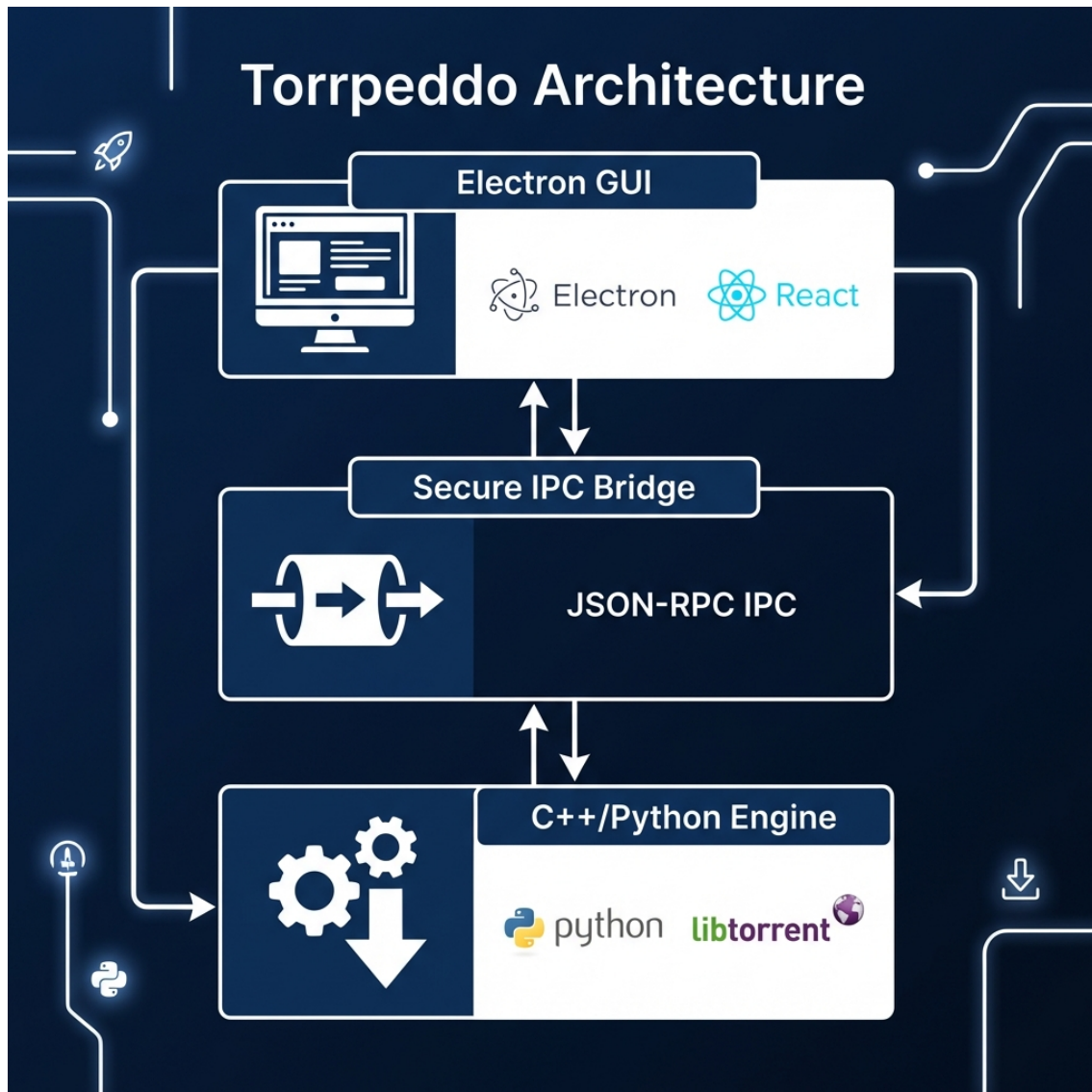


TORRPEDDO PROJECT BOOK



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 for building cross-platform desktop applications using JavaScript. It combines the Chromium runtime (for system

Benefits for Torpeddo:

- Visual Excellence
components to create

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

What is IPC?

IPC, or Inter-Process Communication, is a mechanism for processes to share data and resources. In the context of Torpeddo, it connects the Electron

Implementation: Secure JSON-RPC

Communication is

Why this approach?

- Decoupling: T

touching the UI.

3. Backend Engine: Python & libtorrent

The Core: libtorrent with Python Bindings

At the heart of Torpeddo is the Backend Engine. While the underlying architecture is complex, it utilizes the official libtorrent Python bindings with the bridge layer to manage the data flow.

Multi-threaded Performance:

- Engine Level: The engine is designed for high performance, handling I/O, network polling, and managing multiple torrent files simultaneously.

Development Process & Methodology

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

1. Language Choice: The choice of Python was driven by its ease of integration with the libtorrent C++ library and its rapid development cycle.

(c) 2026 Torpedo Team. All rights reserved.