Dev Days

#XamarinDevDays

Use offline sync to build responsive apps with Azure Mobile Apps

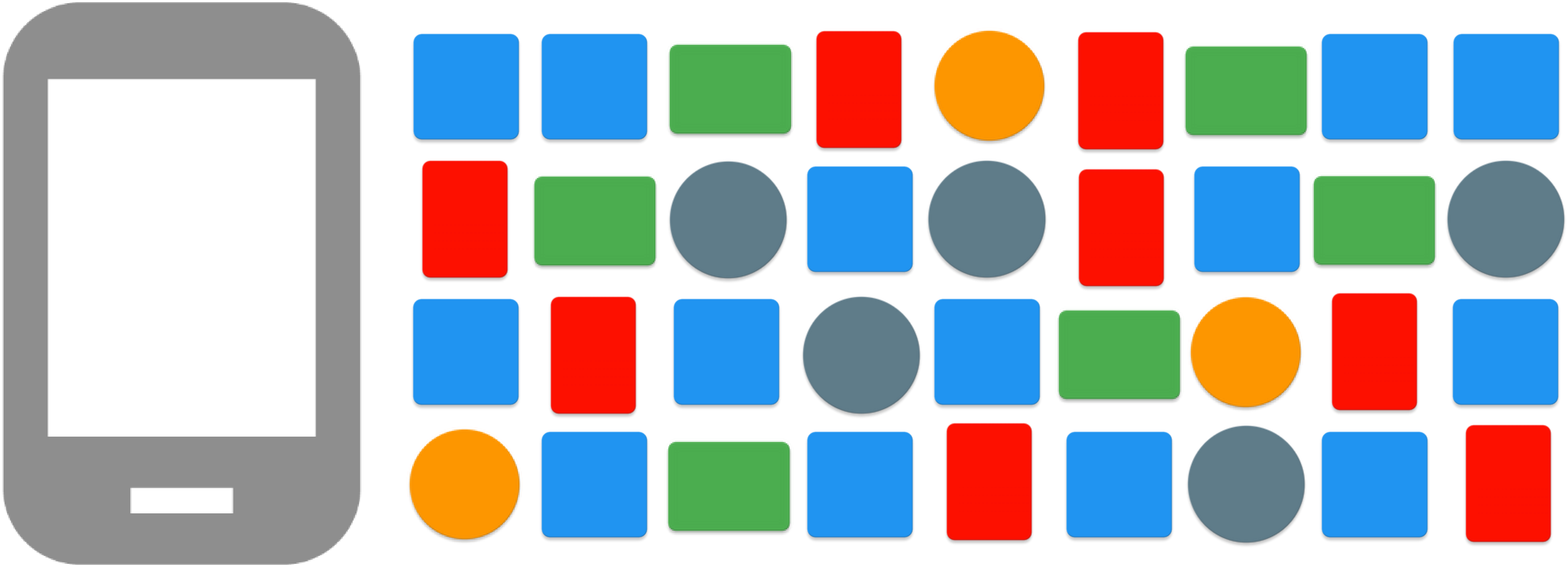
We ❤ Apps!

189M
downloads
a day

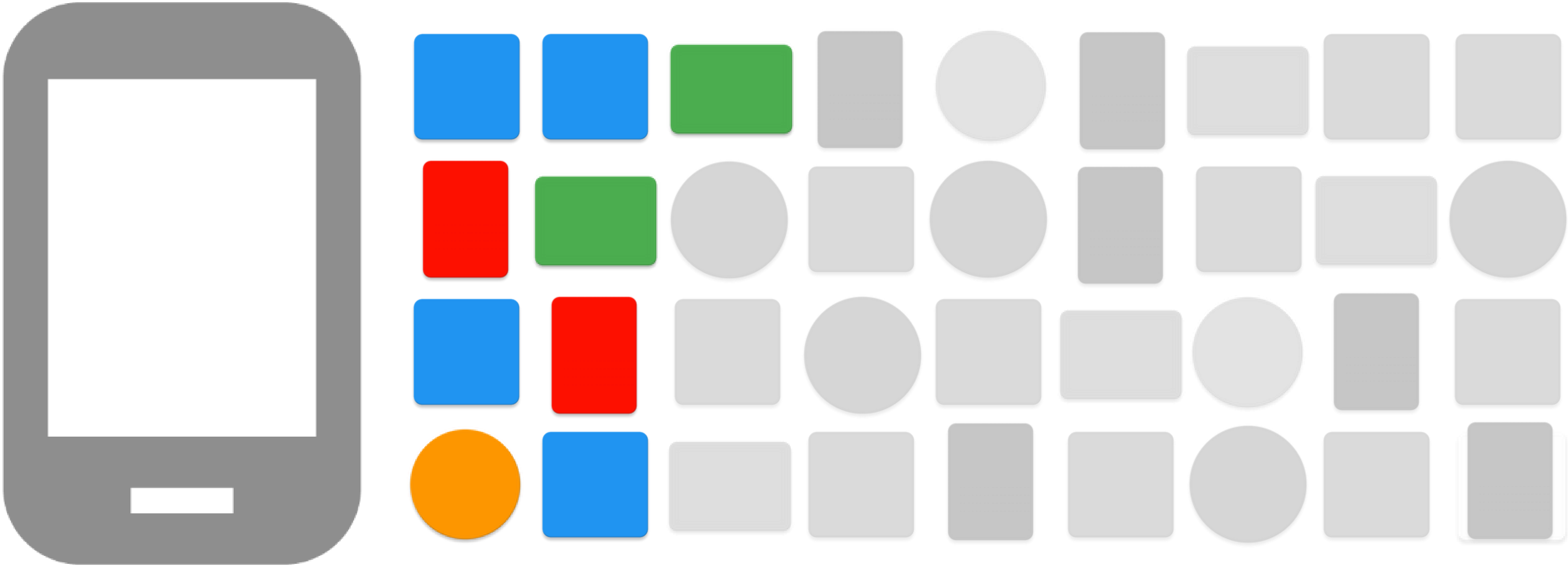
200
mins on
phone

127
mins in
apps

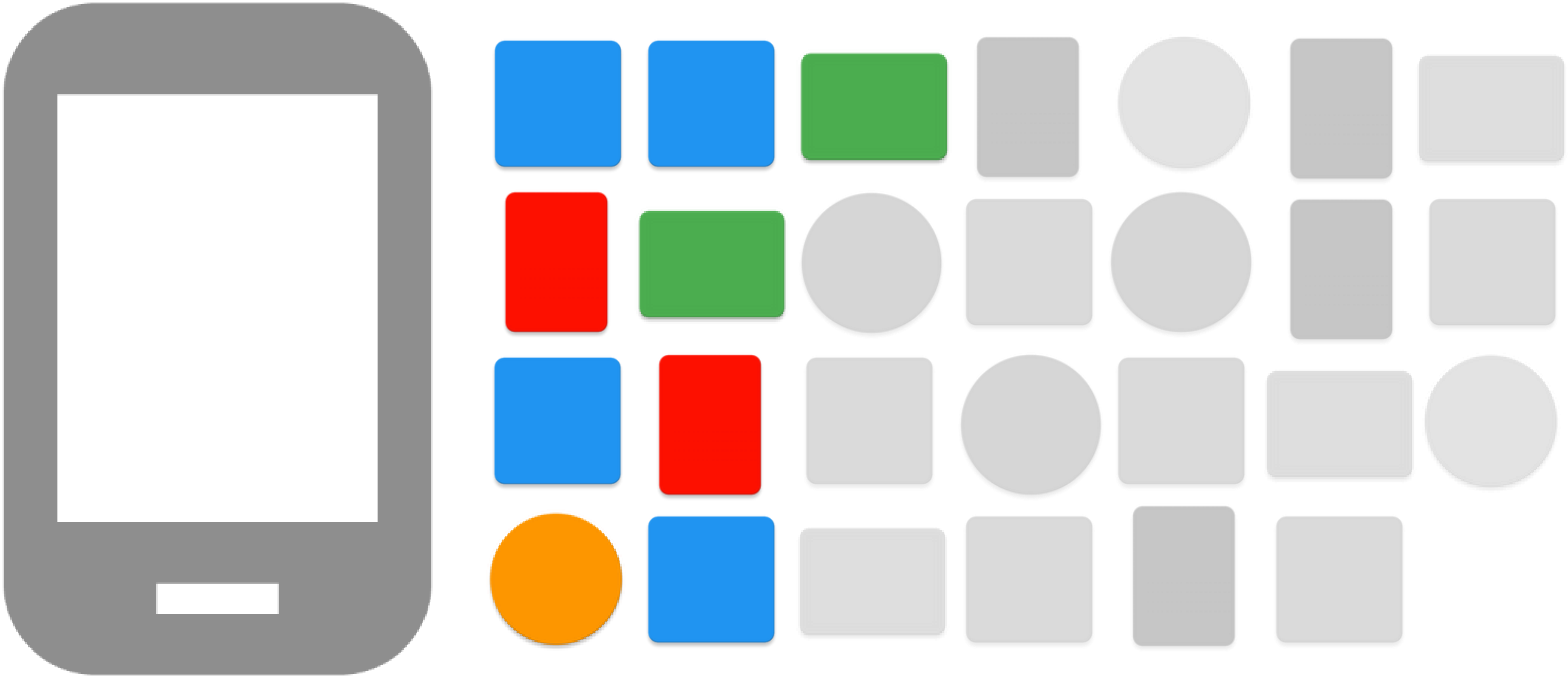
The average app user has **36** apps installed on his or her phone.



Only 1/4 are **used daily**:



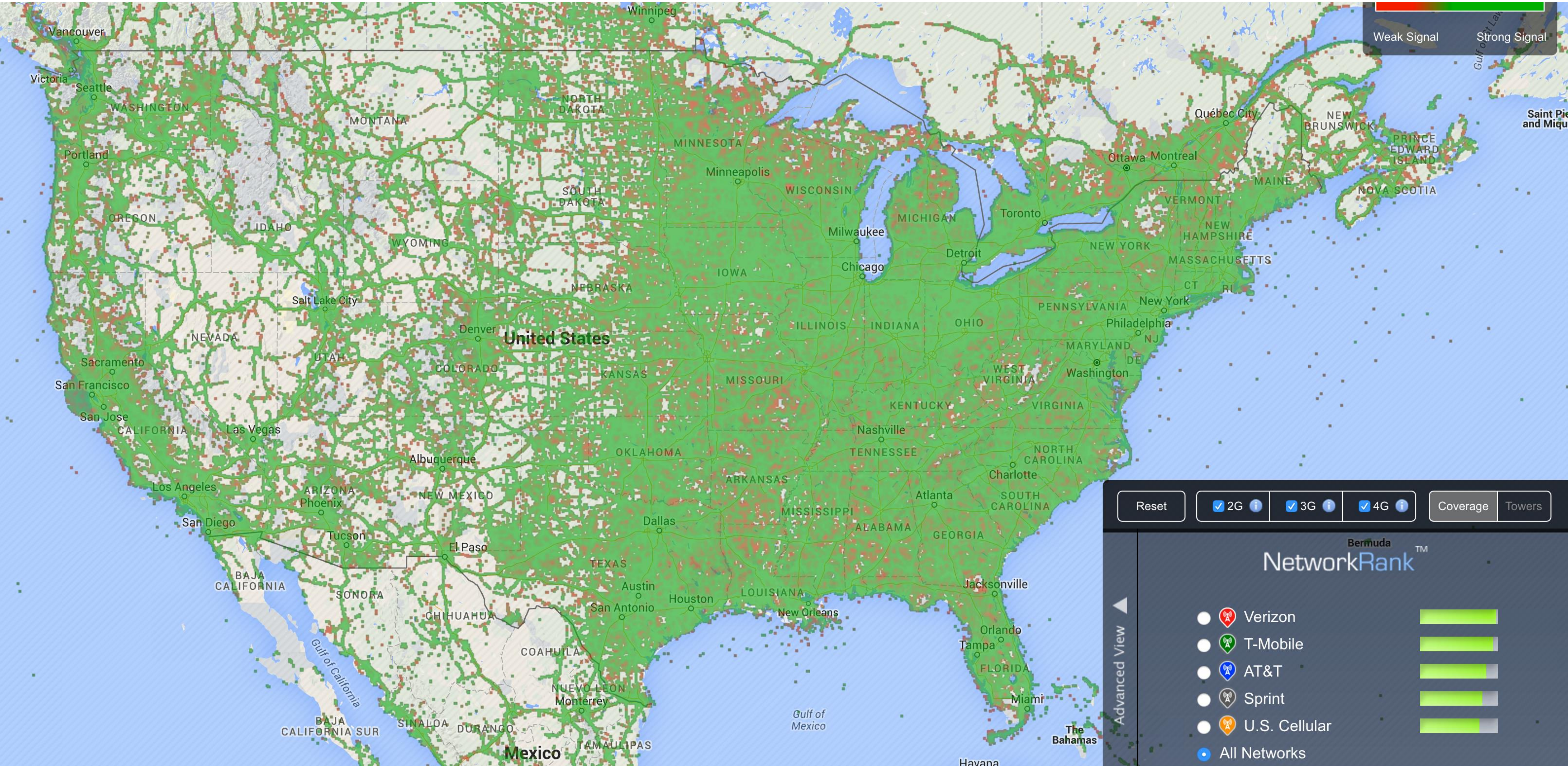
1/4 of apps are **never used**!



Bad App Experiences

- Slow or laggy experience
- Crashes
- Not intuitive & bad user experience
- Features not as advertised
- Data not available when you need it

Always connected?



Reset

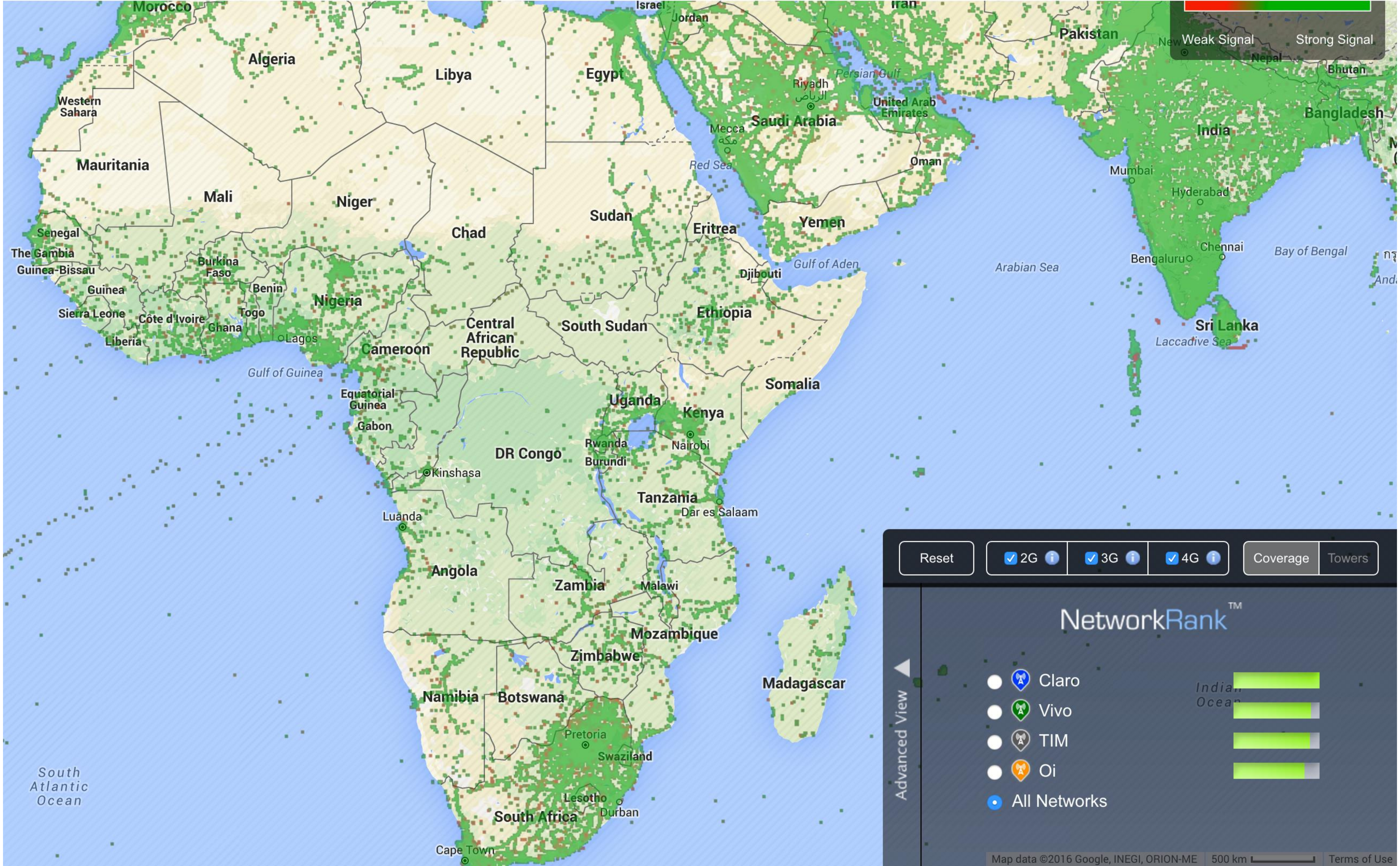
☑ 2G ☑ 3G ☑ 4G

Coverage | Towers

Bermuda NetworkRank™

○ Verizon
○ T-Mobile
○ AT&T
○ Sprint
○ U.S. Cellular
● All Networks

http://opensignal.com/coverage-maps



Reset

☑ 2G ☑ 3G ☑ 4G

Coverage | Towers

NetworkRank™

○ Claro

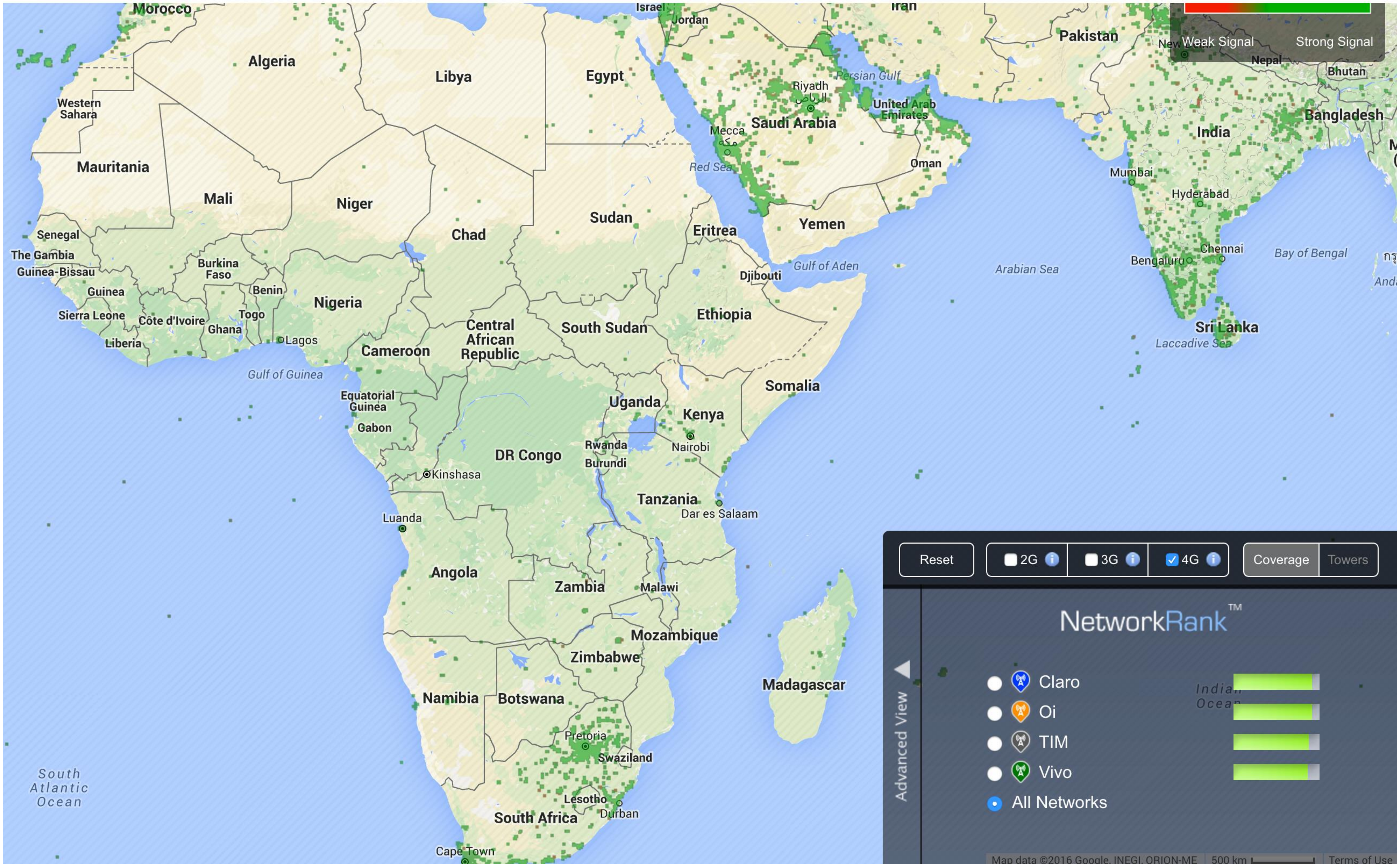
○ Vivo

○ TIM

○ Oi

● All Networks

Map data ©2016 Google, INEGI, ORION-ME 500 km Terms of Use



☐ 2G ☐ 3G ☑ 4G

NetworkRank™

○ Claro
○ Oi
○ TIM
○ Vivo
● All Networks

Map data ©2016 Google, INEGI, ORION-ME 500 km Terms of Use

What about a backend?

Plenty of Options

Azure Mobile Apps

Amazon Web Services

Couchbase

Oracle Mobile Cloud

IBM MobileFirst

SQLClipher

Realm

SQLite-net

Infrastructure designed for Scale

100+ datacenters

Top 3 networks in the world

2x AWS, 6x Google DC Regions

G Series – Largest VM in World
32 cores, 448GB Ram, SSD…

● Operational

● Announced

* Operated by 21Vianet

24

Azure compute regions open today

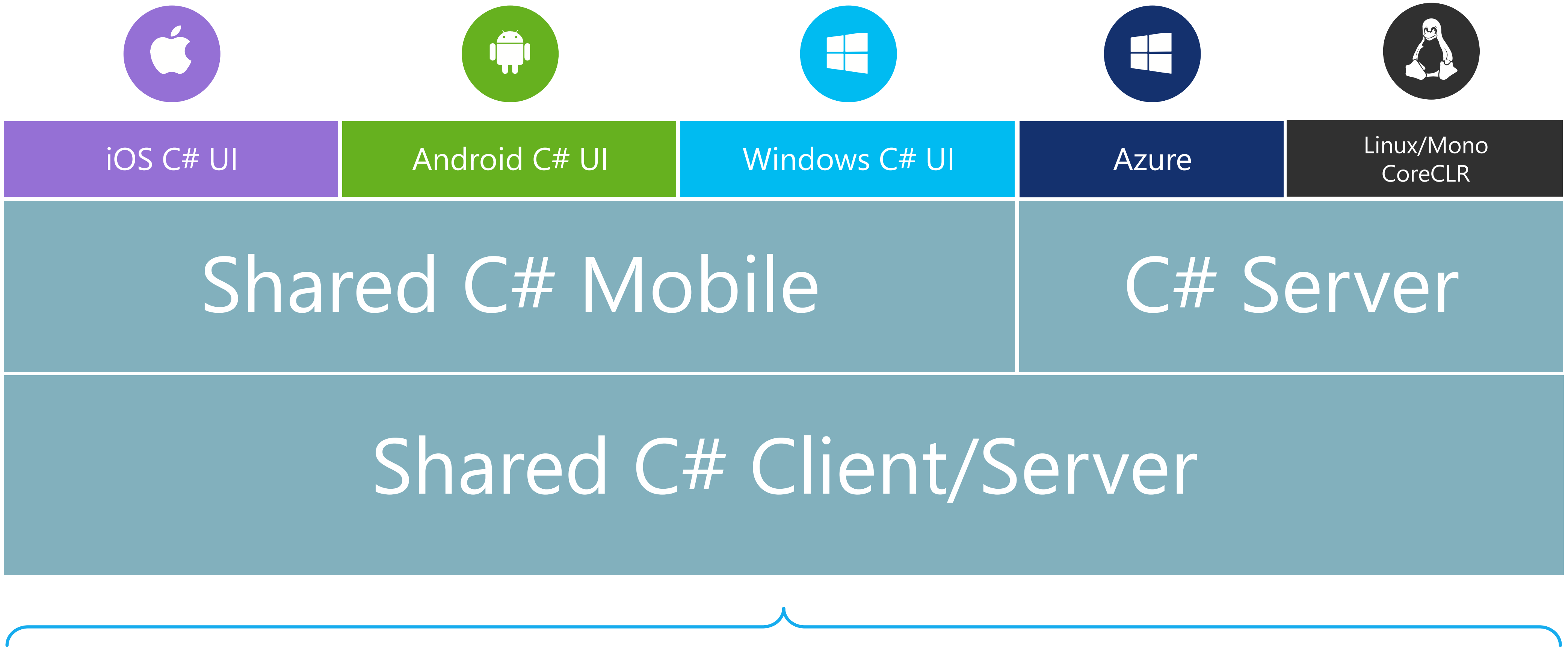
More than AWS and Google Cloud combined

Datacenters recently added in Canada and India

Why Azure?

- *Extremely* powerful
- Flexible
  - Easy Tables
  - App Service
- C# SDKs available everywhere:
  - C#- iOS, Android, & Windows with Xamarin
  - C# clients, written by C# developers (open source)
  - C# backend with ASP.NET

Xamarin Apps + Backend Services



iOS C# UI

Android C# UI

Windows C# UI

Azure

Linux/Mono
CoreCLR

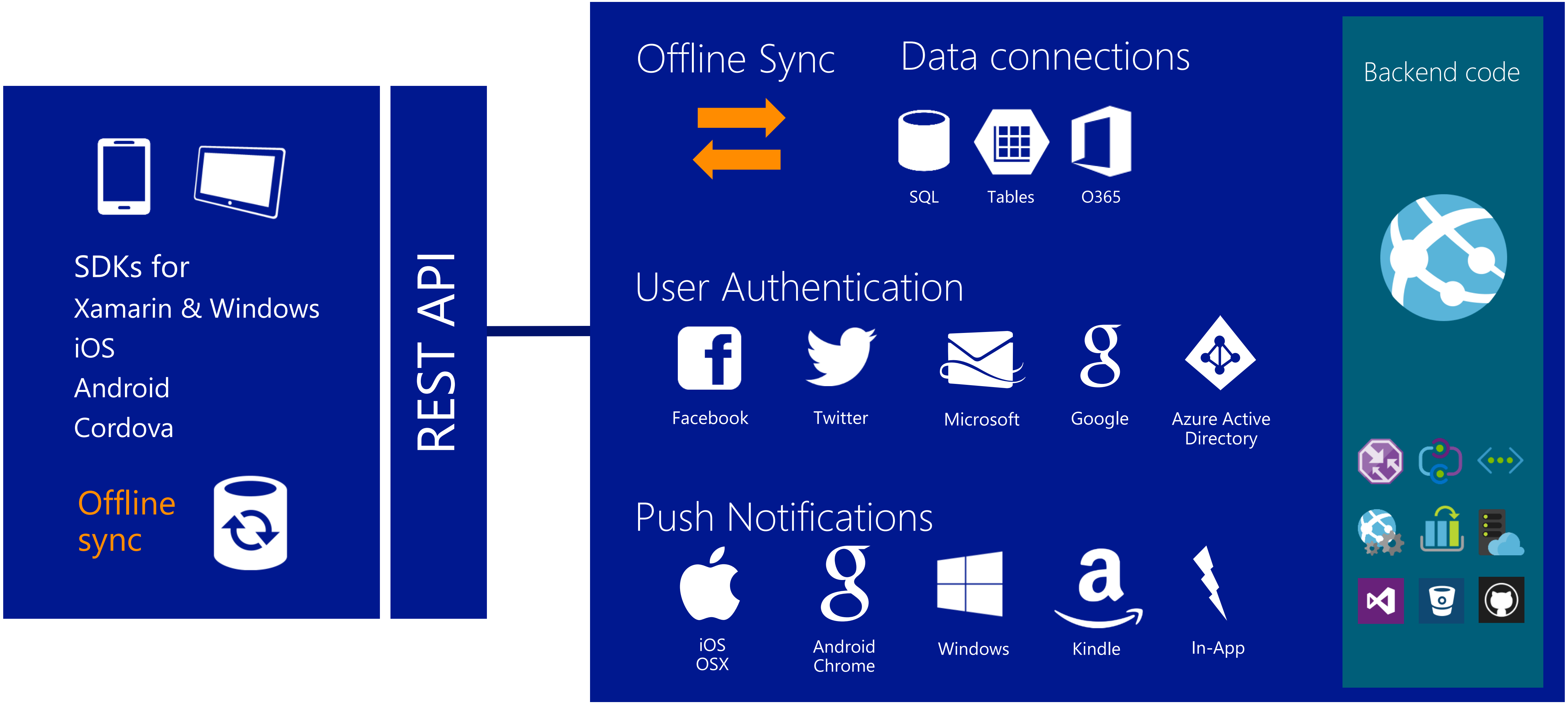
Shared C# Mobile

C# Server

Shared C# Client/Server

Shared C# codebase • 100% native API access • High performance

Azure Mobile Apps



SDKs for
Xamarin & Windows
iOS
Android
Cordova

Offline sync

REST API

Offline Sync

Data connections

SQL
Tables
O365

User Authentication

Facebook
Twitter
Microsoft
Google
Azure Active Directory

Push Notifications

iOS
OSX

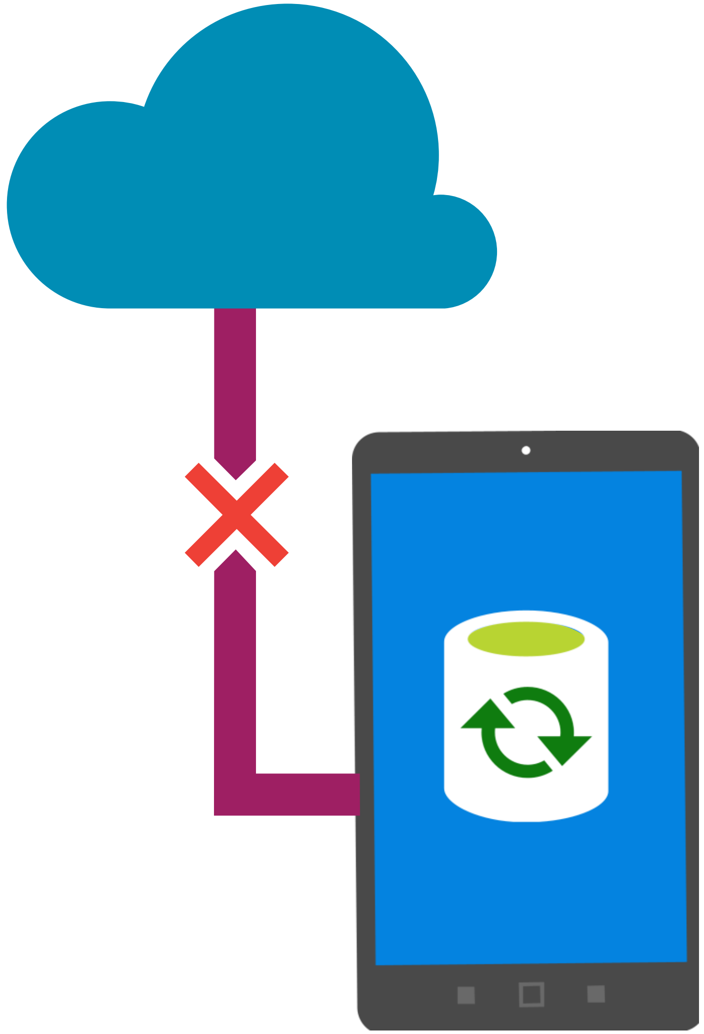
Android
Chrome

Windows
Kindle
In-App

Backend code

Mobile Offline Sync

- Make apps resilient against intermittent network connectivity
- Allow end-users to create and modify data even when there is no network access
- Sync data across multiple devices
- Improve app responsiveness by caching server data locally on the device
- Detect and handle conflicts when the same record is modified by more than one client



Create a Mobile Service

```
MobileService =  new MobileServiceClient(
    "https://myapp.azurewebsites.net");
```

Create Tables

```
IMobileServiceSyncTable<TodoItem> syncTable;

public async Task Init()
{
    const string path = "syncstore.db";
    var db = new MobileServiceSQLiteStore(path);
    db.DefineTable<TodoItem>();

    await MobileService.SyncContext.InitializeAsync(db);
    syncTable = MobileService.GetSyncTable<TodoItem>();
}
```

Push and pull with sync table

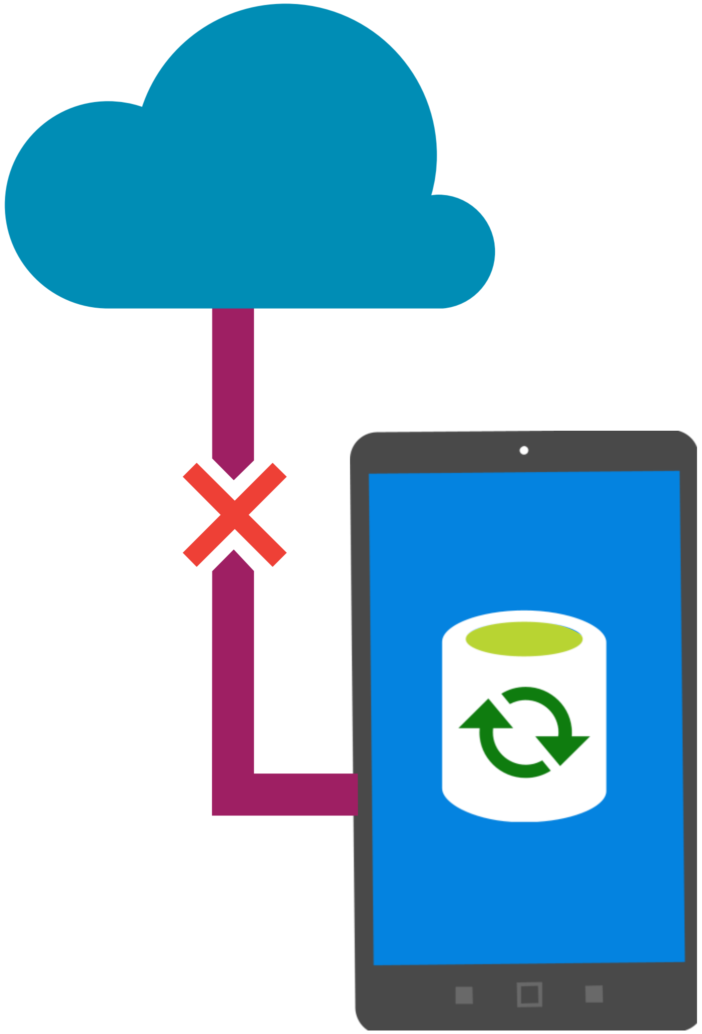
```
private async Task SyncAsync()
{
    await MobileService.SyncContext.PushAsync();
    var query = syncTable.CreateQuery();

    await syncTable.PullAsync("todoItems", query);
}

private async Task InsertTodoItem(TodoItem todoItem)
{
    await syncTable.InsertAsync(todoItem);
    await MobileService.SyncContext.PushAsync();
}
```

Query local table

```
public async Task<IEnumerable<TodoItem>> GetOpenItemsAsync()
{
    return await todoTable
                 .Where(item => item.Complete == false)
                 .ToEnumerableAsync();
}
```

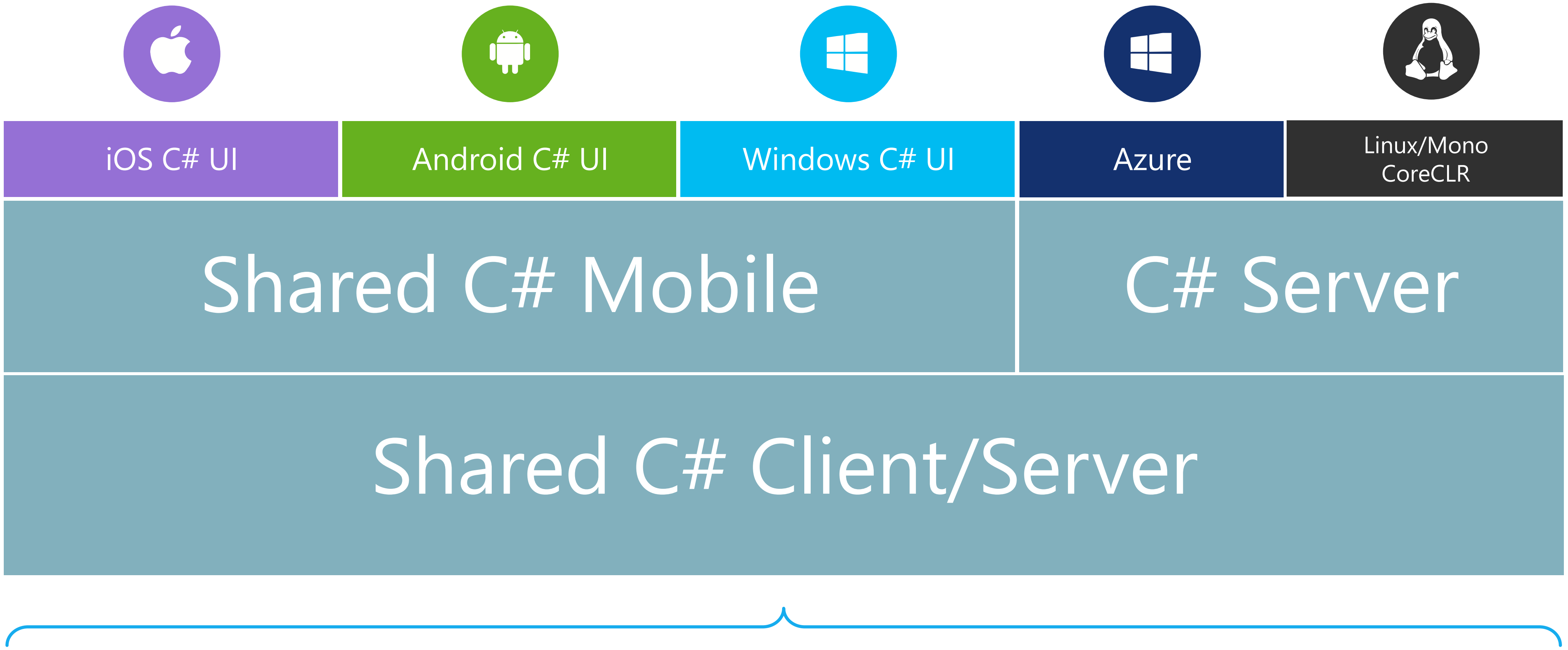
How offline sync works

- Access data from Mobile Apps tables even when app is offline
- Keep a local queue of Create, Update, Delete operations and synchronize with server when app is back online
- Detect conflicts when same item is changed both locally and on server
- Use soft delete to remove deleted records from client data stores
- Can use push notifications to trigger client sync



Let’s add a backend

Mobile + Server



iOS C# UI

Android C# UI

Windows C# UI

Azure

Linux/Mono
CoreCLR

Shared C# Mobile

C# Server

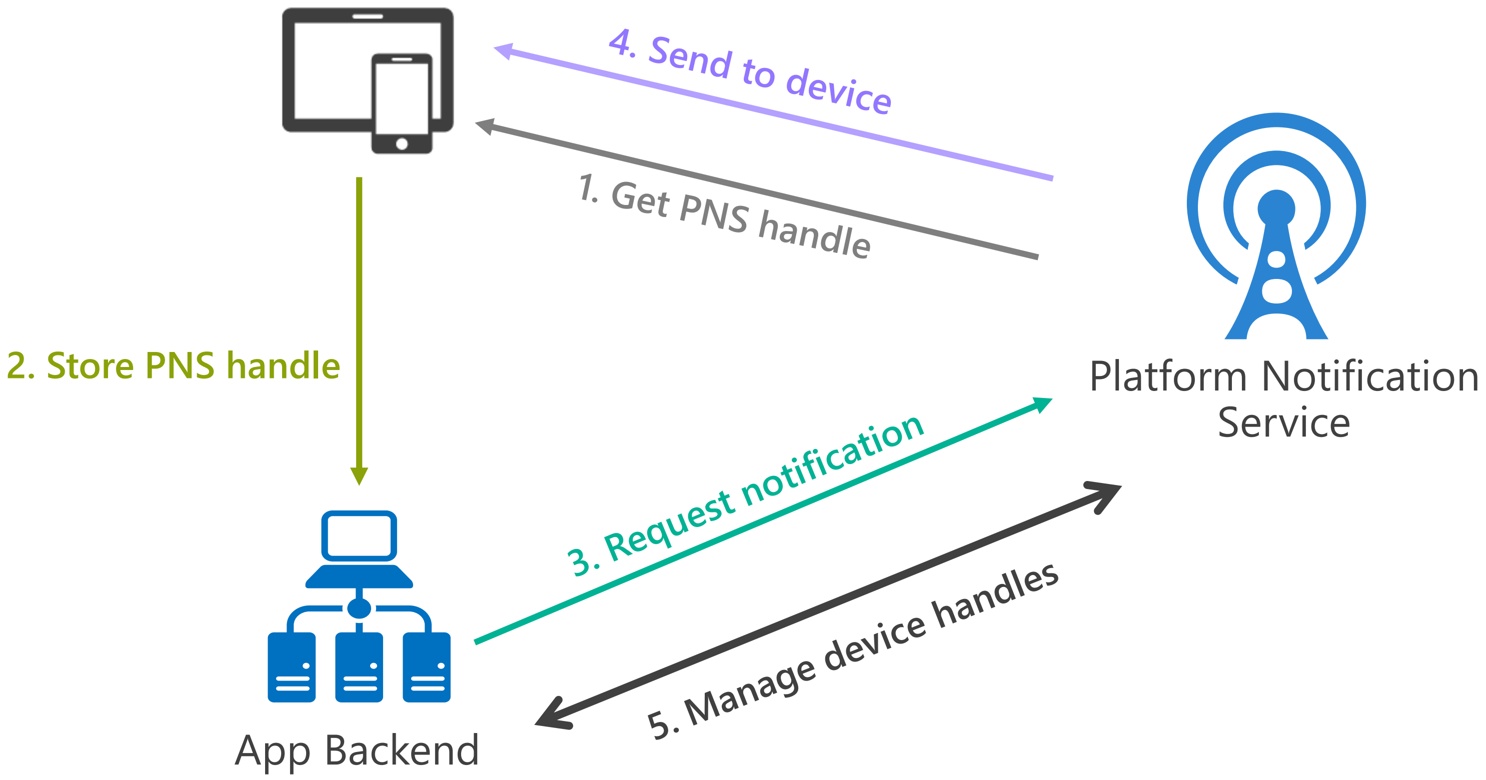
Shared C# Client/Server

Shared C# codebase • 100% native API access • High performance

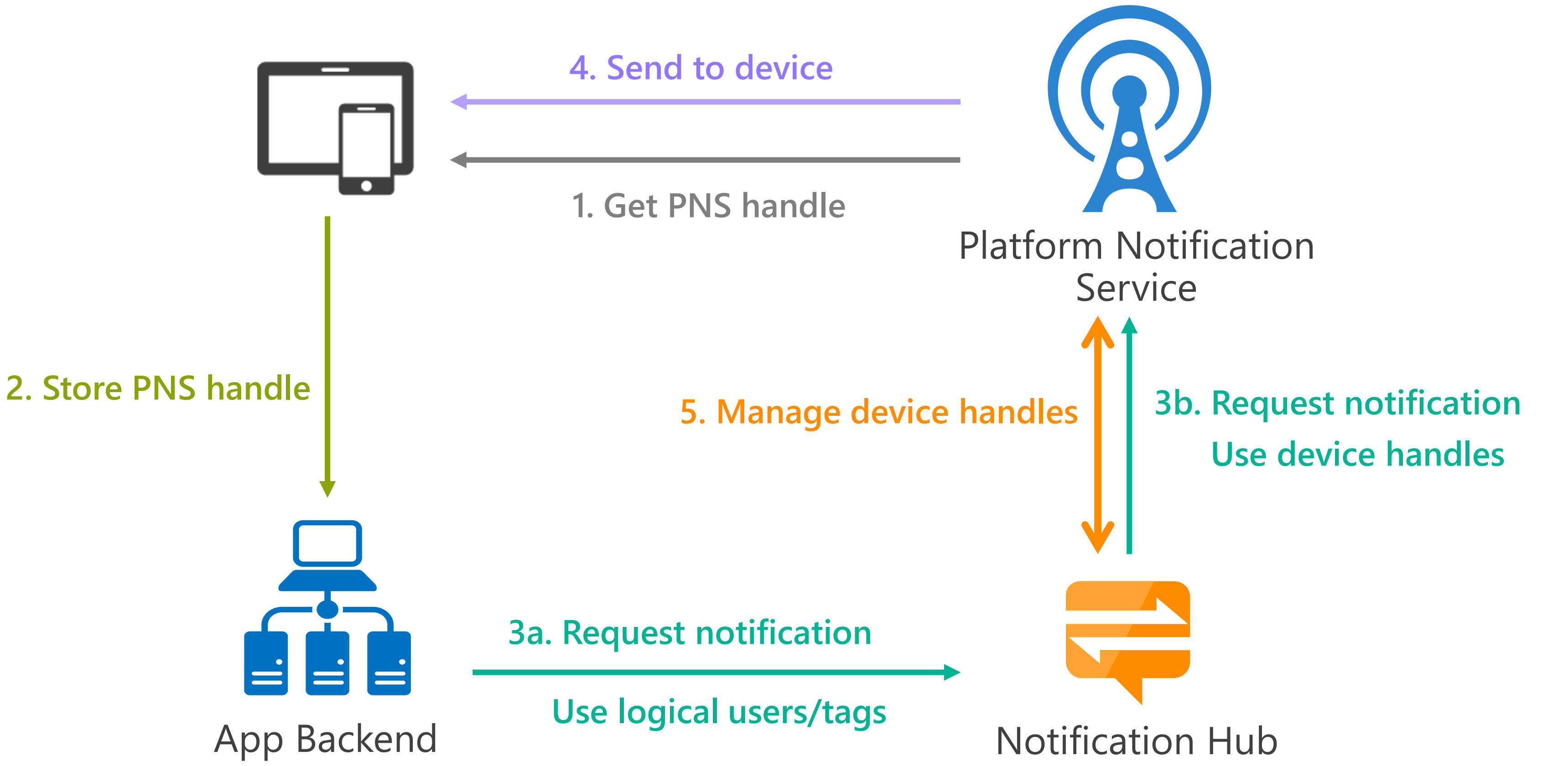
Authentication

- Rolling your own account infrastructure is difficult and time-consuming
- Secure your app with prebuilt authentication providers
    - Facebook
    - Twitter
    - Google
    - Microsoft
    - Azure AD
    - Anything OAuth 2

Push Notifications 101



Push Notifications 101



4. Send to device

1. Get PNS handle

Platform Notification Service

2. Store PNS handle

5. Manage device handles

3b. Request notification

Use device handles

App Backend

3a. Request notification

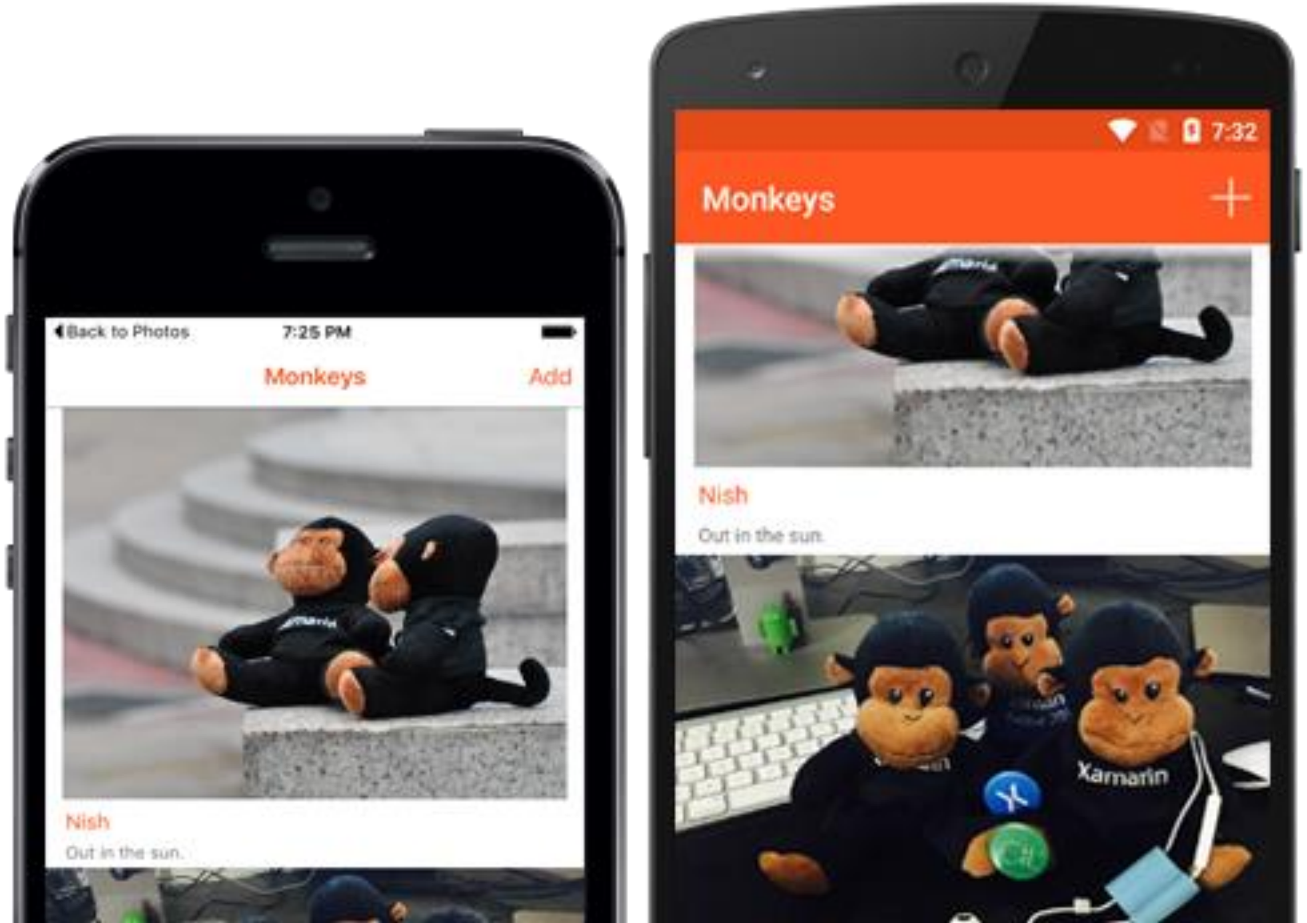
Use logical users/tags

Notification Hub

- Maps between tags and handles

File Sync

- Sync files to Azure Storage, just like you did for structured data.



Sample App: Contoso Moments

- Xamarin.Forms app with file syncing

iOS version in App Store
https://aka.ms/ContosoMomentsiOS

Client and server code (includes Android client):
https://aka.ms/ContosoMomentsCode

App Service Helpers

Add Azure to your .NET app with 4 lines of code

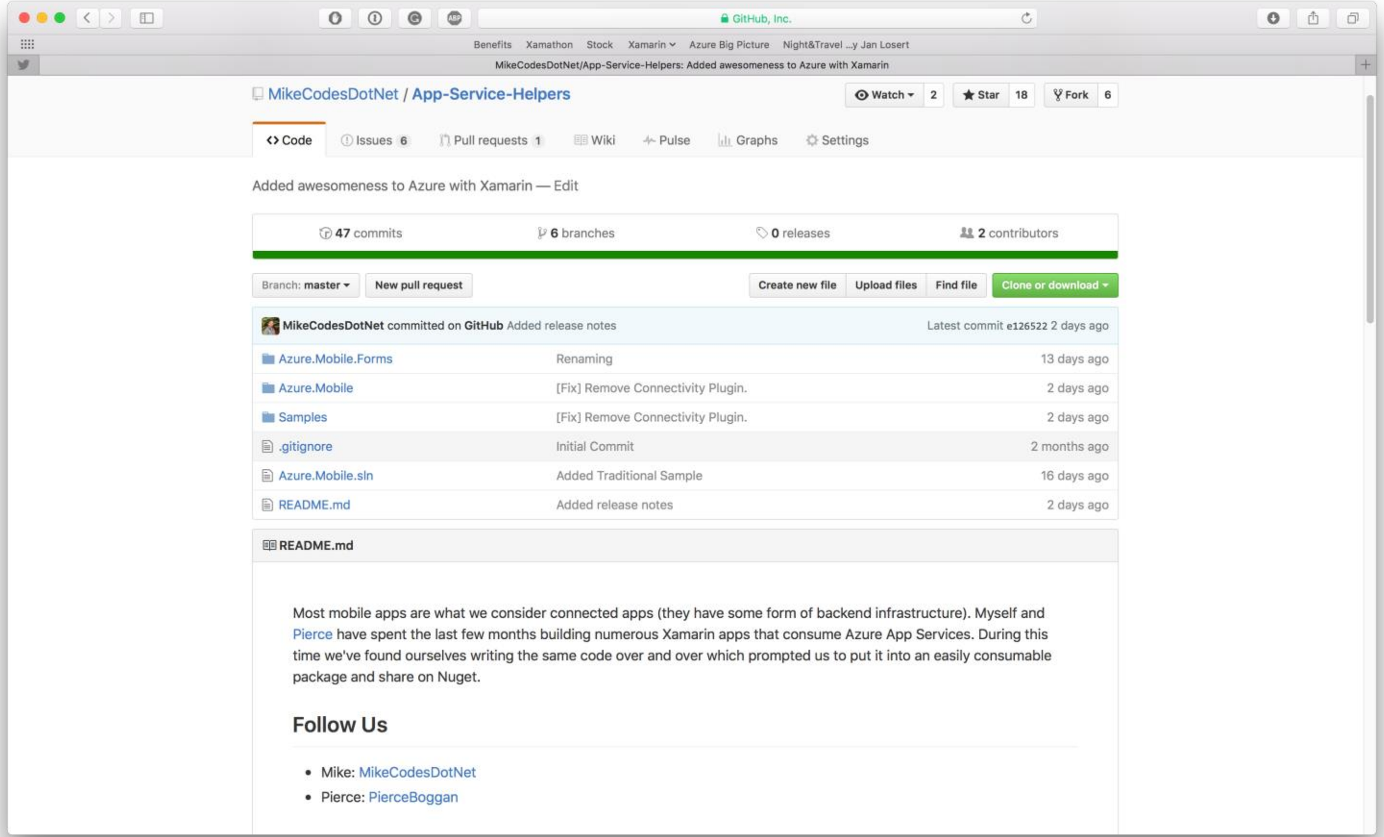
App Service Helper

- Built on top of Azure Mobile SDK
- Modular architecture
- Available on Nuget
- Open Source

Getting Started

```
var client = new AppServiceHelpers.EasyMobileServiceClient();
client.Initialize(Helpers.Keys.AzureServiceUrl);

client.RegisterTable<Headline>();
client.FinalizeSchema();
```

Download today



MikeCodesDotNet/App-Service-Helpers: Added awesomeness to Azure with Xamarin

MikeCodesDotNet / App-Service-Helpers

Added awesomeness to Azure with Xamarin — Edit

47 commits · 6 branches · 0 releases · 2 contributors

MikeCodesDotNet committed on GitHub Added release notes — Latest commit e126522 2 days ago

Azure.Mobile.Forms	Renaming	13 days ago
Azure.Mobile	[Fix] Remove Connectivity Plugin.	2 days ago
Samples	[Fix] Remove Connectivity Plugin.	2 days ago
.gitignore	Initial Commit	2 months ago
Azure.Mobile.sln	Added Traditional Sample	16 days ago
README.md	Added release notes	2 days ago

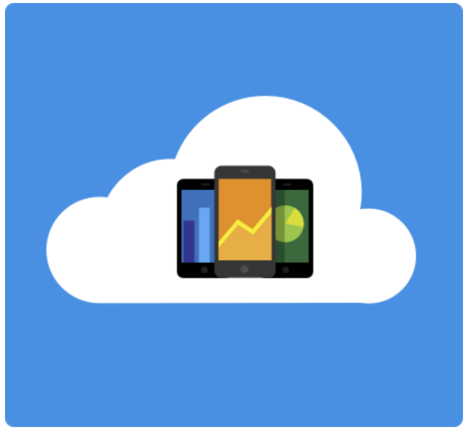
README.md

Most mobile apps are what we consider connected apps (they have some form of backend infrastructure). Myself and Pierce have spent the last few months building numerous Xamarin apps that consume Azure App Services. During this time we've found ourselves writing the same code over and over which prompted us to put it into an easily consumable package and share on Nuget.

Follow Us

- Mike: MikeCodesDotNet
- Pierce: PierceBoggan

github.com/MikeCodesDotNet/App-Service-Helpers



App Service Helpers

Hand’s on during lab after lunch!

Lunch!