

INTRODUCTION

=====

This document explains what TWS and IBis are, what they can do and what they can't.

UNDERSTANDING TRADING SYSTEMS

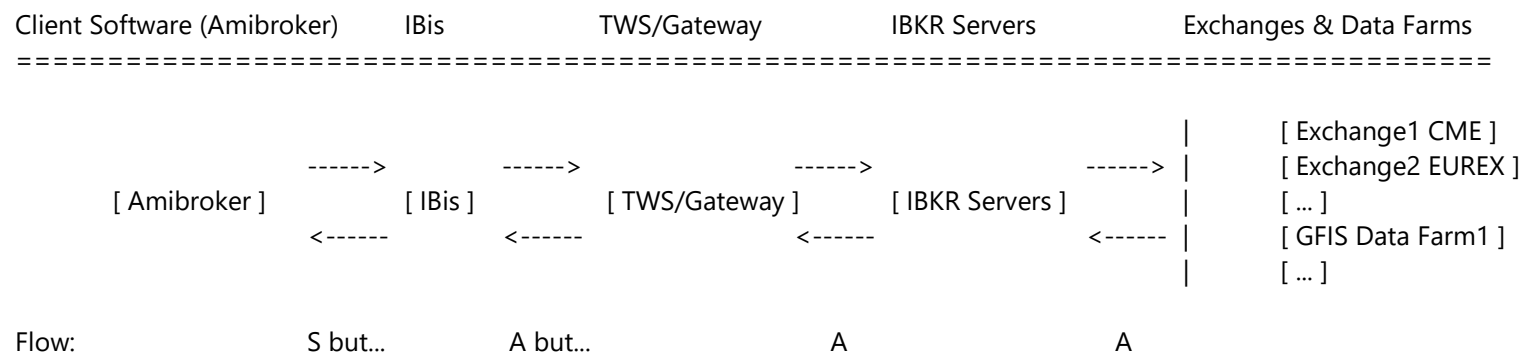
=====

The client is the software used by the user to perform operations. Typically it's Amibroker but could be anything supporting COM/OLE.

IBis is an intermediary to access TWS API easily, with minimal coding.

TWS is IBKR trading platform, used for charts, position and portfolio monitoring, manual trading... but also to provide services such as automatic order placement through its API. Gateway is another platform to provide the same services.

The following scheme illustrates how information comes and goes between your software, IBis, TWS, IBKR servers and exchanges.



There are two different types of communication in computing: synchronous (S) and asynchronous (A).

S: the caller sends a request to the callee and waits for the callee (almost instantaneous) answer.

A: the caller sends a request to the callee but doesn't wait for an answer. The caller can/will (generally) receive an answer later.

Communication between computers is generally A. Communication inside a software (but sometimes between) is generally S (when computation is fast).

Asynchronous also means that there is no order: if two requests R1 and R2 are sent in this order, you can get the answer to R2 before the answer to R1 !

IBKR servers receive information from exchanges and data farms asynchronously. Consequently there's a (small) delay between what happened at exchanges, what data is produced (by exchanges) and what servers receive.

TWS receives information from IBKR servers asynchronously. Thus there's an additional delay.

What IBis does...

=====

When you make a call/ask a question to IBis, in most cases you get an immediate answer. This can give you the illusion that everything is synchronous. It's not. IBis receives information from TWS asynchronously (TWS can be far away, on a remote compute). Thus there's an additional delay.

However some information is memorized (aka 'cached') by IBis, for instance contract specifications or margins. It can be returned immediately to the client, but in some cases this information (eg margins) it can be inaccurate (different from what will be required by IBKR when one places an order).

It does not mean that the answer is wrong, IBis is not faulty, it's the consequence of this system and there's no way to avoid this kind of problem.

When you place an order, IBis first performs a few checks to eliminate typically non-sense orders, second associates an order id to your order, third sends the order to TWS and fourth returns the order id to the client. 0 is returned in case of an error.

However some orders can be rejected later either by TWS, by IBKR servers (the broker) or by the exchanges (eg price/time limits/accuracy, size requirements...) The fact that you place an order does not mean that it's fine and will be executed. No way. You know your order is ok after the fact, when it appears in the list of pending or executed orders (or orders list). Also the status update of an order may be received in an unexpected order, for instance you could receive 'executed' before 'submitted' (remember it's asynchronous). IBis tries to mitigate this kind of problems by discarding updates which are (probably but not certainly) late. However there's no guarantee, it's only a mitigation.

Most beginners think of communication between systems synchronously. It's plain wrong and can lead to a lot of misunderstandings.

And by the way there's no way to get information instantaneously and thus synchronize everything, both because information cannot travel faster than light and because there's no universal clock (relativity).

From the client (AB) to the exchanges and back (such as order placement followed by status confirmation) can take from less than 100ms to a few seconds. This depends on a lot of factors (colocation of software at exchanges) and can change according to market activity, communication traffic, hardware outage...

When IBis does not have the answer, or when the answer could change, a new request is sent to TWS. If an answer is cached/memorized by IBis, it's returned immediately to the client (Amibroker). When an answer is received from TWS, it replaces the one previously memorized.

This new answer will be returned next time to the client. That is why sometimes you can get an answer in AB and at the next refresh (call to IBis) another answer. It's not wrong but updated. And some figures can change very often by small amounts (such as margins, adjusted by IBKR in real time)

Some information (such as the tape) is streaming. It's built from streaming quotes by IBis, updated as soon as possible and the resulting tape is cached waiting for a request.

Roadmap

=====

- immediate message processing
- achieve complete redundancy
- reduce latency between behind-the-scene information updates
- improved connection checks
- complete monotonic updates
- async logic checks with auto requests + distinct async errors
- complex orders toolkit (mutli-legs)
- batch orders (executed in lots with a delay between)
- persistent IBis state
- batch requests through file
- market specific availability
- replacement of ridiculous mandatory Intel decimal library by in-house light version
- command-line linux version

What can go wrong will...

=====

Sometimes, a message can be lost, typically between TWS and IBKR servers, more rarely between IBis and TWS.

IBis tries to mitigate this issue by performing redundant requests in some but not all cases. More cases should be

However there's no guarantee and no way to be sure no message is lost. The user must assume that a message can (and will) be lost anytime and that is why permanent human monitoring is required.

TWS communication with servers can be lost at the worse time, for instance when placing or cancelling an order.

Generally there's a consequence of your actions as an update of your positions, your orders, or your portfolio.

The best way to detect a message loss consists in looking for this consequence. If it does not happen soon, there's a high probability of a problem.

If nothing happens, read the messages. They are displayed in a tab in IBis and can be fetched by the client using a request `Exec("Messages")`.

It's especially important for short-term traders (daytraders & scalpers). But long term investors can be surprised too: orders could be not executed, not cancelled or even wrong. And remember that if there's a problem, it's yours. Rarely a brokerage can be considered legally responsible and never a software company.

Reality bites. Giving this heads-up is necessary. Do not blame IBKR, they're rather good at their job (brokerage). Blame dream sellers who want you to believe trading is easy. And never forget that

=====
PERMANENT HUMAN MONITORING OF TRADING ACTIVITY IS REQUIRED
=====