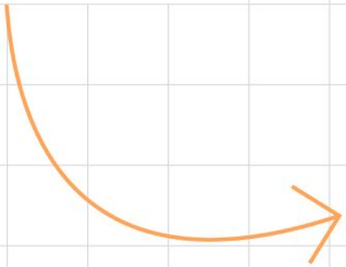
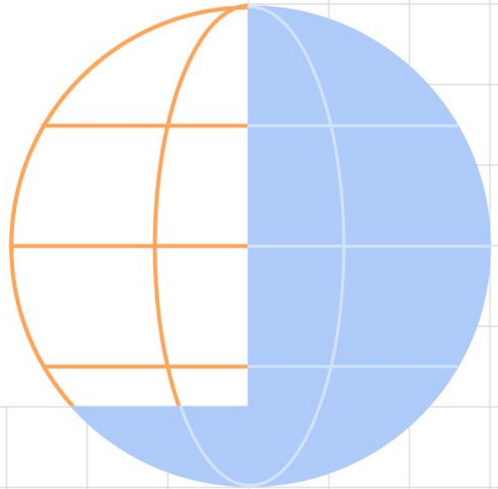
Google Developers



Electronify your Angular app

Run your Angular app in desktop with Electron

Aristeidis Bampakos
GDE Greece
@abampakos



Hello all ☺

Front End Web Developer at Plex-Earth

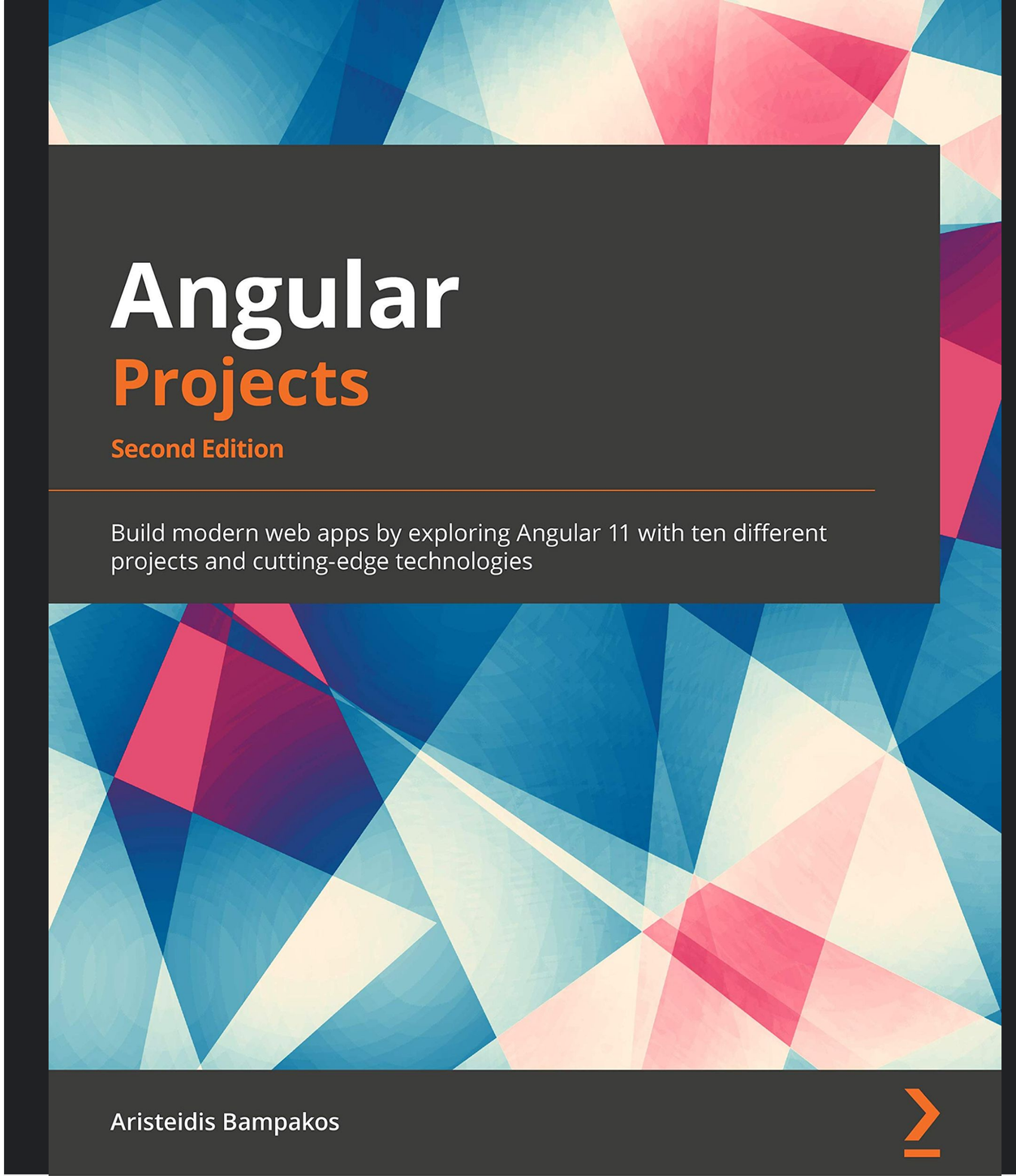
- Angular GDE
- Co-organizer Angular Athens
- Senior Angular Instructor at Code.Hub
- Author for Packt Publishing
- Assassins Creed fan

Coming soon

Available to pre-order

https://amazon.com/dp/1800205260

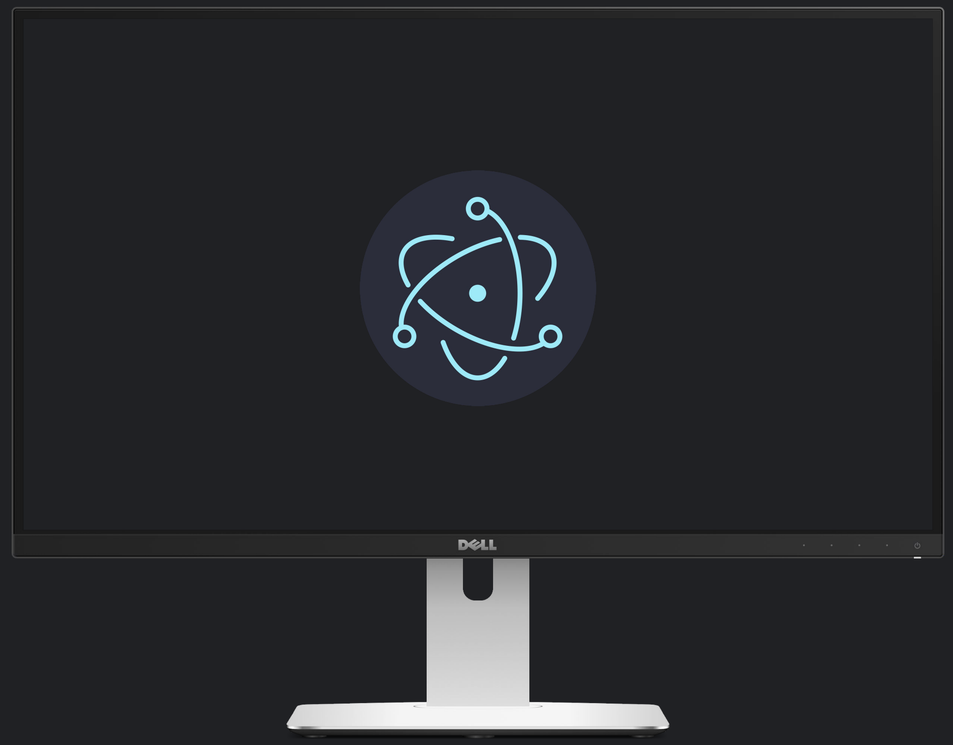
Experts



ElectronJS

npm install -D electron

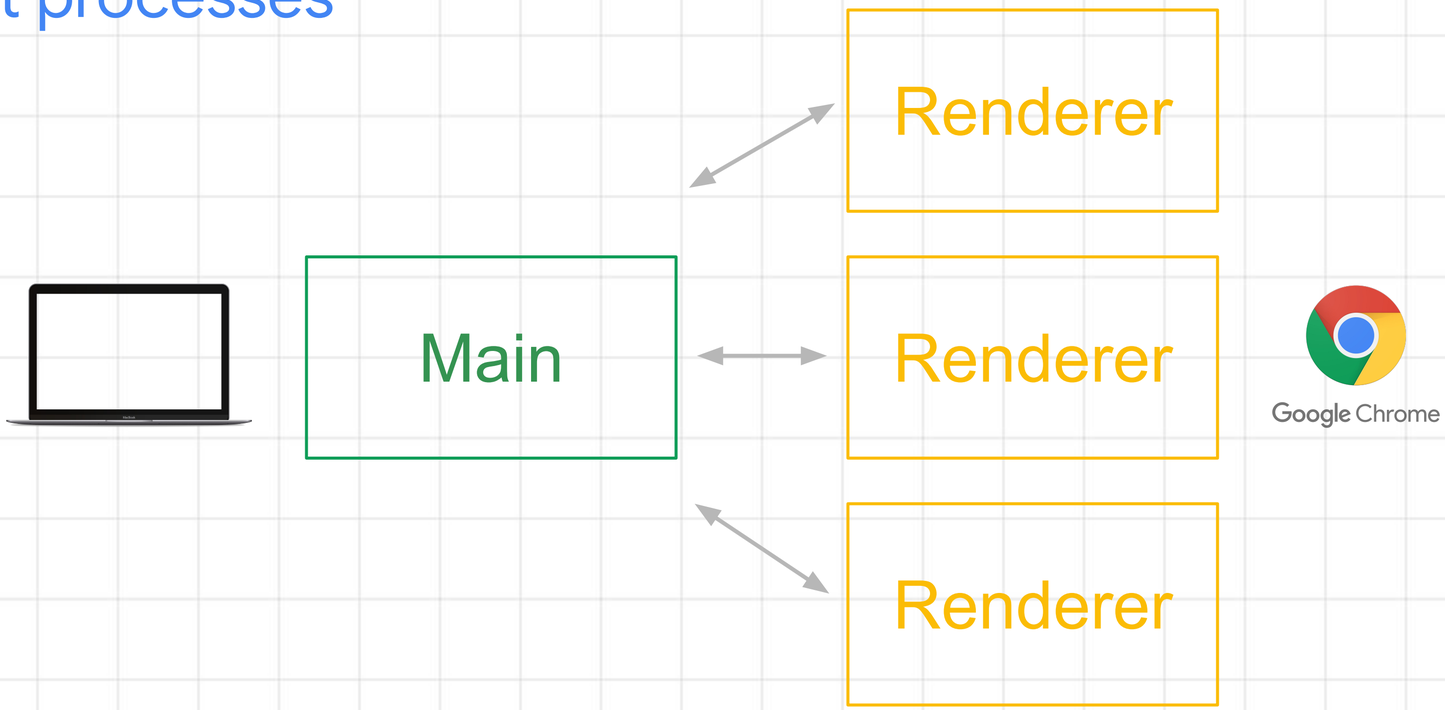
Cross-platform desktop apps with HTML, CSS and JavaScript.



Experts

Electron architecture

Two different processes



```
import { app, BrowserWindow } from 'electron';

function createWindow () {
  const mainWindow = new BrowserWindow({
    width: 800,
    height: 600
  });

  mainWindow.loadFile('index.html');
}

app.whenReady().then(() => {
  createWindow();
});
```

main.js

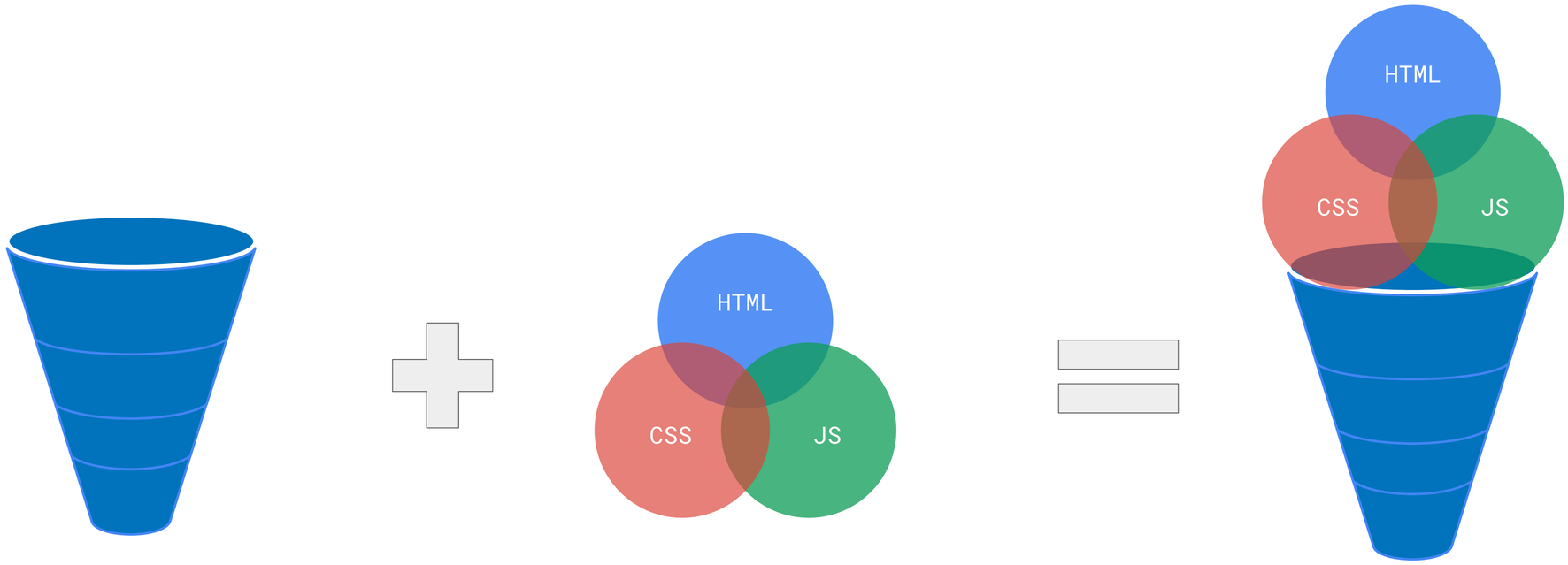
Build pipeline

Electron build process

Experts

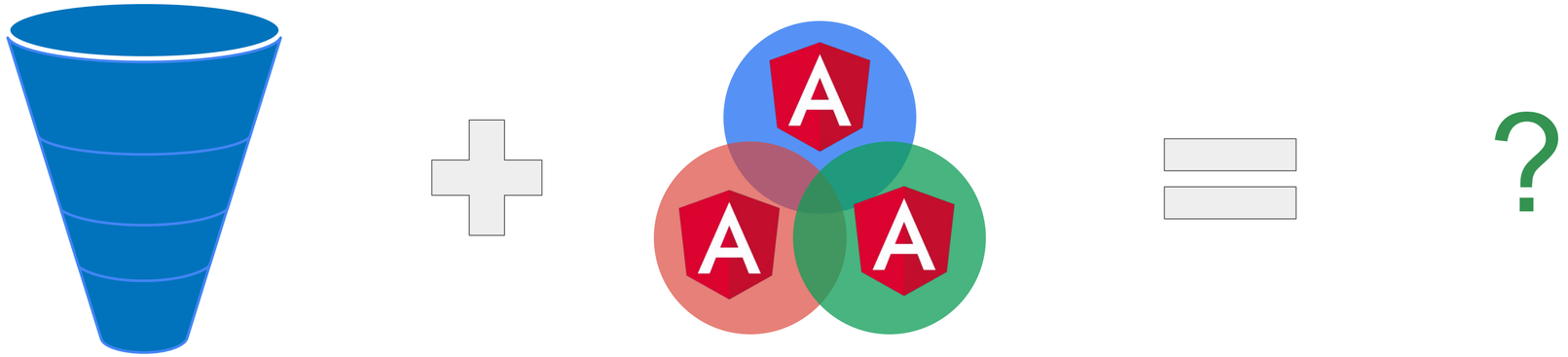
Electron app

High level overview



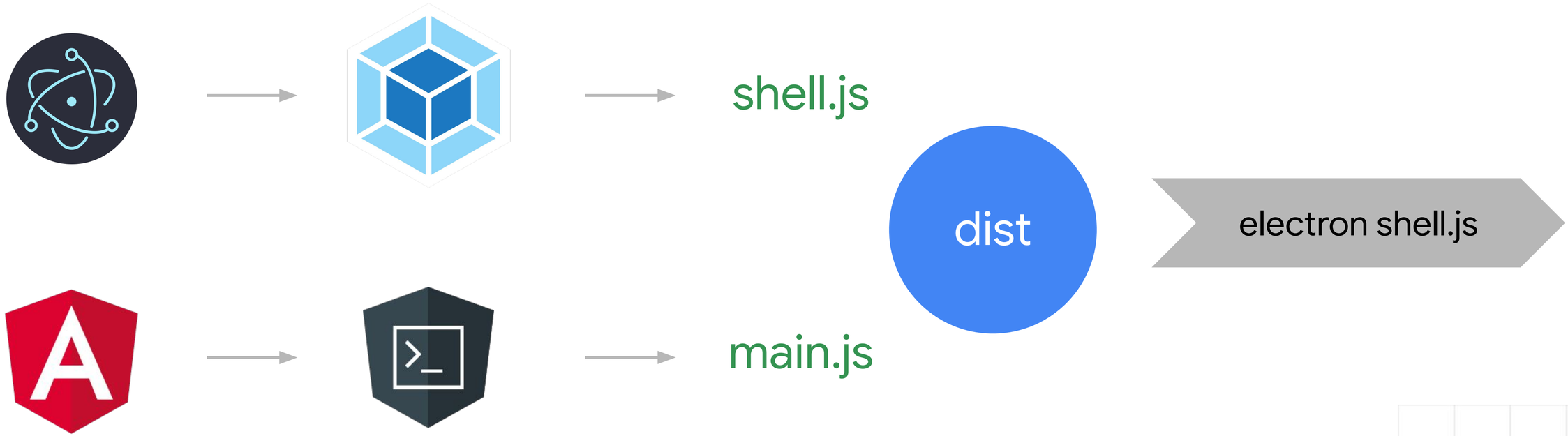
Hybrid app

Angular flavor



Build pipeline

Angular and Electron build process



```
import { app, BrowserWindow } from 'electron';

function createWindow () {
  const mainWindow = new BrowserWindow({
    width: 800,
    height: 600
  });

  mainWindow.loadUrl('http://localhost:4200');
}

app.whenReady().then(() => {
  createWindow();
});
```

main.js (URL based)

File vs URL approach

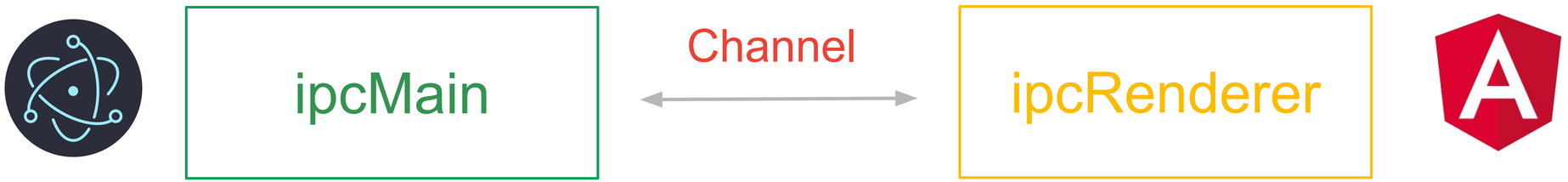
Compare different techniques

- More secure
- Complex deployment
- Unified build
- Difficult to update

- Potential security issues
- Simple deployment
- Separate build
- Easy to update

Inter-Process Communication (IPC)

Asynchronous communication between processes



```
import { app, BrowserWindow, ipcMain } from 'electron';

function createWindow () {
  const mainWindow = new BrowserWindow({
    width: 800,
    height: 600,
    webPreferences: {
      nodeIntegration: true
    }
  });
}

ipcMain.handle('pirates', () => {
  const pirates = ['Henie Greybeard', 'Captain Fragile'];
  return pirates;
});
```

main.js (excerpt)

```ts
import { Injectable } from '@angular/core';
import { ElectronService } from 'ngx-electron';

@Injectable()
export class PirateService {

  constructor(private electronService: ElectronService) { }

  getPirates(): Promise<string[]> {
    return this.electronService.ipcRenderer.invoke('pirates');
  }

}
```

pirate.service.ts

ngx-electronify

ng add ngx-electronify

Angular library to run an Angular app in desktop mode.

Experts

Next version?

Upcoming features

- Export as desktop app
- Integrate Angular DevTools (#11)
- Interact with Electron API

Electron security

What about security

Electron is not a browser
but acts like it

Experts

Why Electron?

Do I need it?

- Emulate multi-tab environment
- Provide native look and feel
- Tool for an existing desktop app

Resources

Helpful links

- Electron
- Electron Quickstart
- Security Best Practises
- Securing Electron apps
- ngx-electronify
- Angular Electron sample
- ngx-electron

Google Developers

Experts

Thank You!

Aristeidis Bampakos
GDE Greece
@abampakos