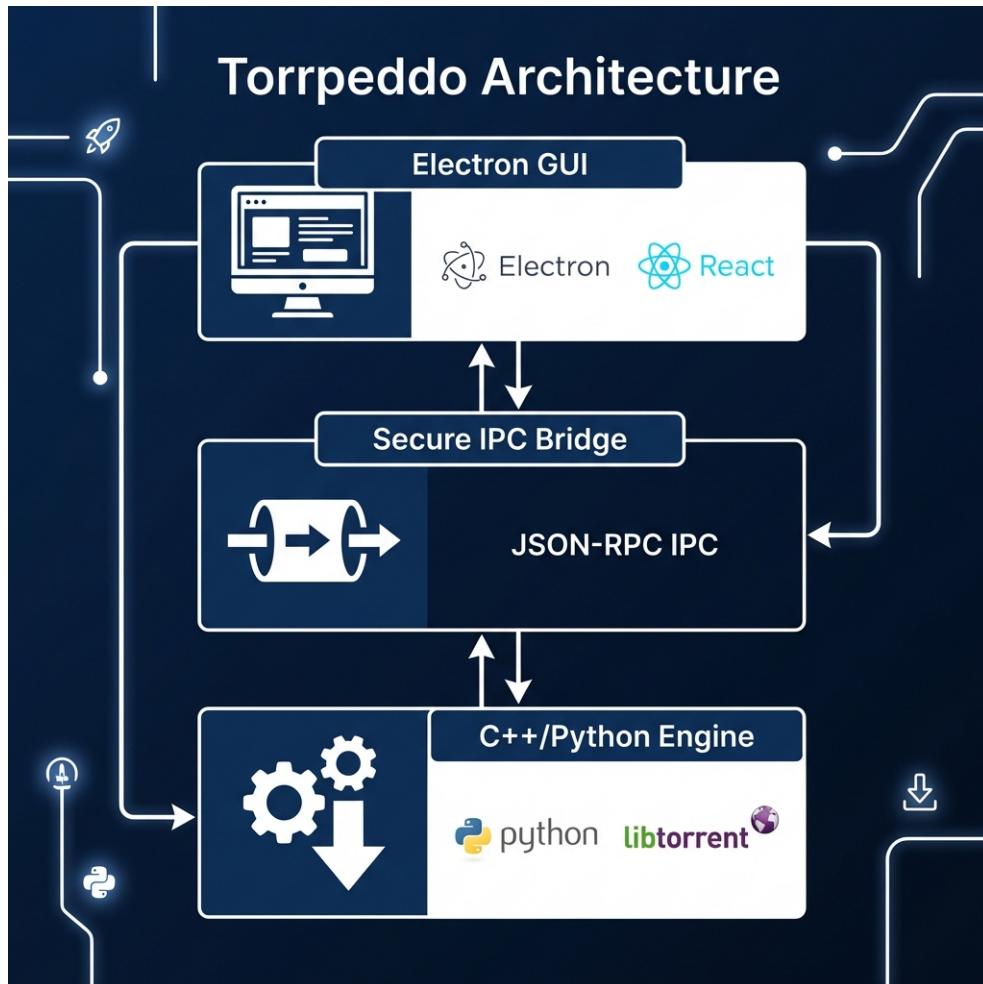


TORRPEDDO PROJECT BOOK



Executive Summary

Torpeddo is an industrial-grade, premium torrent client designed for the modern desktop. Built primarily with Python and the Electron framework, Torpeddo leverages the powerful `libtorrent` suite to offer a seamless, high-performance experience that bridges the gap between complex network protocols and professional user interfaces.

Architectural Deep Dive

Torpeddo follows a decoupled architectural pattern, separating the presentation layer from the core logic and network engine. This is achieved through three primary layers:

1. Frontend: Electron Framework

What is Electron?

Electron is an open-source framework that allows developers to build cross-platform desktop applications using web technologies (HTML, CSS, and JavaScript) and the Node.js runtime environment.

Benefits for Torpeddo:

- Visual Excellence: Leverages modern web technologies and components to create a highly responsive and visually appealing user interface.

2. The Bridge: IPC (Inter-Process Communication)

What is IPC?

IPC, or Inter-Process Communication, is a mechanism that allows different processes to share data and communicate with each other. In the context of Torpeddo, the IPC bridge connects the Electron frontend to the core logic and network engine.

Implementation: Secure JSON-RPC

Communication is handled via a secure JSON-RPC (Remote Procedure Call) protocol.

Why this approach?

- Decoupling: The engine logic is decoupled from the UI, allowing for independent development and testing of each layer.

3. Backend Engine: Python & libtorrent

The Core: libtorrent with Python Bindings

Multi-threaded Performance:

At the heart of Torpeddo is the libtorrent implementation. While the interface is in Python, the core logic is in C++. Torpeddo utilizes the C++ bindings for seamless integration with the libtorrent engine.

- Engine Level: The `libtorrent` engine handles the heavy lifting of disk I/O, network polling, and managing the state of multiple torrent fragments.

Development Process & Methodology

The Torpeddo project followed a "Platform-First" methodology:

1. Language Choice: Python was chosen for its ease of integration with the libtorrent C++ bindings.

(c) 2026 Torpeddo Team. All rights reserved.