts (stocks or indices) with the largest percent gain between current implied volatility and yesterday's closing value of the 15 minute average of implied volatility.
Top Option Imp Vol % Losers (TOP_OPT_IMP_VOLAT_LOSE)*	Shows the top underlying contracts (stocks or indices) with the largest percent loss between current implied volatility and yesterday's closing value of the 15 minute average of implied volatility.

Market Scanner (Scan Code)	Description
13-Week High (HIGH_VS_13W_HL)	The highest price for the past 13 weeks.
13-Week Low (LOW_VS_13W_HL)	The lowest price for the past 13 weeks.
26-Week High (HIGH_VS_26W_HL)	The highest price for the past 26 weeks.
26-Week Low (LOW_VS_26W_HL)	The lowest price for the past 26 weeks.
52-Week High (HIGH_VS_52W_HL)	The highest price for the past 52 weeks.
52-Week Low (LOW_VS_52W_HL)	The lowest price for the past 52 weeks.
EFP - High Synth Bid Rev Yield (HIGH_SYNTH_BID_REV_NAT_YIELD)	Highlights the highest synthetic EFP interest rates available. These rates are computed by taking the price differential between the SSF and the underlying stock and netting dividends to calculate an annualized synthetic implied interest rate over the period of the SSF. The High rates may present an investment opportunity.
EFP - Low Synth Bid Rev Yield (LOW_SYNTH_BID_REV_NAT_YIELD)	Highlights the lowest synthetic EFP interest rates available. These rates are computed by taking the price differential between the SSF and the underlying stock and netting dividends to calculate an annualized synthetic implied interest rate over the period of the SSF. The Low rates may present a borrowing opportunity.

Viewing Market Scanner Results

Like the Historical Data subscription, the Market Scanner creates a new page and gives it the name you specified in the *Page Name* field. Your result will look something like this:



	RANK within 20 at 20080804-17:51:49	SYMBOL	TYPE	EXPIRY	STRIKE	RIGHT	EXCHANG	CURRENC	LOCAL SY	MKT NAM	TRADING	DISTANCE	BENCHMA	PROJECTI	LEGS
0		DISCA	STK		0		SMART	USD	DISCA	NMS	NMS	9.9024	0.035	2.941K	null
1		LEA	STK		0		SMART	USD	LEA	LEA	LEA	9.5437	0.0591	5.898K	null
2		EYE	STK		0		SMART	USD	EYE	EYE	EYE	8.4964	0.0335	9.550K	null
3		MVL	STK		0		SMART	USD	MVL	MVL	MVL	8.0481	0.0318	10.704K	null
4		VRTX	STK		0		SMART	USD	VRTX	NMS	NMS	7.7588	0.038	14.602K	null
5		TTEC	STK		0		SMART	USD	TTEC	NMS	NMS	6.2093	0.0398	2.136K	null
6		BID	STK		0		SMART	USD	BID	BID	BID	5.4009	0.0456	4.688K	null
7		XCO	STK		0		SMART	USD	XCO	XCO	XCO	4.289	0.0413	15.226K	null
8		PMCS	STK		0		SMART	USD	PMCS	NMS	NMS	4.1651	0.0292	15.944K	null
9		TRA	STK		0		SMART	USD	TRA	TRA	TRA	4.035	0.0432	28.697K	null
10		BAX	STK		0		SMART	USD	BAX	BAX	BAX	3.9284	0.0152	9.656K	null
11		WMB	STK		0		SMART	USD	WMB	WMB	WMB	3.3452	0.0319	19.884K	null
12		ETR	STK		0		SMART	USD	ETR	ETR	ETR	3.135	0.0223	5.063K	null
13		IPG	STK		0		SMART	USD	IPG	IPG	IPG	3.0571	0.0253	9.000K	null
14		ERTS	STK		0		SMART	USD	ERTS	NMS	NMS	2.8381	0.025	26.619K	null
15		BHP	STK		0		SMART	USD	BHP	BHP	BHP	2.7326	0.031	12.343K	null
16		HLX	STK		0		SMART	USD	HLX	HLX	HLX	2.7064	0.031	6.655K	null
17		KMX	STK		0		SMART	USD	KMX	KMX	KMX	2.6589	0.044	8.846K	null
18		BIIB	STK		0		SMART	USD	BIIB	NMS	NMS	2.6337	0.0341	15.244K	null
19		RRC	STK		0		SMART	USD	RRC	RRC	RRC	2.5916	0.0434	14.290K	null

Deleting Market Scanner Results

If you want to delete this page at any time, just right click on the tabbed title and select *Delete*.

Canceling a Scanner Subscription

If a scan doesn't seem to be processing, you can cancel it by clicking the **Cancel Scanner Subscription** button. It might also be helpful for you to see why it hasn't processed - perhaps you have an invalid value somewhere along the way. To find out if this is the case, click the Show Errors button and look at the last error code.

Chapter 18 - The Contract Details and Bond Contract Details Pages

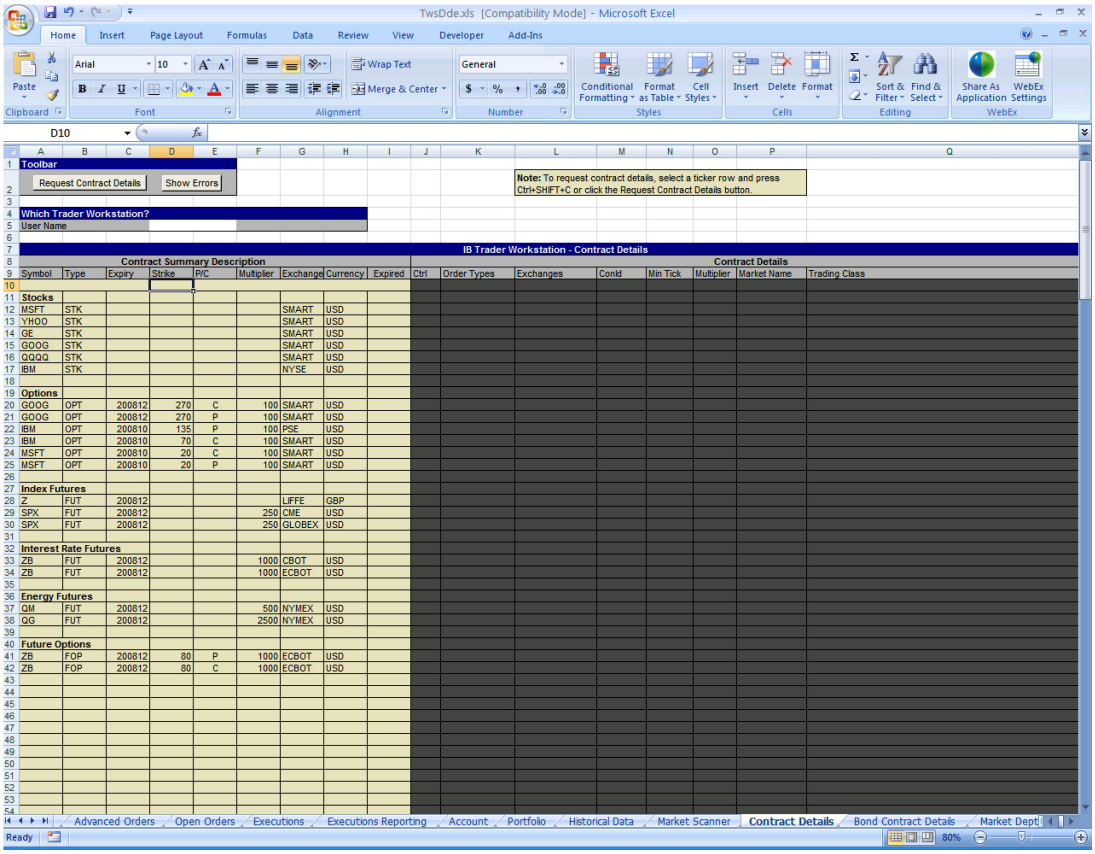
The Contract Details and Bond Contract Details pages may be two of the least complicated, yet most useful, of all the pages. Basically, use these two pages to get important contract-related information that you might need to help you enter valid parameters on other pages. What? Well for example, you might want to find out what order types are supported for specific instrument on a certain exchange. Use the Contract Details page! Or you might need the ConID or some coupon information for a bond to allow you to properly request a ticker. Use the Bond Contract Details page! The main reason these are separated is that the contract information for bonds is so different from that on other instruments, we'd have need a billion columns on one worksheet to cover them all.

As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

Contract Details Page

Follow the examples in the sample application to see how to enter the Summary Description. This basically identifies the contract for which you're trying to find more information. For bonds, you will need to know the CUSIP. To see how easy this is, let's use the Contract Details page and select a contract. Now, simply click the **Request Contract Details** button, and the

application effortlessly displays all the details you might need. You'll have to move down the list and highlight each instrument separately, and click the button each time.



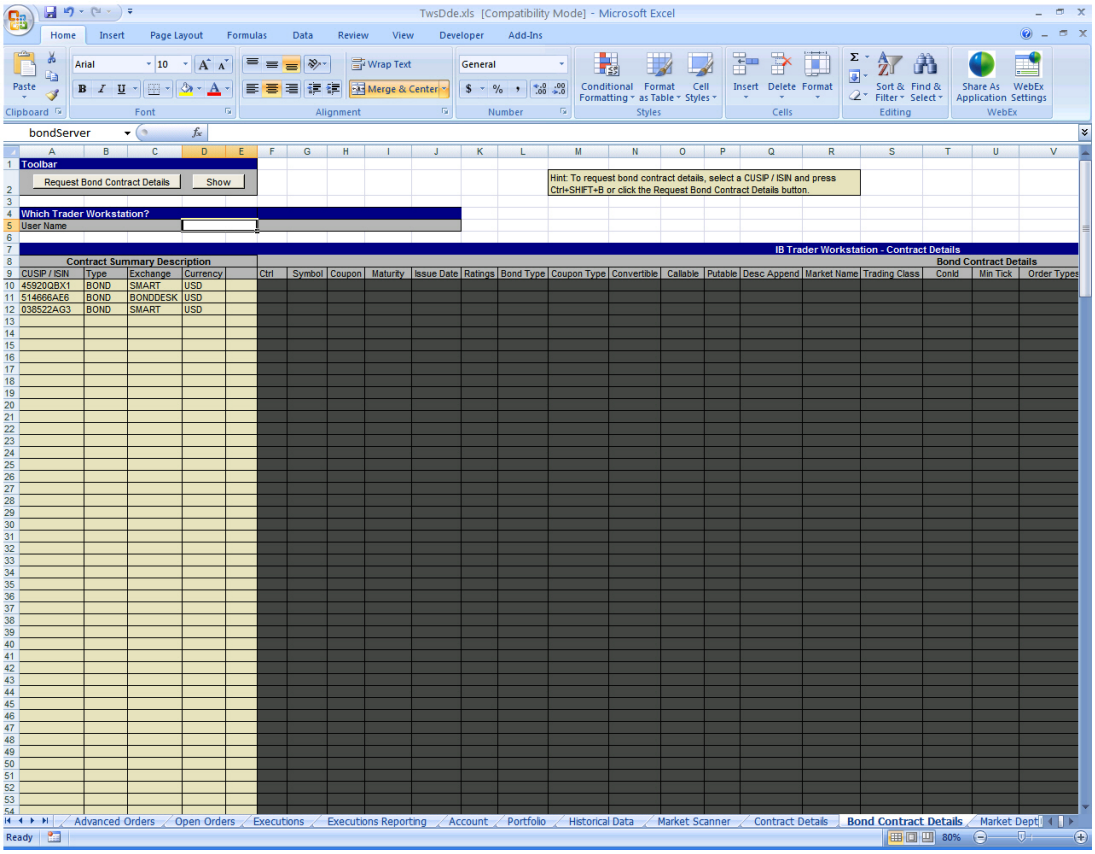
Expired Options

Under the Options heading, notice that since this application was written, some of the options have expired. If you try to get details on these expired options, all you'll get are zeros. Now, click the **Show Errors** button, and look at the last error message. It tells you that "No security definition has been found for the request." In other words, this doesn't exist anymore! If you just change the date to a valid expiry and request the details again, the API will be able to help you out.

Bond Contract Details Page

The Bond Contract Details page is very similar to the Contract Details page, aside from the column headings. And it works the same, too. Along with your TWS user name, you are also required to enter the contract summary description, so the application knows what contract you're trying to find out about. Follow the examples in the sample application to see how to enter the *Summary Description*. This basically identifies the contract for which you're trying to find more information. For bonds, you will need to know the CUSIP. Then you select the contract line and click the **Request Bond Contract Details** button and voila! If the data you entered is valid, the results appear as expected. If the data you entered was invalid, the

results won't display. In this case, use the **Show Errors** button to find out what went wrong, and see if you can rectify your error.



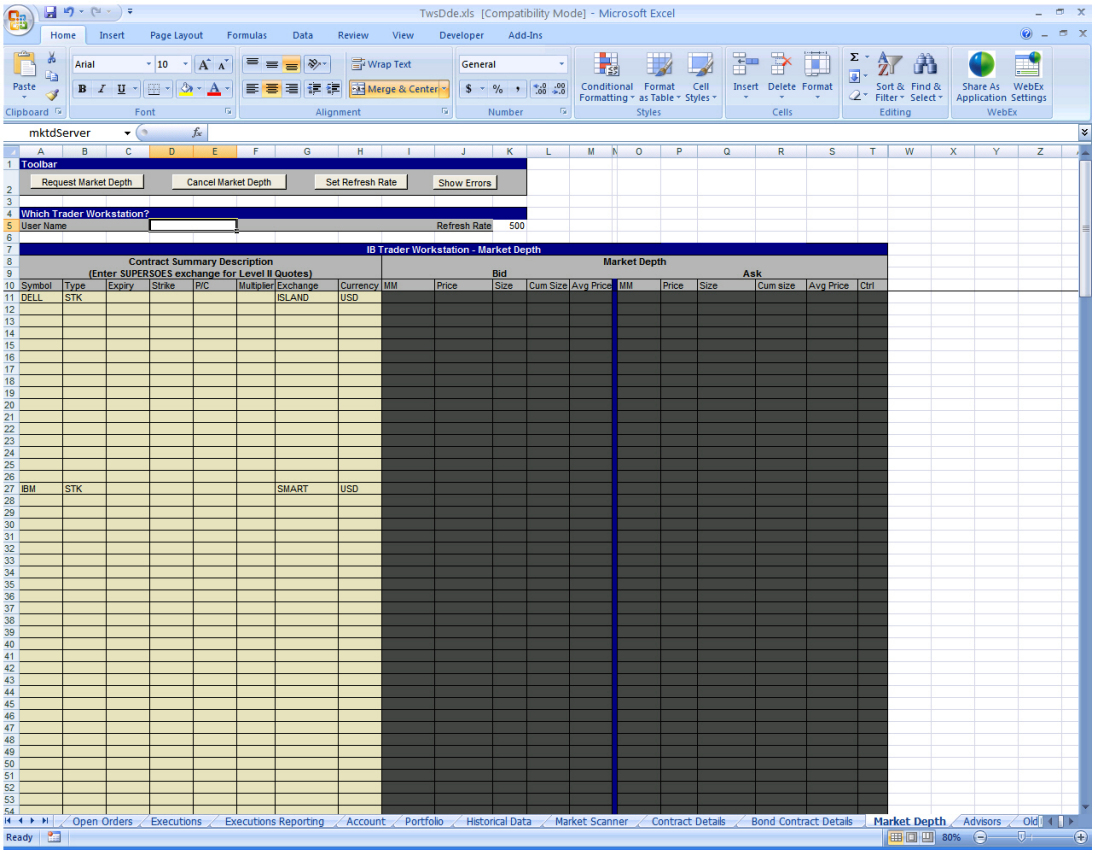
Using Contract Details on Other Pages

To use any of the data from the Contract Details or Bond Contract Details pages on other pages, copy and paste the contents of a cell.

Chapter 19 - The Market Depth Page

Market Depth shows you the alternative bids and offers away from the inside quote, which gives you a broader picture of what's happening and can help you to better gauge market liquidity.

To view market depth, enter a ticker in the *Contract Summary Description* fields, then click the **Request Market Depth** button. Market depth is displayed in the dark-shaded section of the page.



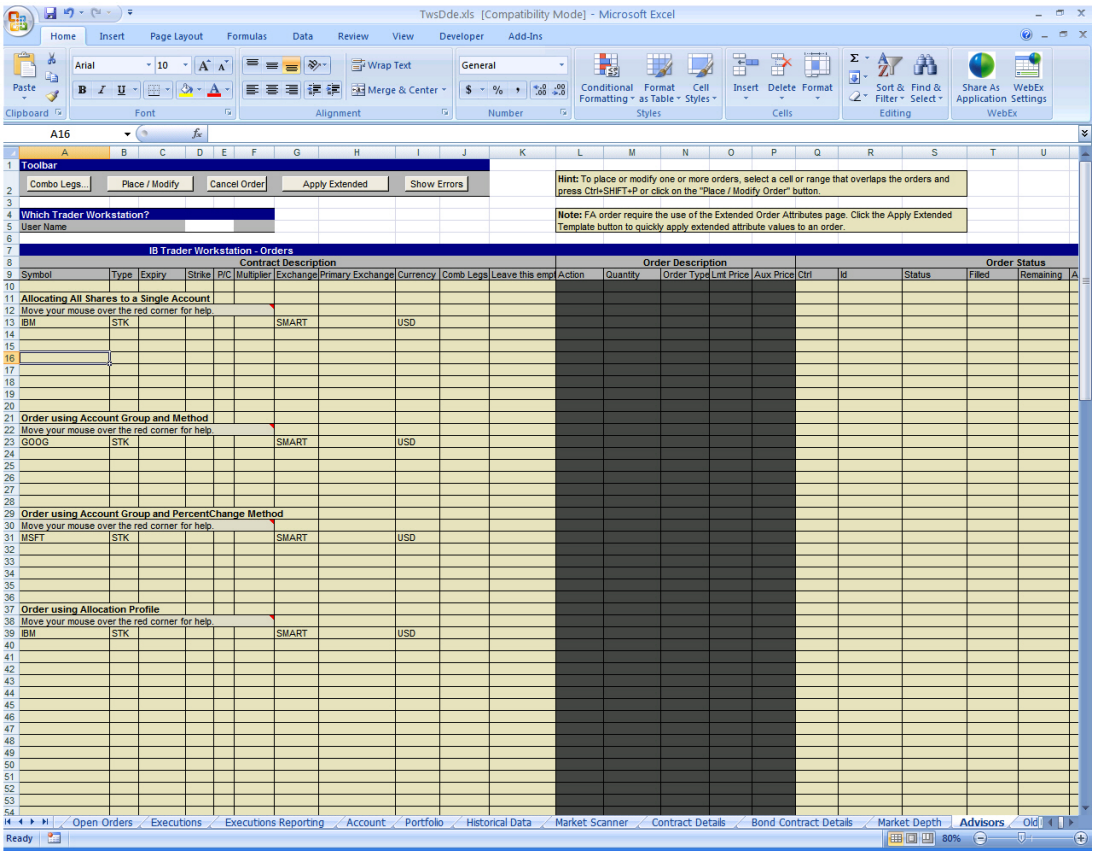
As always, make sure TWS is running, and don't forget to enter your username in the User Name field!

Canceling a Market Depth Request

If you want to cancel your market depth request, highlight the ticker row and click the **Cancel Market Depth** button.

Chapter 20 - The Advisors Page

Much of the content included in the DDE for Excel sample spreadsheet is applicable to both individual traders and advisors, but there are some specific areas of interest for those of you who handle multiple clients. This section focuses on the features that are only available to multi-client account advisors. We've included a special page, the Advisors page, where advisors can create orders for their multiple clients.



Note that at this time, although you can allocate portions of an order to multiple accounts via the DDE API, you must first configure account groups, account profiles, methods and percentages manually through TWS. You can check the *TWS Users' Guide* on the IB website for instructions on how to do this.

You might notice that there are some rows with a little red symbol on the upper right corner of some cells. If you roll your mouse over the red symbol, additional details about each type of FA order appear in a little popup. Of course, you're always encouraged to look at our TWS Users' Guide or the DDE for Excel section of the API Reference Guide if you're interested in learning more.

Setting Up Orders for an Advisor

Let's take a look at how to set up an order for an advisor using a single account, an account group, and an allocation profile. But first let's draw a hypothetical picture of who you are, with a descriptive scenario and some nuggets of information on which we'll base our share allocation examples:

- You are account number DF16045, and you manage 3 subaccounts named DU16046, DU16047, DU16048.
- In TWS, you have created an Account Group called **LowRisk**, added all the subaccounts to this group, and selected *AvailableEquity* as that group's default allocation method. This method dictates that the total number of shares in an executed order will be split between the accounts in the group based on the ratio of their available equity. If account DU16046 has available equity in the amount of $25,000, DU16047 has $50,000 and DU16048 has $100,000, the ratio is 1:2:4, and an order for 700 shares would see DU16046 with 100 shares, DU16047 with 200 shares and DU16048DU10705 with a whopping 400 shares of the order!
- In TWS, you have also created an Allocation Profile that you call **HighRisk**, set the allocation specifics to *Percentages*, and allocated 65% to DU16046 and 35% to DU16048. When you use this Allocation Profile, the total executed order quantity will be split 65/35.

That's all you need to know for now. Let's put our focus back on the Excel spreadsheet and allocate some shares!!

Allocate All Shares to a Single Account

Set up an order to buy 300 shares of YHOO[1] at the market price, and allocate all the shares in the order to a single account, say DU16047. OK, it sounds simple, but how do actually do it?

1 First open the Advisors page, and make sure you're linked up to your TWS account (your TWS user name is entered in the *User Name* field).

2 Enter the order description information as shown in the following figure:

8	Contract Description								Order Description			
9	Symbol	Type	B/	N	Exchange	Primary Exchange	Currency	Comb	Leg	Action	Quantity	Order Type	Lmt Price
10													
11	**Stocks**												
12	MSFT	STK			SMART	ISLAND	USD			buy	300	mkt	
13	YHOO	STK			SMART	ISLAND	USD			buy	300	mkt	

3 Now, you need to use the Extended Order Attributes page.

As we discussed earlier, you can enter these attributes on a per order basis using the area on the order line, or you can leave all that alone and use the Extended Order Attributes page. The catch is, some of these need to have a value of "0" to indicate that you don't want to use them, and it's easier to leave these filled out on the page than to go through and enter them manually for every order. Start by using the Extended Order Attributes page and end by viewing these same attributes on the order line itself.

a Open the Extended Order Attributes page. In this case, where you're giving the entire order to one account, you only need to use one of the attributes, specifically the *Account* attribute. It's labeled "Institutional Only" but don't let that scare you, it's for you too!

b Enter the account number in the *Value* field, and while you're in there take a look and see the other fields that contain values.

Remember, you only entered a value for the *Account* attribute; the other values are there by default.

4 Now let's go back to the Advisors page and scroll to the right to where you can see the Extended Order Attributes section, which is dark gray by default. Note that at this point, there are no values in any of these fields.

5 Select the YHOO order by highlighting the order line, and then click the **Apply Extended** button. If you scroll to the right now, you'll see that some of these fields have values, including the *Account* field, which displays the account number you entered. The template populated all of the others that require a value, and now you're ready to transmit the order. It will probably fill quickly since it's a market order.

6 To see that all of the shares in your order went to the account we specified, open the Executions page, enter your username and click the **Subscribe to Executions** button. Among other things, in the *Execution Description* area you'll see the Account number, Side (BOT) and Quantity of 300. Cool, huh?

Account Groups: There really IS a method …

Now you'll use the account group we created, and although we originally set this up to allocate shares using the Available Equity method, for this order we want all of the accounts in this group to get an equal share of the booty. Here's how easy it is to do that!

1 Submit an order for 700 shares of YHOO[1] and see how they get allocated equally between the three accounts in our group.

2 Start by opening the Extended Order Attributes page. First you need to delete the Account number specified in our last order from the *Account* field. Now, let's find the *FA Group* and *FA Method* fields.

Remember that the Extended Order Attributes page holds all values until you change them. In addition, if you don't manually enter any extended order attributes on your order line (starting with the *Time in Force*), the values on this page are automatically applied to your order EVEN IF you don't click the **Apply Extended** button!

3 Enter the FA Group name, which in this example is "LowRisk." Be careful to enter the name precisely. If you enter "Low Risk" or "lowrisk," the API won't recognize what you're telling it, and it will give you an error and fail to submit the order.

4 Next, enter the method. Since you want to distribute the shares evenly, enter **EqualQuantity** as the method, again being precise with your syntax.

5 Go back to the Advisors page to set up a buy market order for 700 shares of YHOO[1]. Hold on - don't click that **Apply Extended** button. To illustrate how the Extended Order Attributes page works, simply transmit the order.

6 Once it fills, go back to the Executions page and take a look - you'll see a total of 700 shares of YHOO[1] hypothetically BOT and distributed evenly amongst the three accounts in your group! But you're not done. Next, you'll test-drive the final allocation tool - the Allocation Profile.

The Allocation Profile Unmasked

Remember during you earlier work that you created an Allocation Profile called **HighRisk**, set those allocation specifics to *Percentages*, and set the allocation percentages to 65% for account DU16046 and 35% for account DU16048. Now you'll create an order using this profile.

1 Start by deleting the values in the *FA Group* and *FA Method* fields on the Extended Order Attributes page.

2 In the *Value* field for the *FA Profile* attribute, enter the exact name of our profile, **HighRisk**.

There's a field above the FA Profile field called FA Percentage. Although this might sound like a field we'd need for this Percentages profile, this is actually an Account Group Method field used to enter a percent value when the Account Group Method you select is PercentChange. Good thinking, though!

3 Go back to the Advisors page and create a market order for 1000 shares of YHOO[1].

4 Once the order fills, you can take another peek at our executions, and see that 650 shares were allocated to DU16046 and 350 went to DU16048. And in case you didn't feel like actually executing all of these allocation examples, here's what your execution reports look like:

IB Trader Workstation - Executions Subscription													
Contract Description							Execution Description						
Symbol	Type	xpir	ri	P/C	Exchange	Currency	Order Id	ecution	Time	Account	Exchange	Side	Quantity	Price
YHOO	STK				ISLAND	USD	000001b5.	0000ea	20061	DU16047	ISLAND	BOT	300	28
YHOO	STK				ISLAND	USD	000001b5.	000087	20061	DU16046	ISLAND	BOT	234	27.96
YHOO	STK				ISLAND	USD	000001b5.	000087	20061	DU16047	ISLAND	BOT	233	27.96
YHOO	STK				ISLAND	USD	000001b5.	000087	20061	DU16048	ISLAND	BOT	233	27.96
YHOO	STK				ISLAND	USD	000001b5.	000087	20061	DU16048	ISLAND	BOT	350	28.04
YHOO	STK				ISLAND	USD	000001b5.	000087	20061	DU16046	ISLAND	BOT	650	28.04

Advisor Fields

That's all folks! Except for reference, we've put together a table that tells you a bit about each of the advisor-oriented fields.

Advisor Field	Description	Valid Values
Account	Used to allocate all shares of an order to a single account. Use on the Advisors page and Extended Order Attributes page.	A valid account number. Note that an account alias WILL NOT be recognized. To see a list of your managed accounts, click the **Request Managed Accounts** button on the Account page.
FA Group	An advisor-defined account group comprising a set of subaccounts to which executed shares will be allocated based on an associated method. Use on the Advisors page and Extended Order Attributes page.	If you enter a value in this field, you also need a value in the *FA Method* field.
FA Method*	Defines the allocation method for the subaccount within an account FA group. Use on the Advisors page and Extended Order Attributes page.	EqualQuantity NetLiq AvailableEquity PctChange
FA Percentage	If the FA Method is PctChange, enter the percentage in this field. Use on the Advisors page and Extended Order Attributes page.	This method can only be used to modify an existing position.
FA Profile*	A profile allocates order shares to specific accounts by percentage, ratio, or an absolute number. Use on the Advisors page and Extended Order Attributes page.	Enter the name of the FA Profile. The allocation specifics have already been defined via TWS.
Acct Code	Found only on the Account page, used to define that account to which you want to subscribe.	A valid account number. Note that an account alias WILL NOT be recognized. To see a list of your managed accounts, click the **Request Managed Accounts** button.
Request Managed Accounts *button*	Located on the Account page, used to view a list of valid account numbers.	N/A

Advisor Field	Description	Valid Values
FA Managed Accounts *received cell*.	After you click the Request Managed Account button, the account values are displayed in this cell.	N/A

Chapter 21 - The Old Style Executions and Account-Portfolio Pages

The development team left two vestiges of the old-style screens: the Old Style Executions and the Old Style Account-Portfolio screens. They present the same data as the newer Executions, Account and Portfolio pages, but require you to click a button for each line of data. In addition, the latest and more efficient versions of both also break out the *Account* and *Portfolio* features to two pages, and include a new Executions Reporting page. These pages were left intact in deference to users who already have code linked to them. If by chance you happen to be one of these users, you should probably think about ending your dependence on these pages, and start using the "new fangled" pages instead.

Old Style Executions

The Old Style Executions page displays the current day's executions. As always, the most important thing to remember is to enter your username to identify the trader workstation to which you're linking up. Then click the **Request Executions** button. Each time you click you get one execution report. If you scroll to the right you'll see the Execution Filter Criteria box. You can elect to only see specific reports by entering a value in any of these filter fields. For example, if you enter **NYSE** in the *Exchange* field, you'll only get results for executions on the NYSE.

Old Style Account-Portfolio

The Old Style Account-Portfolio page displays everything about the status of your account and portfolio. Once you've entered your TWS username, simply click the **Req Account Value** button for each line of account information you want to view. The 32 lines in the *Account Values* section will all be filled with values, from your Equity with Loan Value through your margin and P&L values. The *Account Portfolio* section displays your portfolio for all of your current positions.

In Summary...

Well, you've done it. You've made your way through the DDE for Excel API spreadsheet, and you're still alive to talk about it! And along the way you've created the beginnings of your own Application Programming Interface. Hopefully you've also lain to rest some of the fears you had about tackling this type of project. It's not so intimidating when you break it down and take the time to focus on each piece separately.

5

Where to Go from Here

If you've come this far and actually read the book, you now have a pretty decent grasp on what the DDE API can do, and how to make it do some of the things you want. Now we'll suggest some helpful outside resources you can use to help you keep moving forward.

Here's what you'll find in this section:

Chapter 22 - Additional Resources

There are many resources out there that will be adequate in getting you where you need to go. If you have some books or places that you like, feel free to stick with them. The following are the resources we find most helpful, and perhaps they'll be good to you, too!

Help with Visual Basic for Applications Programming

For VBA help, your primary and first best resource should be the VBA help that comes with the program. To access this help, just press the F1 key from within the VBA. If you have a specific word highlighted the help will be specific to that topic; otherwise you'll get the opening page in the right pane, and a table of Contents, Answer Wizard and Index on the left. It's very well organized, but the best feature is the context-sensitivity. If you become comfortable with this tool you'll never be sorry.

For a broader view of VBA, a book might give you more. We recommend any of the following publications, some more basic than others:

- Walkenbach, John. *Excel 2003 Power Programming with VBA*. Indianapolis, IN: Wiley Publishing, 2004.
- Jacobson, Reed. *Microsoft Excel 2002 Visual Basic for Applications Step-by-Step*. Redmond WA: Microsoft Press, 2001.
- Bovey, Rob et al. *Excel 2002 VBA: Programmers Reference*. Indianapolis, IN: Wiley Publishing, 2003.
- Walkenbach, John. *Excel VBA Programming for Dummies*. Indianapolis, IN: Wiley Publishing, 2004.

Help with the TWS API

For help specific to the Excel TWS API, the one best place to go, really the ONLY place to go, is the Interactive Brokers website. Once you get there, you have lots of resources. Just type www.interactivebrokers.com in your browser's address line. Now that you're there, let me tell you where you can go.

Hold on - first we need to clarify to you that, as of this writing in November of 2006, the IB website looks as we're describing. IB has a tendency to revamp the look and organization of their site every year or two, so have a little patience if it looks slightly different from how it is described in this book. Here are the best places to find API help:

The API Reference Guide

The API Reference Guide includes sections for each API technology, including the DDE for Excel. The upper level topics which are shown directly below the main book are applicable across the board to all or multiple platforms.

To access the API Reference Guide from the IB web site, select *Application Programming* from the **Software** menu, then, then click the **Proprietary API** tab, then click *Reference Guide*. Click the **Online API Reference Guide** button to open the online guide, which contains a section devoted entirely to the DDE for Excel API.

The API Beta and API Production Release Notes

The beta notes are in a single page file, and include descriptions of any new additions to the API (all platforms) that haven't yet been pushed to production. The API Release Notes opens an index page that includes links to all of the past years' release notes pages. The index provides one-line titles of all the features included in each release.

To access these notesrom the IB web site, select *Application Programming* from the **Software** menu, then click the **Proprietary API** tab, then click either *API Beta Notes* or *API Release Notes*.

The API Highlights

This publication is very high level and designed to touch upon some key features of the different API platforms. It's most helpful as an aid in choosing the type of access technology you'll use to design your API. But if you are not a programmer, you'll probably want to stick with the DDE for Excel.

To view the API Highlights rom the IB web site, select *Application Programming* from the **Software** menu, then click the **Proprietary API** tab. Click *Getting Started Guide*, then select the *API Highlights* link in the text.

The TWS API Tour

The API tour presents a broad view of the API, its different access technologies, and a look at the DDE for Excel API sample spreadsheet. IB provides a page with different teaching tour that you can access from our web site. Click **Education**, select *Interactive Tours*, then select the *IB Application Program Interface Tour* link.

The TWS API Webinars

IB hosts free online webinars through WebEx to help educate their customers and other traders about the IB offerings. They present the API webinar about once per month, and have it recorded on the website for anyone to listen to at any time.

- To register for the API webinar, from the IB web site click **Education**, then select *Webinars*. On the **Live Webinars** page, click the **Application Program Interface** tab, then click the *TWS Application Programming Interface (API)* expandable topic. You will find a brief description of the webinar, the date and time of the next presentation, and a registration link.
- To view the recorded version of the API webinar, from the **Live Webinars** page click the **Watch Previously Recorded Webinars** button. Links to recorded versions of previously recorded webinars are listed on the page.

API Customer Forums

You can trade ideas and send out pleas for help via the IB customer base accessible through both the IB Bulletin Board and the Traders' Chat. The bulletin board includes a thread for the API, and thus provides an ongoing transcript of questions and answers in which you might find the answer to your question. The Traders' Chat is an instant-message type of medium and doesn't retain any record of conversations.

- "To view or participate in the IB Bulletin Board, go to the **Education** menu and click *Bulletin Boards & Chats*. Click the **Bulletin Board** tab, then click the **Launch IB Discussion Forum button** to access all of our bulletin boards, including the TWS API bulletin board.
- To participate in the Traders' Chat, you need to click the **Chat** icon from the menu bar on TWS. Note that both of these customer forums are for IB customers only.

IB Customer Service

IB customers can also call or email customer service if you can't find the answer to your question. However, IB makes it clear that the APIs are designed for use by programmers and that their support in this area is limited. Still, the customer service crew is very knowledgeable and will do their best to help resolve your issue. Simply send an email to:

api@interactivebrokers.com

IB Features Poll

The IB Features Poll lets IB customers submit suggestions for future product features, and vote and comment on existing suggestions.

From the IB web site, click **About IB**, then select *New Features Poll*. Suggestions are listed by category; click a plus sign next to a category to view all feature suggestions for that category. To submit a suggestion, click the *Submit Suggestion* link.

A

Appendix A - Extended Order Attributes

Attribute	Valid Values
timeInForce	DAY GTC OPG IOC
ocaGroup	String
account	The account number, used for institutional and advisor accounts.
open/close	O, C (for institutions)
origin	0, 1 (for institutions)
orderRef	String
transmit	0 (don't transmit) 1 (transmit)
Parent order Id	String (the order ID used for the parent order, use for bracket and auto trailing stop orders)
blockOrder	0 (not a block order) 1 (this is a block order)
sweepToFill	0 (not a sweep-to-fill order) 1 (this is a sweep-to-fill order)
displaySize	String (publicly disclosed order size)

Attribute	Valid Values
triggerMethod	Specifies how simulated Stop, Stop-Limit, and Trailing Stop orders are triggered. **0** - the default value. The "double bid/ask" method will be used for orders for OTC stocks and US options. All other orders will use the "last" method. **1** - use "double bid/ask" method, where stop orders are triggered based on two consecutive bid or ask prices. **2** - "last" method, where stop orders are triggered based on the last price. **3** - "double-last" method, where stop orders are triggered based on last two prices. **4** – "bid-ask" method. For a buy order, a single occurrence of the bid price must be at or above the trigger price. For a sell order, a single occurrence of the ask price must be at or below the trigger price. **7** – "last-or-bid-ask" method. For a buy order, a single bid price or the last price must be at or above the trigger price. For a sell order, a single ask price or the last price must be at or below the trigger price. **8** – "mid-point" method, where the midpoint must be at or above (for a buy) or at or below (for a sell) the trigger price, and the spread between the bid and ask must be less than 0.1% of the midpoint.
Hidden	Only valid for orders routed to Island. 0 - False 1 (order not visible when viewing market depth)
Discretionary Amount	Used in conjunction with a limit order to give the order a greater price range over which to execute.
Good After Time	Enter the date and time after which the order will become active. Use the format YYYYMMDD hh:mm:ss
Good 'Till Date	The order continues working until the close of market on the date you enter. Use the format YYYYMMDD. To specify a time of day to close the order, enter the time using the format HH:MM:SS. Specify the time zone using a valid three-letter acronym.
FA Group	For Advisor accounts only. The name of the Account Group.
FA Method	For Advisor accounts only. The share allocation method. EqualQuantity NetLiq AvailableEquity PctChange
FA Percentage	For Advisor accounts only. The share allocation percentage.
FA Profile	For Advisor accounts only. The name of the Share Allocation profile.
Short Sale Slot	For institutional accounts only; for SSHORT actions. 1 – If you hold the shares 2 – Shares will be delivered from elsewhere.
Short Sale Location	If shares are delivered from elsewhere, enter where in a comma-delimited list with no spaces. For institutional accounts only.

Attribute	Valid Values
OCA Type	1 = Cancel on Fill with Block 2 = Reduce on Fill with Block 3 = Reduce on Fill without Block
Rule 80A	Individual = 'I' Agency = 'A', AgentOtherMember = 'W' IndividualPTIA = 'J' AgencyPTIA = 'U' AgentOtherMemberPTIA = 'M' IndividualPT = 'K' AgencyPT = 'Y' AgentOtherMemberPT = 'N'
Settling Firm	Institutions only
All or None	0 = false 1 = true
Minimum Qty	Identifies the order as a minimum quantity order.
Percent Offset	The percent offset for relative orders.
Electronic Trade Only	0 = false 1 = true
Firm Quote Only	0 = false 1 = true
NBBO Price Cap	Maximum SMART order distance from the NBBO.
Auction Strategy	For BOX exchange only. match = 1 improvement = 2 transparent = 3
Starting Price	The starting price. For BOX orders only.
Stock Ref Price	Used for VOL orders to compute the limit price sent to an exchange (whether or not Continuous Update is used), and for price range monitoring. Also used for price improvement option orders.
Delta	The stock delta. For BOX orders only.
Stock Range Lower	The lower value for the acceptable underlying stock price range. For price improvement option orders on BOX and VOL orders with dynamic management.
Stock Range Upper	The upper value for the acceptable underlying stock price range. For price improvement option orders on BOX and VOL orders with dynamic management.
Volatility	The option price in volatility, as calculated by TWS ' Option Analytics. This value is expressed as a percent and is used to calculate the limit price sent to the exchange.

Attribute	Valid Values
Volatility Type	1 = daily 2 = annual
Reference Price Type	1 = average 2 = BidOrAsk
Hedge Delta Order Type	Prior to TWS Release 859, use "1" to send a market order, "0" for no order. After TWS 859, enter an accepted order type such as: MKT, LMT, REL.
Continuous Update	0 = false 1 = true
Hedge Delta Aux Price	Enter the Aux Price for Hedge Delta order types that require one.
Trail Stop Price	Used for Trailing Stop Limit orders only. This is the stop trigger price for TRAILLIMIT orders.
Scale Num Components	NO LONGER SUPPORTED
Scale Component Size	Used for Scale orders only, this value defines the order size of the each order component.
Scale Price Increment	Used for Scale orders only, this value is used to calculate the per-unit price of each component in the order. This cannot be a negative number.
Outside RTH	0 = false 1 = true

B

Appendix B - Account Page Values

Field	Description	Notes
Account Code	The account number.	
Account Type	Identifies the IB account type.	
Accrued Cash	Reflects the current month's accrued debit and credit interest to date, updated daily.	At the beginning of each month, the past month's accrual is added to the cash balance and this field is zeroed out.
Available Funds	**For securities:** Equity with Loan Value - Initial margin **For commodities:** Net Liquidation Value - Initial margin	
Buying Power	Cash Account : (Minimum (Equity with Loan Value, Previous Day Equity with Loan Value)-Initial Margin) Standard Margin Account : Available Funds*4	
Cash Balance	**For securities:** Settled cash + sales at the time of trade **For commodities:** Settled cash + sales at the time of trade + futures PNL	
Currency	Shows the currency types that are listed in the Market Value area.	
Cushion	Shows your current margin cushion.	
Day Trades Remaining	Number of day trades left for pattern day trader period.	
Day Trades Remaining T+1, T+2, T+3, T+4	The number of day trades you have left for a 4-day pattern day-trader.	

Field	Description	Notes
Equity With Loan Value	For Securities: • Cash Account: Settled Cash • Margin Account: • Total cash value + stock value + bond value + (non-U.S. & Canada securities options value) For Commodities: • Cash Account: Total cash value + commodities option value - futures maintenance margin requirement + minimum (0, futures PNL) • Margin Account: Total cash value + commodities option value - futures maintenance margin requirement	
Excess Liquidity	Equity with Loan Value - Maintenance margin	
Exchange Rate	The exchange rate of the currency to your base currency.	
Full Available Funds	For securities: Equity with Loan Value - Initial margin For commodities: Net Liquidation Value - Initial margin	
Full Excess Liquidity	Equity with Loan Value - Maintenance margin	
Full Init Margin Req	Overnight initial margin requirement in the base currency of the account.	
Full Maint Margin Req	Maintenance margin requirement as of next period's margin change in the base currency of the account.	
Future Option Value	Real-time mark-to-market value of futures options.	
Futures PNL	Real-time change in futures value since last settlement.	
Gross Position Value	Long Stock Value + Short Stock Value + Long Option Value + Short Option Value.	
Init Margin Req	Initial margin requirement in the base currency of the account.	

Field	Description	Notes
Leverage	For Securities: • Gross Position value / Net Liquidation value For Commodities: • Net Liquidation value - Initial margin	
Look Ahead Available Funds	For Securities: • Equity with loan value - look ahead initial margin. For Commodities: • Net Liquidation value - look ahead initial margin.	
Look Ahead Excess Liquidity	Equity with loan value - look ahead maintenance margin.	
Look Ahead Init Margin Req	Initial margin requirement as of next period's margin change in the base currency of the account.	
Look Ahead Maint Margin Req	Maintenance margin requirement as of next period's margin change in the base currency of the account.	
Maint Margin Req	Maintenance margin requirement in the base currency of the account.	
Net Liquidation	For Securities: • Total cash value + stock value + securities options value + bond value For Commodities: • Total cash value + commodities options value	
Net Liquidation by Currency	Same as above for individual currencies.	
Option Market Value	Real-time mark-to-market value of securities options.	
PNL	The difference between the current market value of your open positions and the average cost, or Value - Average Cost.	
Previous Day Equity with Loan Value	Marginable Equity with Loan Value as of 16:00 ET the previous day, only applicable to securities.	

Field	Description	Notes
Realized PnL	Shows your profit on closed positions, which is the difference between your entry execution cost and exit execution cost, or (execution price + commissions to open the positions) - (execution price + commissions to close the position).	
Reg T Equity	Initial margin requirements calculated under US Regulation T rules.	
Reg T Margin	For Securities: • Cash Account: Settled Cash • Margin Account: Total cash value + stock value + bond value + (non-U.S. & Canada securities options value) For Commodities: • Cash Account: Total cash value + commodities option value - futures maintenance margin requirement + minimum (0, futures PNL) • Margin Account: Total cash value - futures maintenance margin requirement	
SMA	Max ((EWL - US initial margin requirements)*, (Prior Day SMA +/- change in day's cash +/- US initial margin requirements** for trades made during the day.)) *calculated end of day under US Stock rules, regardless of country of trading. **at the time of the trade	Only applicable for securities.
Stock Market Value	Real-time mark-to-market value of stock	
Total Cash Balance	Cash recognized at the time of trade + futures PNL	
Total Cash Value	Total cash value of stock, commodities and securities	

Index