Telegram官网

Telegram官网开源生态说明页，介绍客户端代码公开、Bot API 与 TDLib 开发者工具及开源范围差异。

https://www.tao-zhifeiji.ltd/

Telegram官网

客户端代码公开、Bot API 与 TDLib 开发者工具的开源范围差异，一次说清。

客户端代码公开，但开源范围有边界

Telegram官网所呈现的开源生态并非“全部开放”，而是有选择地公开不同组件。官方客户端（Android、iOS、桌面版及 Web 版）的源代码在公开仓库中可获取，允许开发者审查、编译和修改。但这不代表服务端代码同样公开——服务端逻辑与后端基础设施并未开源。

对于希望基于 Telegram 构建应用的开发者，官网明确区分了两类工具：Bot API 和 TDLib。两者面向不同场景，开源程度与使用方式差异显著。理解这些差异，是选择正确开发路径的前提。

适合谁阅读？

本页面向正在评估 Telegram 平台能力的开发者、技术决策者以及希望了解开源范围的研究者。不需要预先掌握 Telegram 内部架构，但具备基本的客户端-服务端概念会更有帮助。

如果你只想知道“能不能拿到全部源代码”，答案很直接：不能。但可以拿到的部分足以支撑大量自动化、机器人、自定义客户端等应用。

三个必须知道的开源事实

· 客户端代码公开：官方移动端、桌面端和 Web 客户端的源码可公开访问，允许第三方构建兼容客户端或学习其实现。

· Bot API 不是开源库：Bot API 是一种基于 HTTPS 的远程接口，开发者通过 HTTP 请求与 Telegram 服务器交互，无需下载或编译任何 Telegram 代码。

· TDLib 是跨平台库：TDLib（Telegram Database Library）以开源库的形式提供，封装了与 Telegram 服务器通信的完整逻辑，适合构建自定义客户端或深度集成。

Bot API 与 TDLib 的开源范围差异

Bot API：轻量、远程、无需编译

Bot API 让开发者通过简单的 HTTP 接口创建机器人。你不需要接触 Telegram 客户端源码，也不需要编译任何 C++ 代码。只需向 api.telegram.org 发送请求，就能收发消息、管理群组、处理回调查询。
Bot API 本身不是以“开源库”形式分发的，而是作为网络服务存在。开发者使用的 SDK（如 Python、

Node.js 封装）由社区或官方提供，但核心通信仍通过远程调用完成。

这种设计降低了入门门槛，但灵活性受限于 API 暴露的能力。例如，Bot 无法主动发起与用户的私聊，也无法完全模拟真实用户的所有操作。

TDLib：完全开源、本地编译、深度控制

TDLib 是一个完整的开源库，用 C++ 编写，并提供了多种语言的绑定（如 Java、Python、Go、JavaScript）。开发者将 TDLib 编译进自己的应用后，可以获得与官方客户端几乎相同的底层能力，包括完整的消息同步、文件传输、群组管理、语音通话等。

与 Bot API 不同，TDLib 需要开发者自行管理网络连接、更新循环和本地数据库。这意味着更高的复杂度，但也带来了更大的控制权。TDLib 的开源范围包含完整的客户端逻辑，但不包含 Telegram 服务端代码。

“开源的不是 Telegram 的全部，而是足够让你构建自己客户端的核心工具集。”

关键结论

客户端代码公开意味着你可以审查、修改并重新发布自己的 Telegram 客户端，但必须遵守相应的开源许可证（通常为 GPLv2 或更高版本）。

Bot API 适合快速构建自动化、通知、客服机器人，无需维护底层连接。

TDLib 适合需要完全控制客户端行为的场景，如自定义聊天应用、嵌入式消息系统或研究项目。

服务端代码始终不公开，这是 Telegram 官网开源生态说明中最重要的边界。

选择适合你的开发路径

访问 Telegram 官方文档，查看 Bot API 参考或下载 TDLib 源码，开始构建你的应用。