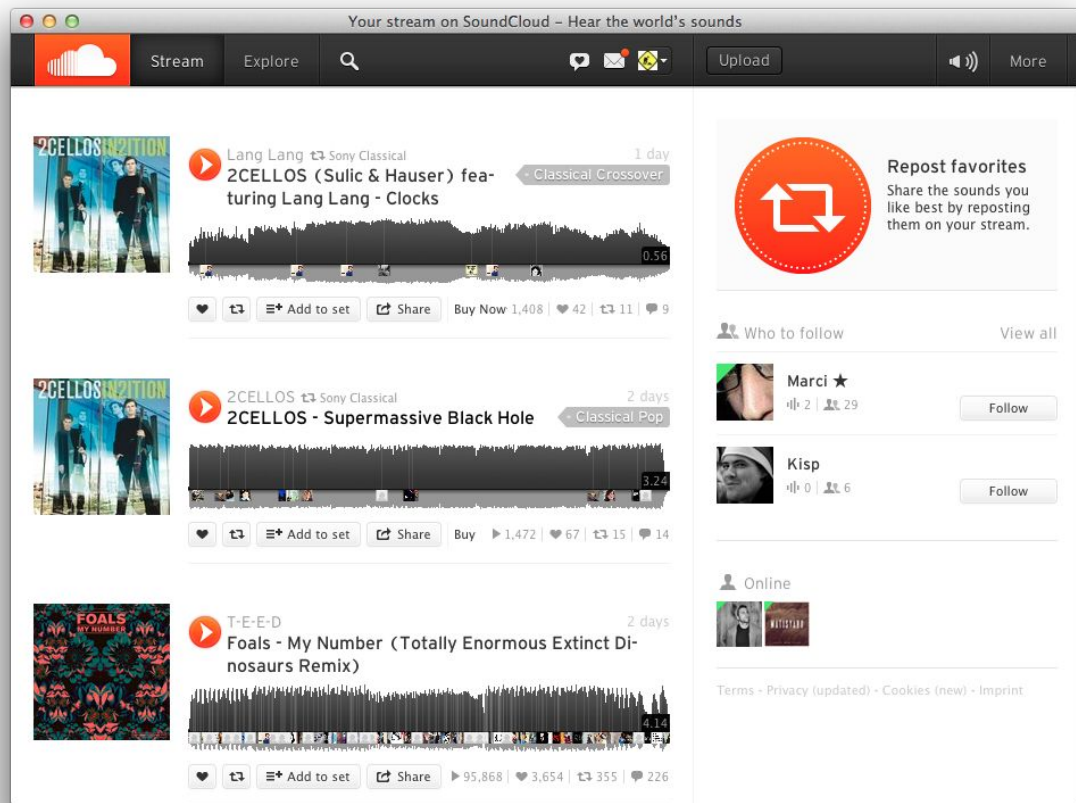


Delivering Applications with Electron


Márton Salomváry
@salomvary

What is SoundCloud?



What is SoundCleod?

```
- (void)webView:(WebView *)sender decidePolicyForNavigationAction:(NSDictionary *)actionInformation
    request:(NSURLRequest *)request frame:(WebFrame *)frame decisionListener:(id)listener
{
    // window.open navigation
    if(![self isFirstLoad] || [PopupController isLoginURL:[request URL]]) {
        // new popup can only opened with login url, from there navigation
        // anywhere is allowed
        [listener use];
        [self setIsFirstLoad:FALSE];
        [window setIsVisible:TRUE];
    } else {
        [listener ignore];
        // open external links in external browser
        [[NSWorkspace sharedWorkspace] openURL:[actionInformation objectForKey:WebActionOriginalURLKey]];
    }
}
```



What is SoundCleod?

- SoundCleod was originally Objective-C
- Not cross platform
- UIWebView based on an old Safari
- WKWebView broke everything
- Installer was 1.58 MB



What is SoundCleod?

“Business logic” was rewritten to Electron **in an afternoon**.

*

Shipping the first version with Electron took **a week**.

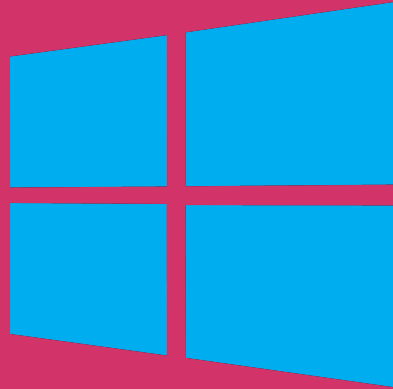
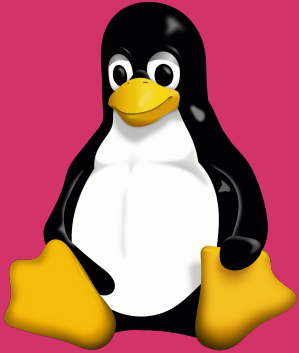


What is Electron?



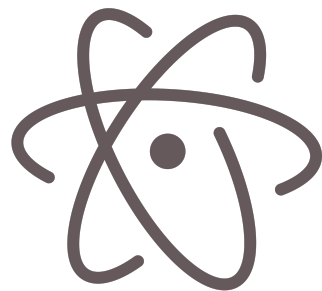
Chromium
+
Node.js
+
Operating system APIs

Works on Windows, Linux and macOS



What is Electron?

- Created and maintained by GitHub
- Originally for their Atom editor
- Now used by mainstream apps



Electron 101

```
const {app, BrowserWindow} = require('electron')
```

```
app.on('ready', () => {  
  const win = new BrowserWindow()  
  
  win.loadURL('file://some/local/file.html')  
  // or  
  win.loadURL('http://example.com')  
})
```

See the [Electron Quick Start](#) for a better example.



Electron 101

```
npm install -g electron  
electron main.js
```



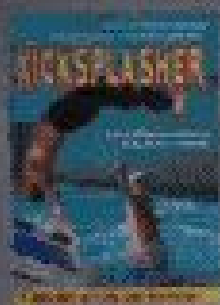
Electron, Node.js and modern web APIs
will be available in the browser window!



Example Domain

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

[More information...](#)




But now you need
to ship it



Shipping means...

Not `npm publish`.

- Write desktop glue
 - Write platform specific glue
 - Bundle app code + Electron binary + installer
 - ...for each platform
 - Publish somewhere
 - Provide automatic updates
- 




Desktop glue

Desktop glue

- Electron core is very barebones
- Provides low-level APIs
- Everything else is community modules*
- (Many common problems are actually solved)
- But not all size fits all
- Still need to write your own glue

* The usual wide range range of “community code” quality applies.



Desktop glue

Examples of “missing” features:

- Default application menu
- Browser-like features (eg. `window.open`)
- No error handling UI (network, SSL)
- No about box



Platform specific glue

```
if (process.platform == 'darwin')  
else if (process.platform == 'linux')  
else if (process.platform == 'win32')
```

- Window management
- Menus
- Dock, start menu, etc.
- ...and many others



Application bundling

Application bundling

What you have during **development**:

- Your own code (*.js, *.css and other assets)
- node_modules
- Electron executable

What you want ship in **production**:

- A single file (executable, installer) to distribute
- Have you own branding



Application bundling

- Electron comes with no tools
- Doable manually but complicated
- Community projects for the rescue
 - ◆ electron-packager
 - ◆ **electron-builder** - the “complete solution”

Btw. a bundled application starts at 35MB.




Bundling with electron-builder

“A complete solution to package and build a ready for distribution Electron app with “auto update” support out of the box”

- Builds *from any for any* platform*
- Supports numerous target formats
- Supports code signing
- Uploads to GitHub releases (and others)

* Almost. Native builds and macOS code signing are exceptions. [Read more here.](#)



Latest release

v1.1.7

5e74e11

1.1.7









Edit



salomvary released this 2 days ago · **2 commits** to master since this release

- Add experimental swipe back/forward navigation
- Fix media key flakyness under some circumstances
- Fix crash on startup [#130](#)

Downloads

 latest-mac.json	126 Bytes
 RELEASES	80 Bytes
 soundcleod-1.1.7-full.nupkg	52.2 MB
 soundcleod-1.1.7-mac.zip	38.9 MB
 soundcleod-1.1.7.dmg	36.5 MB
 soundcleod-Setup-1.1.7.exe	53.1 MB
 Source code (zip)	
 Source code (tar.gz)	



Automatic updates

Automatic updates

- You will most likely want updates
- Unless using App Store for Mac or Windows
- ...or targeting *only* Linux

Electron has built-in updater, but:

- It has no user interface
- Requires a server :(





Source

Update server

- Server-side versioning logic
- Open source update servers exist
 - ◆ One of them is Nuts
- You will have host it yourself
- Runs on a free Heroku plan
- Backed by GitHub releases

Latest electron-builder added serverless updates.



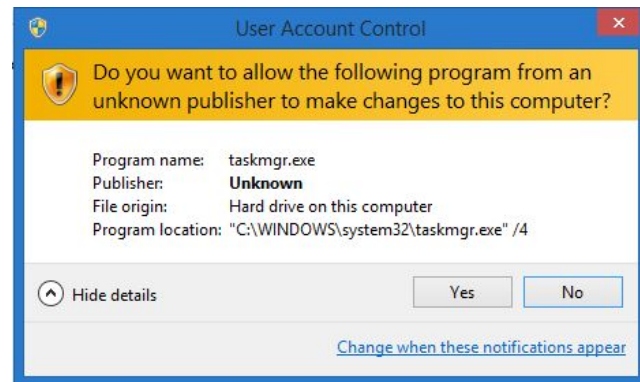
Code signing

Code signing

- Avoid scary warning dialogs
- Protect from attacks
- Certificate costs \$\$\$
- Required for automatic updates

Workaround: self-signed certificate

- Very little protection
- Makes automatic updates work



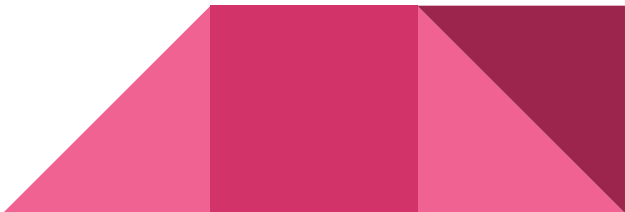
Continuous delivery

Continuous delivery

What you want:

- `npm test` on each commit
- ...on different platforms
- Do unattended releases

What you get:

- Travis CI and AppVeyor are free for open source
 - ...both super easy to set up
 - electron-builder can publish releases
- 

Travis CI example

os:

- osx

language: node_js

node_js:

- "6"

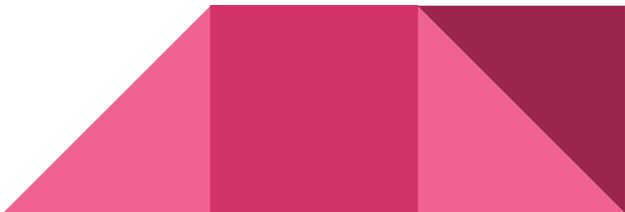
before_install:

See <https://github.com/travis-ci/travis-ci/issues/6307>

- rvm get head

script:


- npm run eslint
- npm test



Talking of testing...

- **Unit testing:** use your favorite framework
- **Integration testing:** use Spectron
 - ◆ Part of the official “Electron suite”
 - ◆ Electron API + Webdriver
 - ◆ No API for “desktop interactions” :(

Recommended reading

- [Electron Quick Start](#)
 - [How Slack was built with Electron](#)
 - <https://github.com/atom/atom>
 - <https://github.com/salomvary/electron-hello-world>
 - <https://github.com/salomvary/soundcleod>
 - <https://www.npmjs.com/search?q=electron>
- 



Thanks!

Márton Salomváry
@salomvary