Electron npm Electron versioning doc Elect
ad-hoc Electron (, x64 ia32),
Format asar tar - Electron ASAR
c++ , , . NET DMG macOS DMG
Java C++ JavaScript IPC IPC Electron IPC main( ) renderer( ) JSO
Content module (, Blink V8) “libcc” github.com/electron/libchromiumconten
main process, commonly a file named main.js, is the entry point to every Electron app. It controls
the life of the app, from open to close. Dock The main process is responsi
new renderer process in the app. The full Node API is built in. package.json
Electron : process, renderer process MAS Apple's Mac App Store For details
app to the MAS, see the Mac App Store Submission Guide. Mojo IPC , , Chror
chromium.googlesource.com/chromium/src/+/master/mojo/README.md native modules Native
modules (also called addons in Node.js) are modules written in C or C++ that can be loaded into
Node.js or Electron using the require() function, and used as if they were an ordinary Node.js
module. JavaScript Node.js C/C++ Electron Node Electron
Node Modules. NSIS Nullsoft Scriptable Install System Windows
NSIS OSR OSR (Off-screen) ( )
main process, renderer process renderer process The renderer process is a browser windc
your app. Unlike the main process, there can be multiple of these and each is run in a separate
process. web Electron Node.js API
autoUpdater API Squirrel userland "userland" "userspace" Unix
Node Electron API Electron Us
V8 C++ Electron V8 Chromium Node V8 V8 Google
v8.dev nodejs.org/api/v8.html docs/development/v8-development.md webview webview tags are
used to embed 'guest' content (such as external web pages) in your Electron app. They are similar
to iframes, but differ in that each webview runs in a separate process.