Migrating a JavaScript Application to TypeScript

Chris B. Behrens
Senior Software Architect

@chrisbbehrens

Gang of Four Patterns

Erich Gamma,
Richard Helm,
Ralph Johnson,
John Vlissides

Battlefield tactics

Derive the tactic
from circumstances

The Patterns

Observer

Pub-Sub

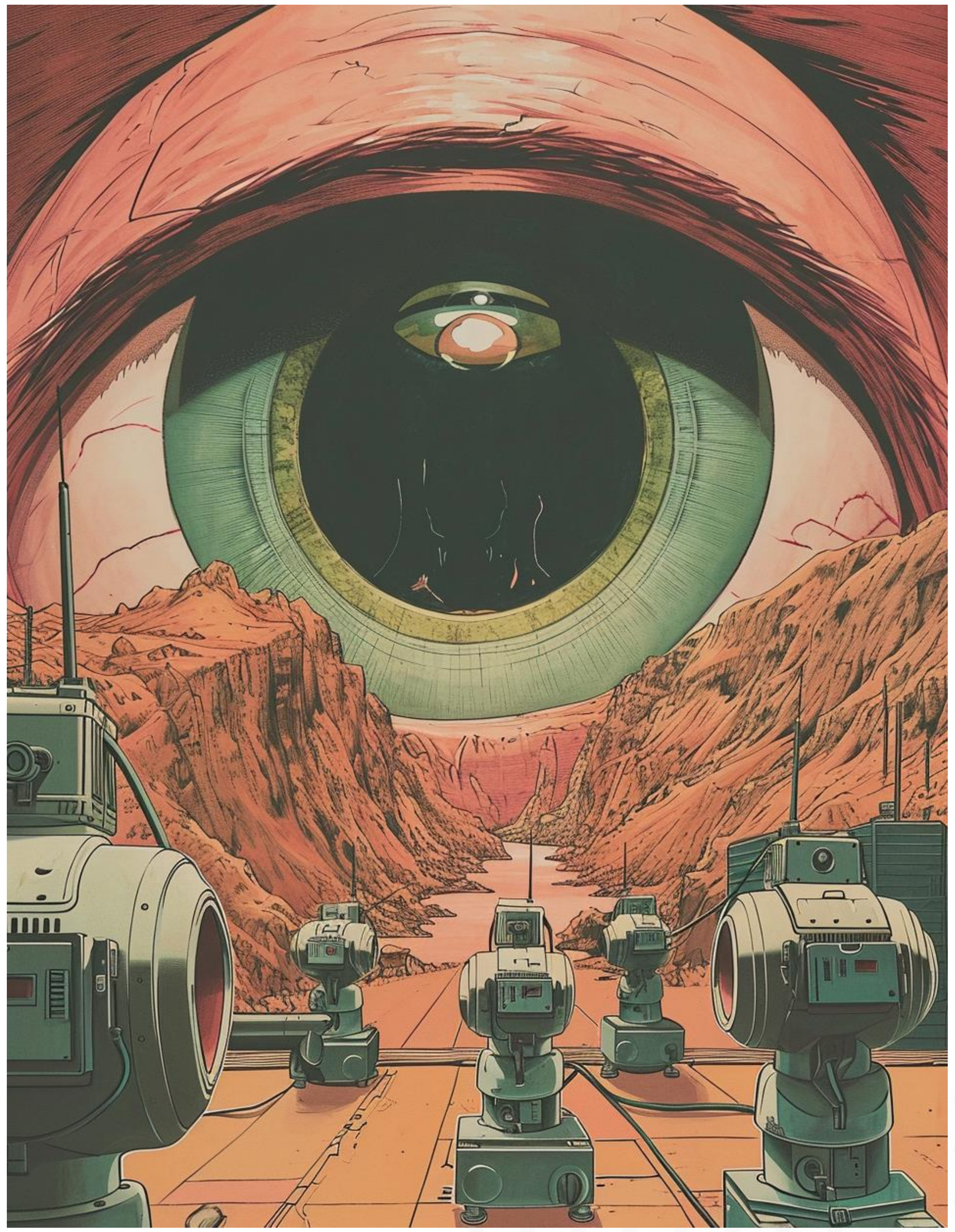
Our Client JavaScript App

The Three Observers

Things we do with the data

Currently, they're
all tangled up together

The Base Observer



The base observer is very simple

receiveNotification

It will receive a dataPacket (at first)

Three observers:

- ChartObserver
- TableObserver
- ConsoleObserver

Adding a new one later on is very easy

We are *refactoring*

Moving from one working system to another

Demo: Migrating Our Client JS to TypeScript

Review the application

Review DataPacket

Create an abstract base observer class

Begin implementing it with the ChartObserver class

Demo: Migrating Our Client JS to TypeScript, Part II

Finish up our Chart Observer

Create our Dispatcher class

Dispatch our websocket packets to our observers

Skip ahead to a number of solved problems

Review the solutions

Look at how the other code is implemented as Observers

Module Resolution Problems

Wow – this is really simple!

We can struggle with this, or just use webpack

Two Primary Problems

The name resolution strategies for JavaScript and NodeJs are in conflict

***node_modules* doesn't work for client-side**

I Have Not Yet Begun to Refactor

I would continue with ChartObserver and make it more fluent

And more testable

Issue a known set of packets and check the svg against a known set

Could we have just done all of this in JavaScript? Yes...

But TypeScript represents all these ideas more clearly and directly

Directions for Further Research

Creating and Using Decorators in JavaScript

Ivan Mushketyk

Mixins

A way of combining classes

Human + Robot == Cyborg

Gets around TypeScript not allowing multiple inheritance

Iterators and Generators

Allow your element to take part in *for in* and *for of* loops

Makes code more comprehensible and easier to test

Generators are the other half

Course Summary

Version controlling your work

Debugging TypeScript

The TypeScript type system

Unit testing TypeScript

Generics and Interfaces

Driving our compilation with the tsconfig file

A long refactoring of a client app and the server that serves it

Our GitHub Repo

https://github.com/FeynmanFan/pstypescriptcc

Thank you very much for watching!!!