**期货行情接口FMAPI用户开发文档**

**一、接口概述**

FMAPI（Future Market Api）是基于CTP（中国金融期货交易所行情接口）封装的期货行情接口，提供期货行情数据的连接、订阅、采集、存储及查询功能。通过该接口，用户可获取实时Tick数据及5秒、1分钟、5分钟周期的K线数据，并支持数据本地存储（文件形式）与回调通知，适用于期货行情分析、策略开发等场景。

**二、基本功能**

**1. 核心功能模块**

FMAPI主要通过三类核心组件实现功能，涵盖行情连接、数据处理、存储查询等核心能力，为用户提供从行情获取到数据应用的全流程支持。

**2. 具体功能说明**

**（1）行情连接与登录**

支持设置行情前置地址、流文件路径、Tick数据输出路径、运行目录等初始化参数，满足不同环境下的配置需求。

提供登录、登出行情服务器的操作接口，自动维护与CTP服务器的连接状态，确保行情数据稳定获取。

支持接口授权认证，未授权状态下可能限制数据存储等部分功能的使用。

**（2）行情订阅与取消**

支持订阅/取消指定期货合约的实时行情，精准获取目标合约的动态数据。

支持订阅/取消报价请求，满足不同场景下对行情报价信息的获取需求。

**（3）数据采集与处理**

实时接收CTP行情数据，并转换为标准化Tick数据，统一数据格式，降低后续处理难度。基于Tick数据自动生成5秒、1分钟、5分钟周期的K线数据，无需用户手动计算，提升开发效率。

支持通达信格式K线数据转换为标准K线数据，兼容不同来源的K线数据，扩大数据应用范围。

支持源周期K线数据转换为目标周期K线数据（如5秒K线转1分钟K线），灵活满足不同分析周期的需求。

**（4）数据存储与查询**

支持Tick数据和K线数据本地文件存储，存储格式包括二进制、XML、JSON三种，用户可根据需求选择，且存储路径可灵活配置。

提供历史Tick数据和历史K线数据查询接口，方便用户回溯历史行情，用于策略回测、行情分析等场景。

支持从Tick文件数据转换生成K线数据，并可选择是否将生成的K线数据写入文件，丰富数据处理方式。

**（5）回调通知**

通过回调函数实时通知连接状态、登录结果、行情更新等关键事件，让用户及时掌握接口运行动态。

支持行情数据、Tick数据更新、K线数据更新（5秒、1分钟、5分钟周期）的实时回调，确保用户第一时间获取最新行情数据。

**三、行情输出基本流程**

**1. 初始化与连接流程**

* 调用初始化配置接口设置参数，包括流路径、Tick输出路径、回调函数、行情输出模式、运行目录，完成接口基础配置。
* 调用接口认证方法，对接口认证进行初始化。
* 配置行情输出模式，行情前置服务器地址等，调用登录接口登录服务器，发起连接请求。
* 登录成功则接口进入就绪状态，并通过回调通知用户；登录失败则触发错误回调，告知用户失败信息。
* 可选调用缓存配置接口，配置Tick和K线数据的缓存存储模式，优化数据读写效率。

**2. 行情订阅与数据处理流程**

* 接口就绪后，调用行情订阅接口订阅目标期货合约行情。
* 服务器确认订阅后，开始接收实时行情；若订阅失败，则触发错误回调，提示用户订阅异常。
* Tcik数据处理组件将接收的行情数据根据配置转换为标准化Tick数据，触发Tick更新回调，并按配置的存储方式和路径，将Tick数据存储到本地文件。
* K线数据处理组件基于行情数据，自动生成对应周期（5秒、1分钟、5分钟）的K线数据，触发对应周期K线更新回调，同时按配置将K线数据存储到本地文件。

**3. 数据查询流程**

* 用户调用历史Tick数据查询接口，Tick数据处理组件读取本地存储的Tick数据文件，返回Tick数据数组（该数组为动态分配，使用后需用户手动释放内存，避免内存泄漏）。
* 用户调用历史K线数据查询接口，K线数据处理组件读取本地存储的K线数据文件，返回K线数据数组（同样需用户在使用后手动释放内存）。
* 可选调用Tick转K线数据接口，从本地Tick文件中读取数据并转换生成K线数据，满足特殊场景下的K线数据获取需求。
* 支持对本地通达信K线数据的提取和转换，并根据配置是否写入到系统K线文件。

**四、接口调用注意事项**

**1. 初始化与权限**

接口初始化时，必须通过初始化配置接口设置流路径、输出路径、回调函数等必要参数，缺少关键参数可能导致初始化失败，影响接口正常使用。

接口需通过专属授权接口完成权限验证，若授权编码错误或未授权，可能会限制数据存储等核心功能的使用，需确保授权信息正确且有效。

确保Tick数据输出路径、运行目录等配置路径已存在且具备写入权限，若路径不存在或权限不足，会导致数据无法正常存储，影响数据回溯与查询。

**2. 内存管理**

通过历史Tick数据查询接口和历史K线数据查询接口获取的数据为动态分配的数组，用户在使用完毕后，必须通过delete[]手动释放内存，防止内存泄漏，保障程序稳定运行。

回调函数中传递的临时数据指针（如指向Tick数据、K线数据的指针）无需用户手动释放，由接口内部自动管理，避免误操作导致程序异常。

通过K线转换接口返回的K线数据数组，同样需要用户在外部手动释放，确保内存资源合理回收。

**3. 线程安全**

接口内部通过线程同步机制（如std::mutex）保证文件操作和数据缓存的线程安全，用户在多线程环境下调用接口时，无需额外添加锁机制，简化多线程开发流程。

回调函数可能在接口内部线程中执行，因此避免在回调函数中进行大量IO操作、复杂计算等耗时任务，防止阻塞行情数据接收线程，确保行情数据实时性。

**4. 连接与重连**

若行情连接意外断开（会触发断开回调通知），用户需及时调用重登录接口重新登录服务器，恢复行情数据接收，避免长时间断连导致关键行情数据丢失。

登录服务器前，需确保网络连接通畅，且配置的行情前置地址（如CTP行情服务器地址）正确无误，地址错误或网络异常会导致登录失败，无法获取行情数据。

**5. 数据格式与存储**

Tick数据文件存储路径格式为[配置的Tick输出路径]/Tick/[合约ID]\_[交易日].tick，数据记录按标准化Tick数据格式存储（1字节对齐），确保数据读取时格式统一。

K线数据文件存储时，通过文件后缀区分不同周期（如sec5代表5秒周期、min1代表1分钟周期、min5代表5分钟周期），数据记录按标准K线数据格式存储（1字节对齐），方便用户识别和读取对应周期数据。

若需要将通达信K线数据转换后存储，转换后的K线数据会按配置的存储模式（二进制、XML、JSON）写入文件，保持与其他K线数据存储模式一致，便于统一管理。

**6. 回调函数实现**

回调函数需能够处理多种不同类型的消息（如连接状态消息、行情更新消息等），通过消息标识参数区分不同消息类型，针对性进行处理。

回调函数中需判断批量消息标识参数（如bIsLast），确认是否为批量消息的最后一条数据，避免重复处理或遗漏处理数据，确保数据处理准确性。

**五、核心数据结构说明**

|  |  |  |
| --- | --- | --- |
| 数据结构 | 用途 | 关键字段 |
| TickData | 标准化Tick数据 | 时间戳（秒+毫秒）、最新价、开盘价、最高价、最低价、成交量、持仓量、买卖一档价格/成交量等 |
| KlineData | 标准K线数据（多周期） | 时间戳、开盘价、最高价、最低价、收盘价、成交量、成交额、持仓变化、成交次数 |
| KlineDataTDX | 通达信K线数据 | 日期时间、开盘价、最高价、最低价、收盘价、成交额、成交量 |

**六、开发者信息**

开发者：永轩软件

技术支持QQ：77766508

官方网站：www.yx10000.com