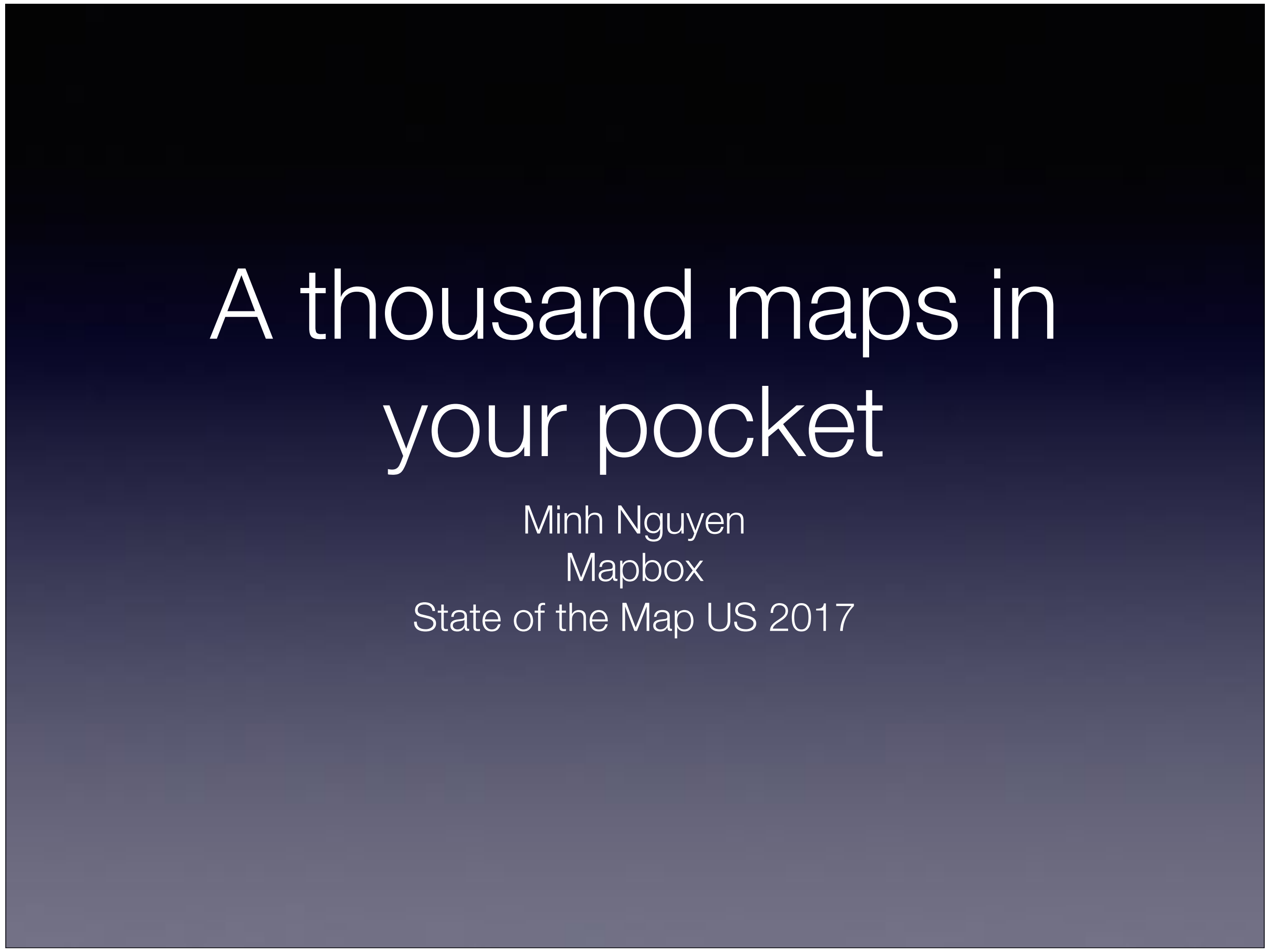
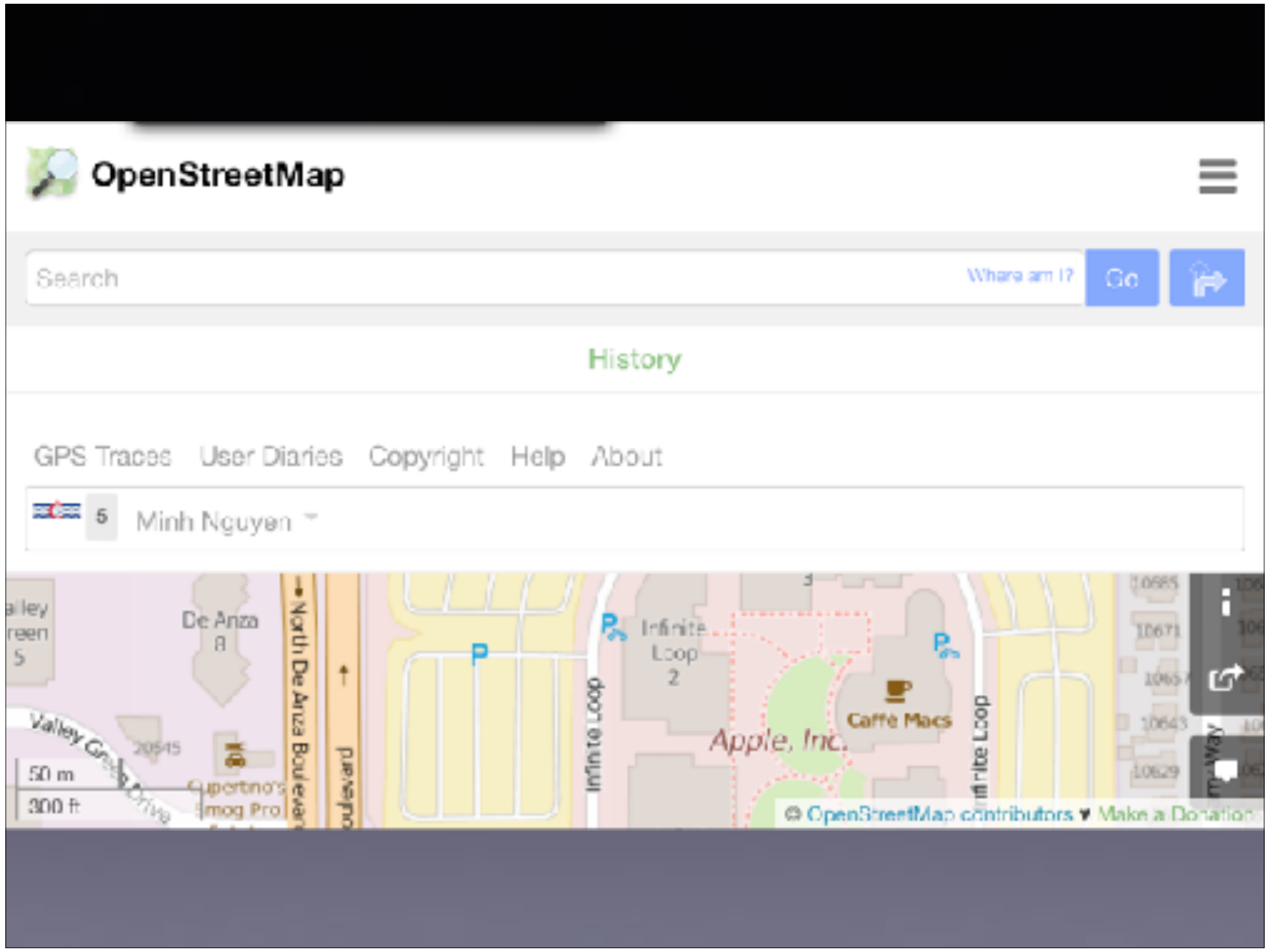


A thousand maps in your pocket

Minh Nguyen
Mapbox
State of the Map US 2017

Hi, I’m Minh Nguyen. If you saw my lightning talk yesterday, you might think I spend my whole day leafing through phone books. But I also work at Mapbox on map and navigation libraries for iOS applications.



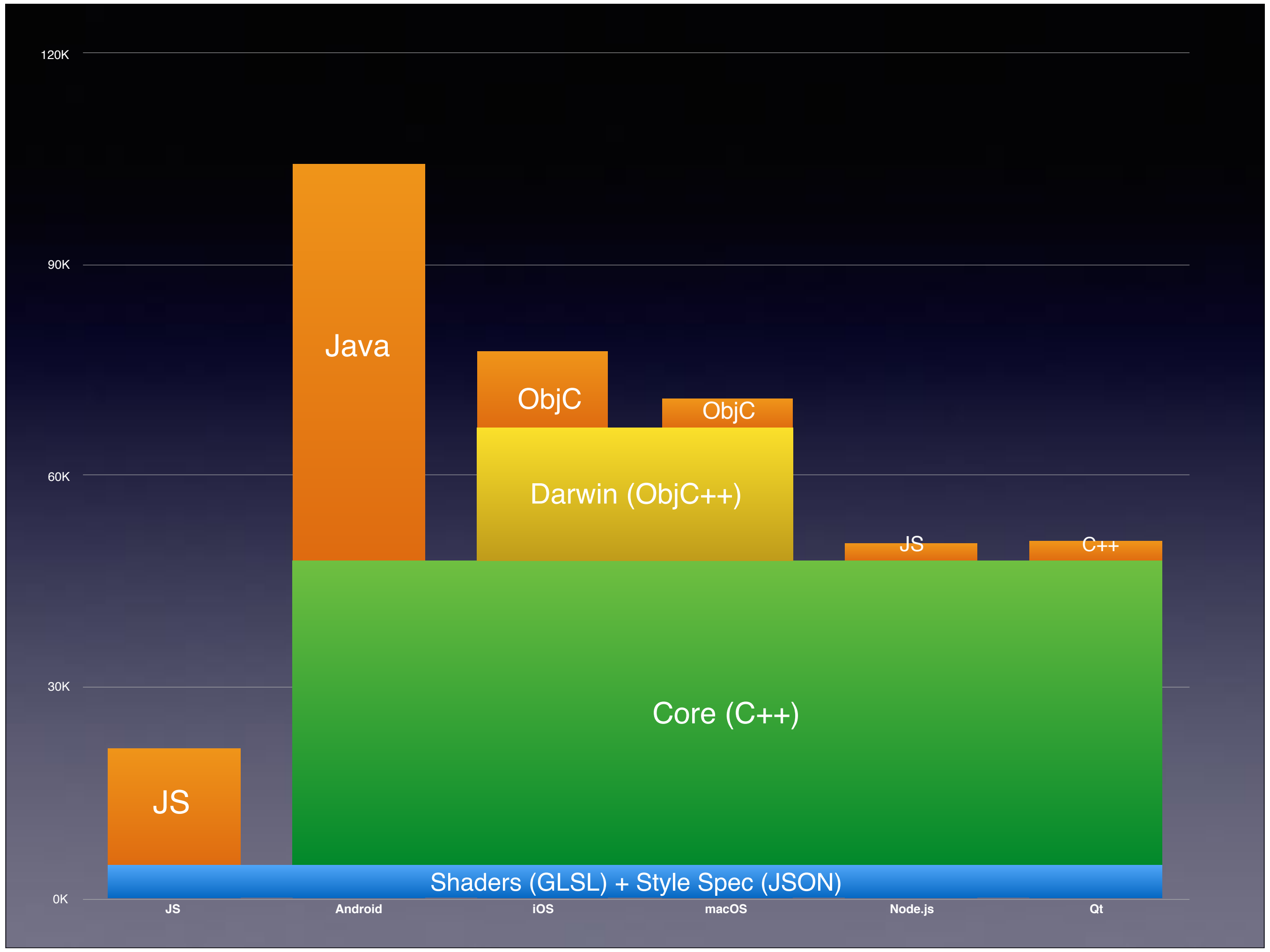
Think about the technologies you’ve heard about during this conference. How many of them work well on a phone or tablet? This is a weak spot for the OSM ecosystem.

The Phone Comes First

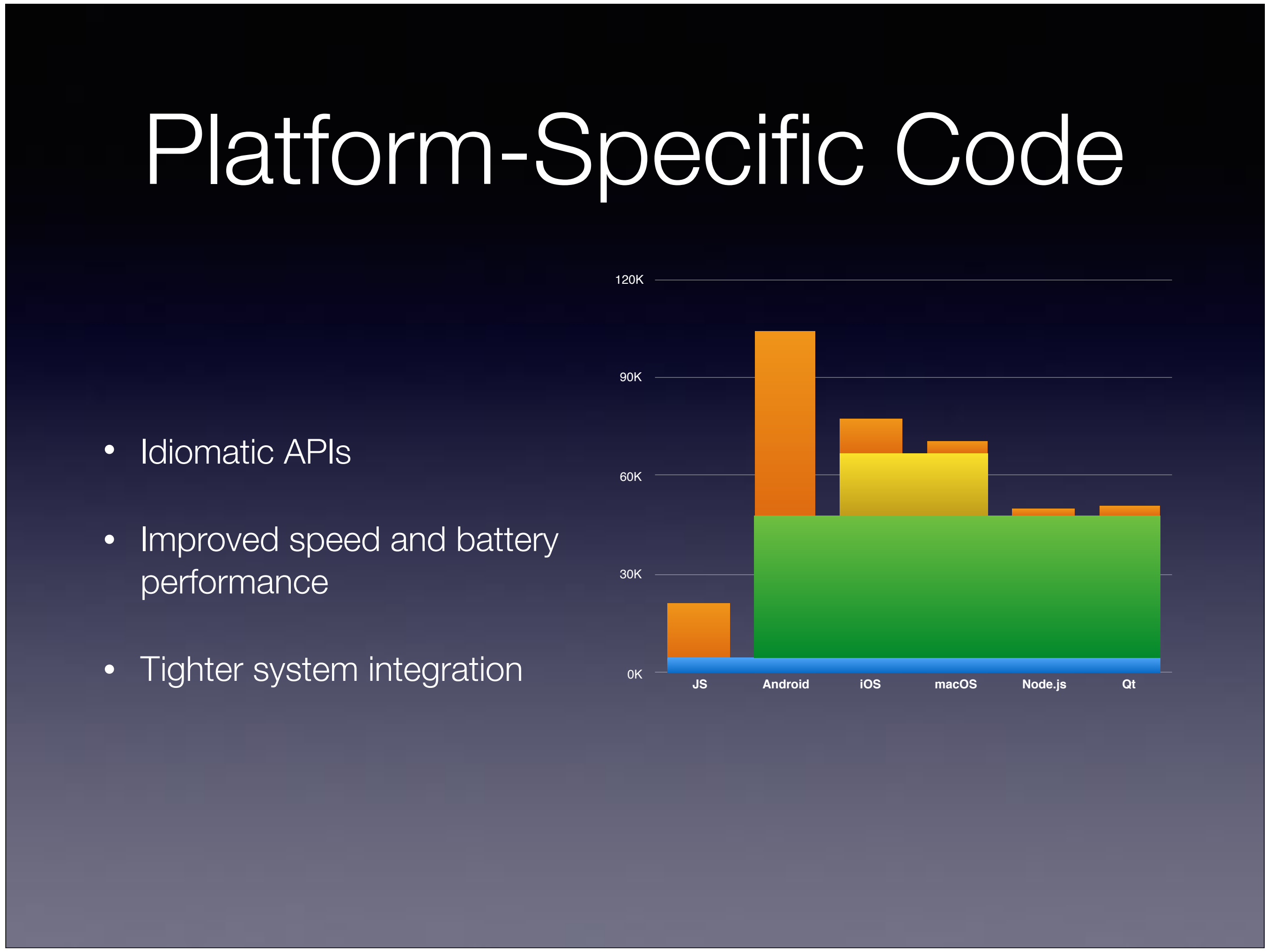
Why is it so important to focus on maps on mobile devices? Because people reach for their phones before their computers, especially when it comes to the most basic use case for maps – navigation. A map that doesn’t feel right at home on a phone might as well not exist.

Decisions, Decisions

A mobile developer faces a constellation of choices. They choose which platforms to target, which languages to code in, whether to go native or hybrid, which tools to manage library dependencies, whether to use design tools or do everything in code, etc. At Mapbox, we want to see OSM used widely, so it's on us to support as many of these combinations as possible, because a developer will've already decided on all these things long before they choose to put a map inside their application.



All those combinations mean lots of different languages, from JavaScript to Java to Objective-C, but we share a lot of code between each platform to keep them all in lockstep in terms of how they download and render the map data – all the while ensuring that our map libraries are a first-class citizen on each platform.



Platform-Specific Code

- Idiomatic APIs
- Improved speed and battery performance
- Tighter system integration

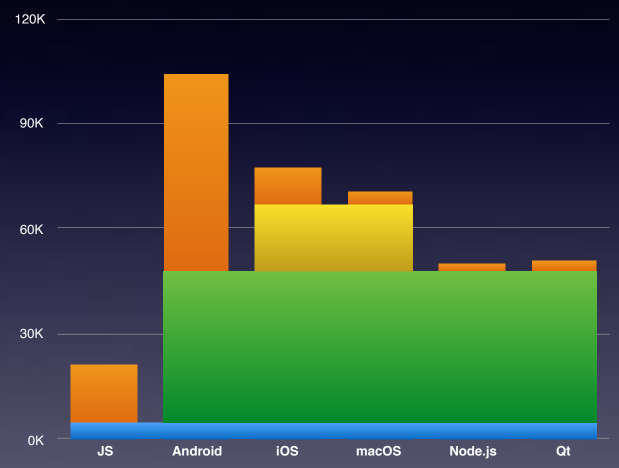
So all those orange parts sitting at the top of the stack – what’s all that platform-specific code doing? Platform-specific code puts a pretty face on the raw rendering engine – idiomatic APIs that, say, an iOS developer who knows nothing about maps in particular can still find intuitive. It also integrates much more tightly with the system than any library running in a Web browser can, resulting in better speed and battery performance, more natural and responsive gestures, and compatibility with system features like drag-and-drop and GPS tracking.

Zero

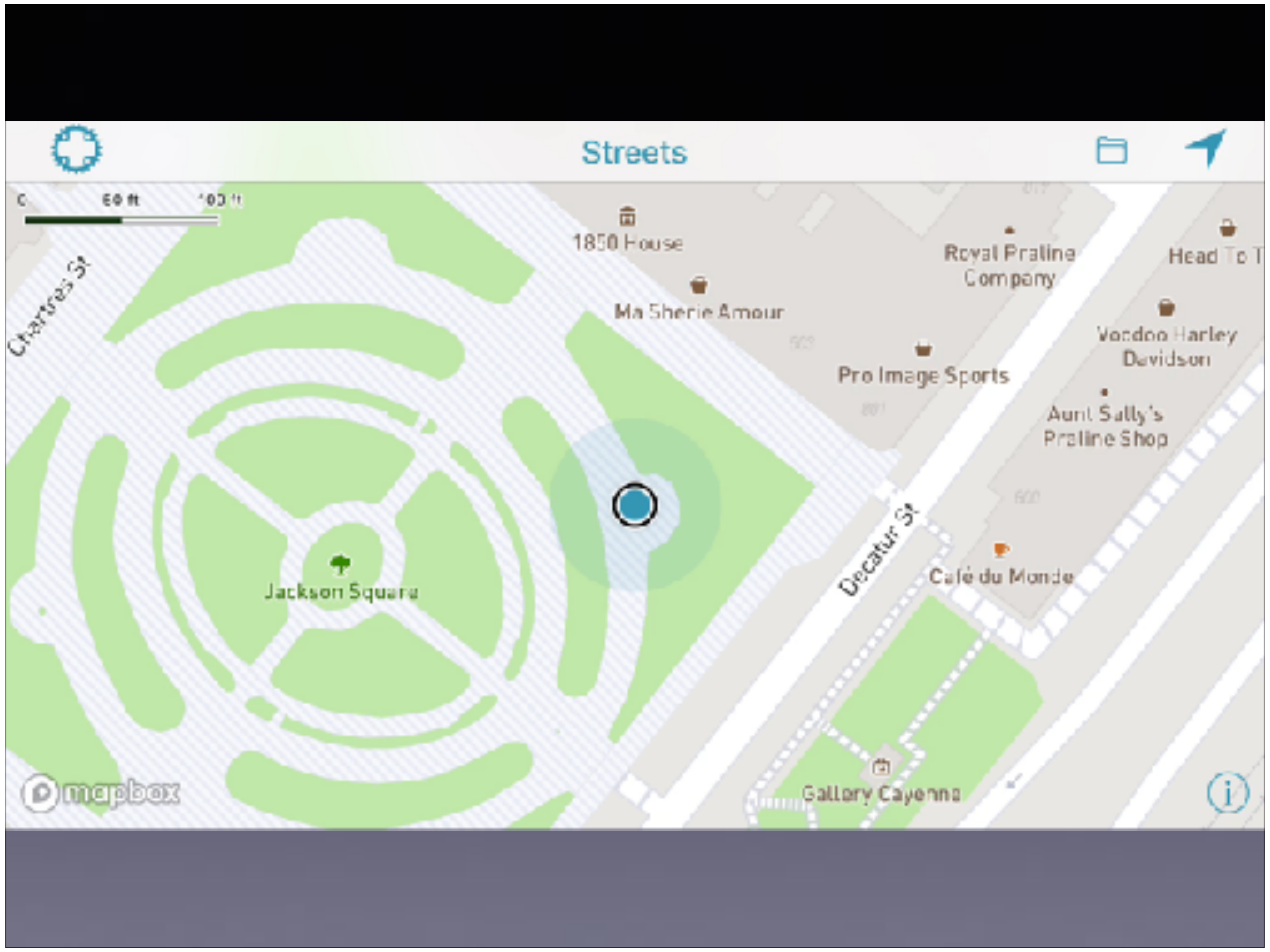
At Mapbox, we took the approach of writing a map rendering engine from scratch. But here’s what we’re up against: zero. That’s how much it costs for an iOS developer to use Apple Maps in their application. That’s also roughly how much Apple Maps adds to the size of their application, because it comes preinstalled with iOS. It’s a similar story on Android with the Google Maps SDK.

Client-Side Vector Tiles

- Enables customizability
- Offline maps
- Accessibility

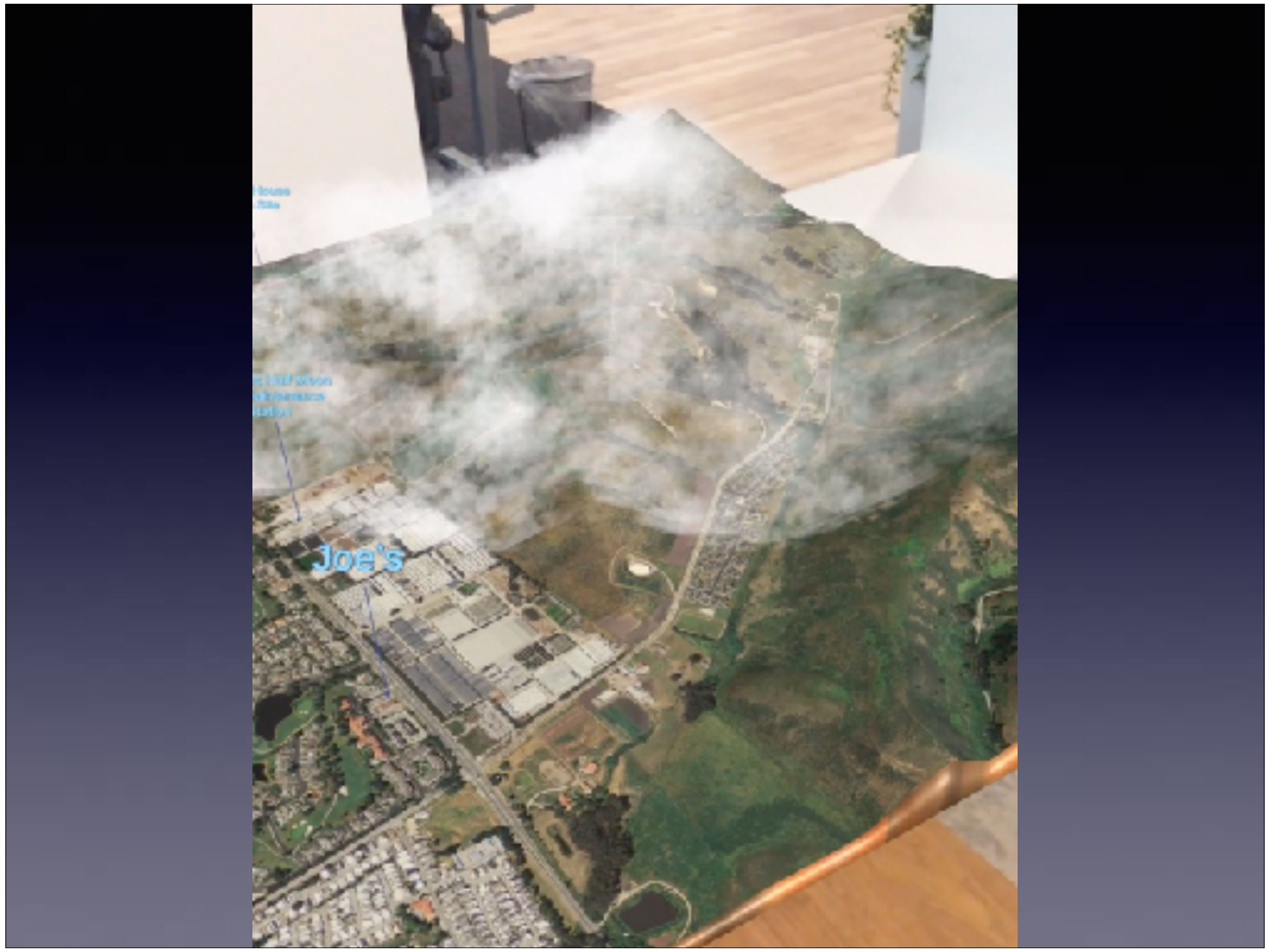


Third-party, OSM-powered libraries can’t necessarily compete with the incumbents on price or size, but we can focus on features like offline maps and customizability. Let’s say you’ve got a mobile application that people take out into the mountains back there where a cell connection can be a luxury. People need to download the maps offline so they can depend on them. And let’s say the app needs to switch to a trail-centric map when the user’s on the trail and a darker map after the sun sets. Instead of downloading three different sets of raster tiles, the app can download the vector tile data just once then apply three different styles effortlessly.



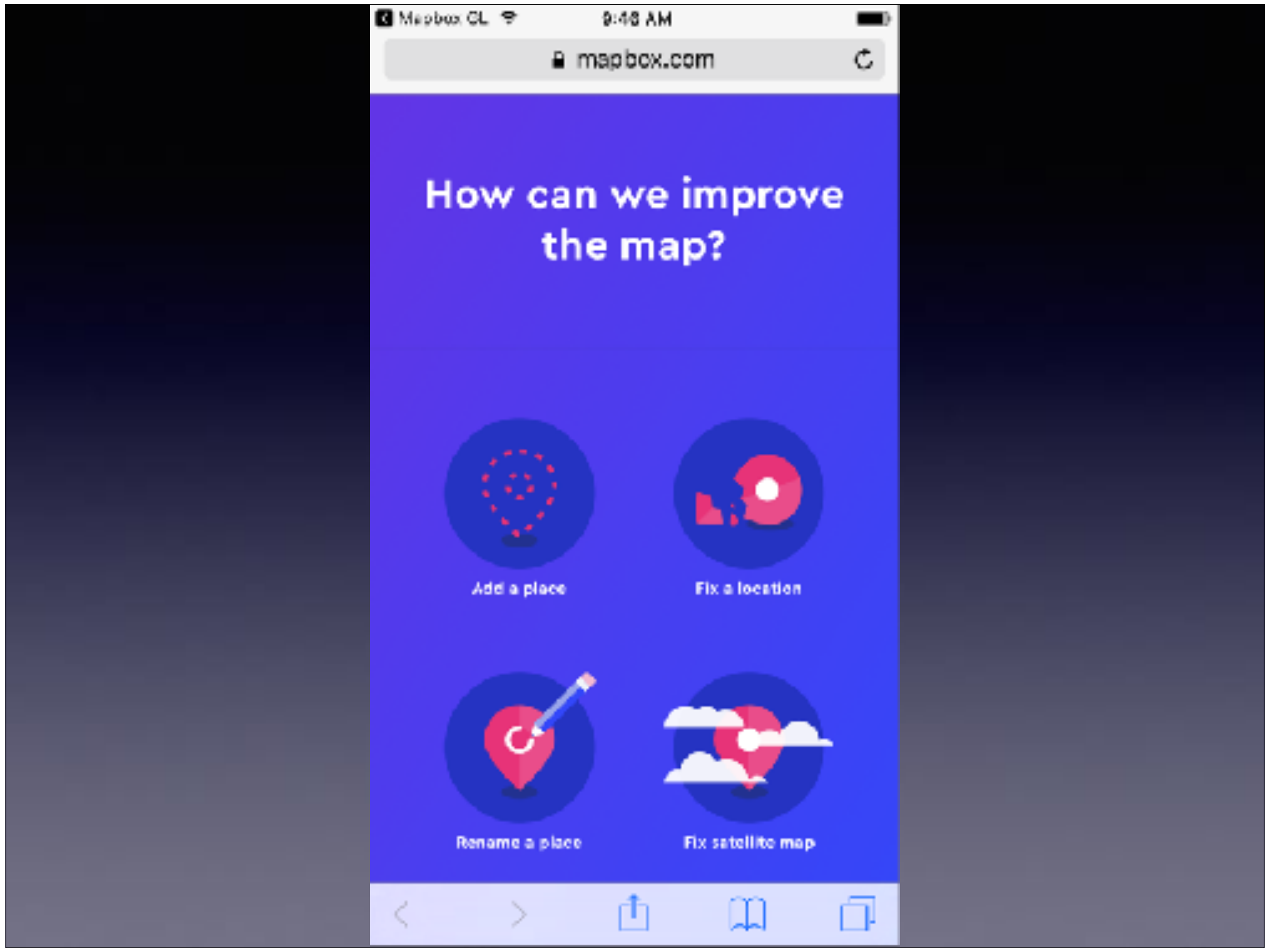
It turns out that same vector data can be used for nontraditional purposes too. Let's say a user is blind or visually impaired. What good is a raster map tile? It's just a picture. But vector data on the mobile device itself can integrate with the system's built-in screen reader. Labels and roads are read aloud, making the same information accessible to sighted and blind users alike.

Map © Mapbox / Mapbox GL © Mapbox, see license / map data © OpenStreetMap contributors, ODbL



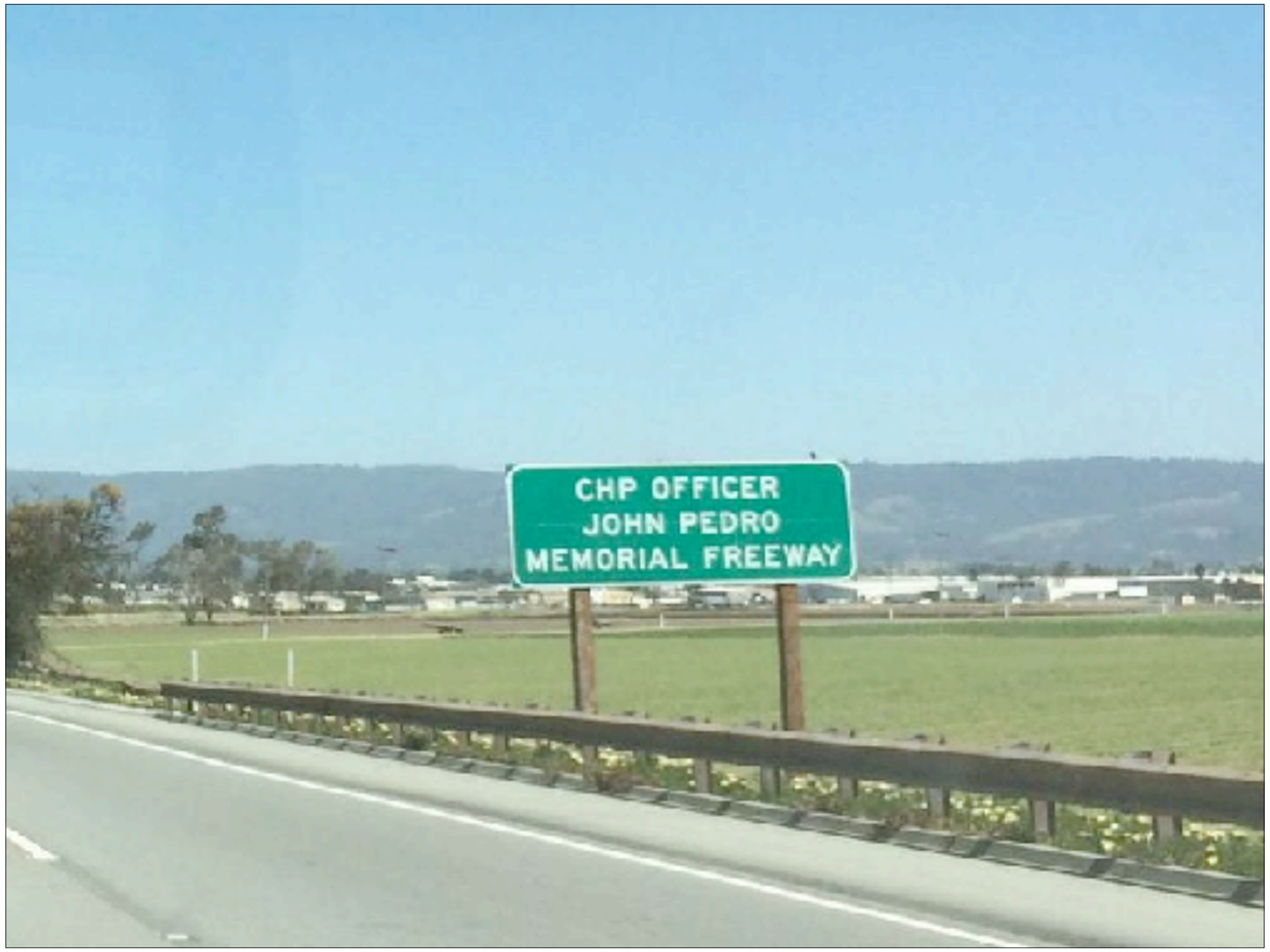
And now OSM data is accessible to mobile game developers too via our Unity SDK and augmented reality frameworks like ARKit. This 3D terrain map sitting on a dinner table is sort of the bread and butter of augmented reality right now, and a great many of the projects you’re seeing are based on OSM data – because it’s open.

Image © Mapbox / map data © OpenStreetMap contributors, ODbL



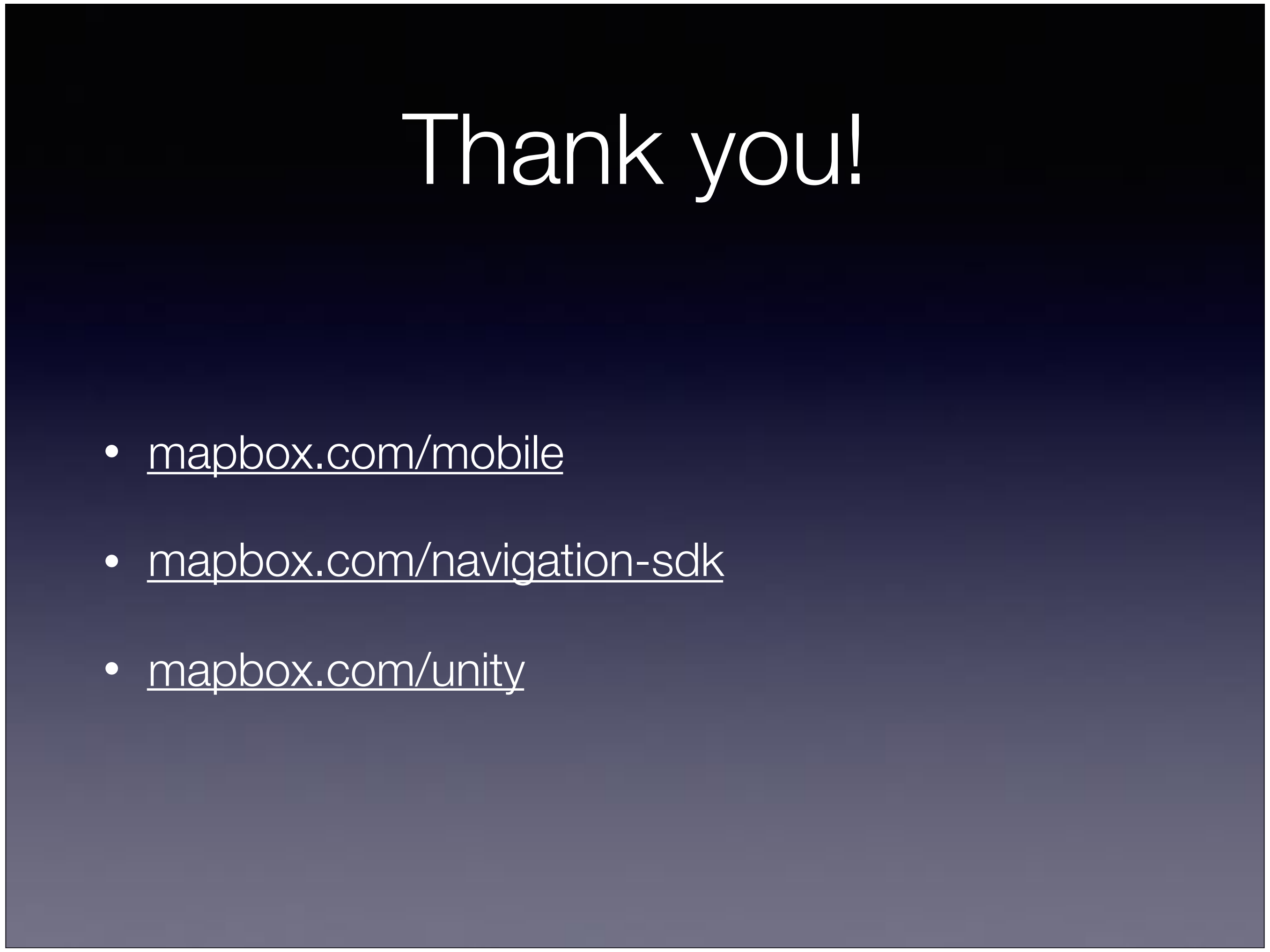
How can we improve the map?

There’s a whole lot that I’m unable to cover in this lightning talk, because open mobile maps are such an unexplored area. For example, how does someone casually contribute to the map? There are mobile editors, but we’re getting the map in front of a much wider audience who may not be prepared to edit fruitfully on a mobile device, with all its UI constraints. For that reason, Mapbox maps come with a mobile-friendly feedback tool as a backstop.



And when OSM data is placed inside turn-by-turn navigation applications, suddenly there are new considerations, like keeping instructions brief so drivers can focus on driving. Here’s a memorial designation that was applied to a stretch of California Highway 1 and originally tagged as the motorway’s name in OSM. Imagine hearing, “Take the ramp onto CHP Officer John Pedro Memorial Freeway”. No offense to officer Pedro, but you can imagine how a mobile app might need to prioritize other information over these unwieldy names.

Photo by kaflurbaleen / CC BY-SA



Thank you!

- mapbox.com/mobile
- mapbox.com/navigation-sdk
- mapbox.com/unity

That’s all the time I have for today, but check out Mapbox’s mobile offerings on our website, and visit our booth any time this weekend to try out our Unity and screen reader demos.