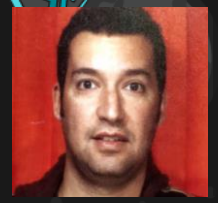
H.O.T. Day

Mobile Performance Management

Dominik Punz
Product Management, Dynatrace



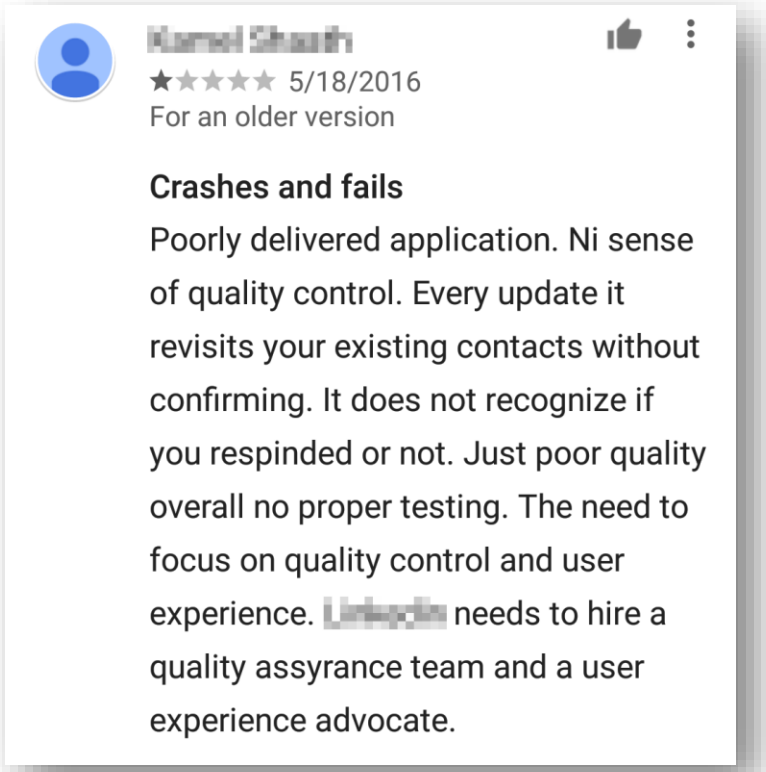
Steve Caron
Sales Engineering, Dynatrace

What to expect from that session

- Learn about mobile monitoring with Dynatrace
- Differences between AppMon and Dynatrace
- Hands-on experience with our mobile agents
- Instrument an Android app in Android Studio
- Instrument an Ionic app using the Cordova plugin
- Some coding to report additional data
- Sample apps and instructions available on GitHub

What is mobile monitoring and why should I care?

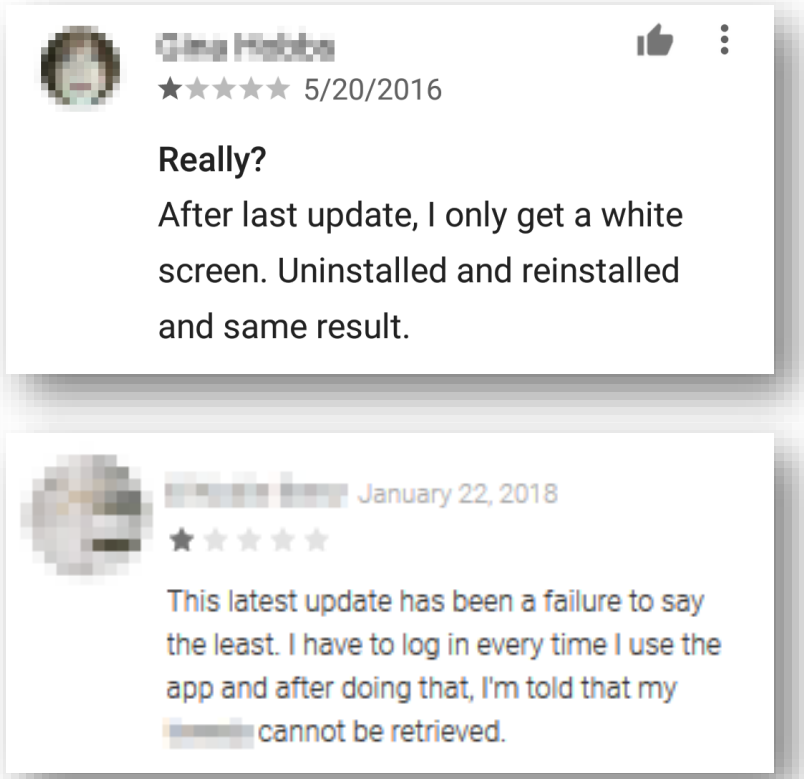
„One star“ ratings



[redacted]
★☆☆☆☆ 5/18/2016
For an older version

Crashes and fails

Poorly delivered application. Ni sense of quality control. Every update it revisits your existing contacts without confirming. It does not recognize if you respinded or not. Just poor quality overall no proper testing. The need to focus on quality control and user experience. [redacted] needs to hire a quality assyrance team and a user experience advocate.



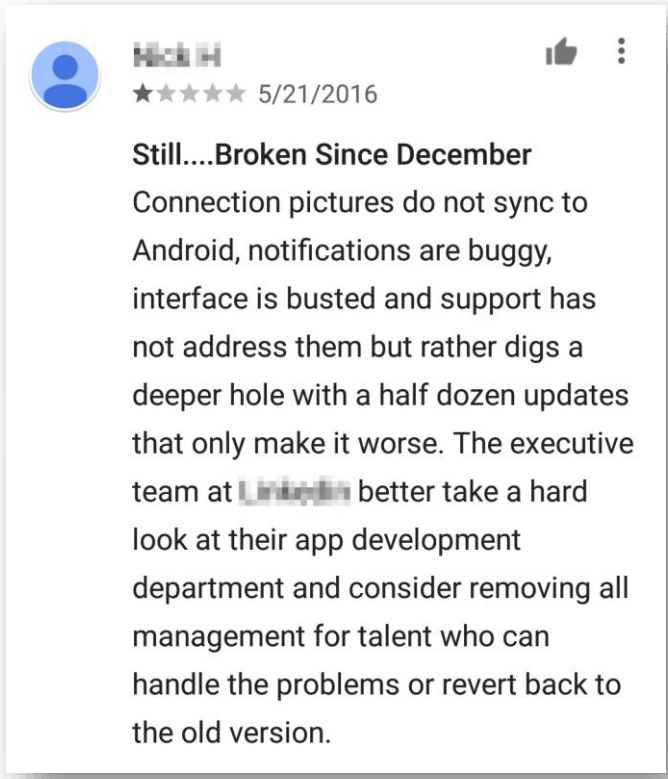
[redacted]
★☆☆☆☆ 5/20/2016

Really?

After last update, I only get a white screen. Uninstalled and reinstalled and same result.

[redacted] January 22, 2018
★☆☆☆☆

This latest update has been a failure to say the least. I have to log in every time I use the app and after doing that, I'm told that my [redacted] cannot be retrieved.



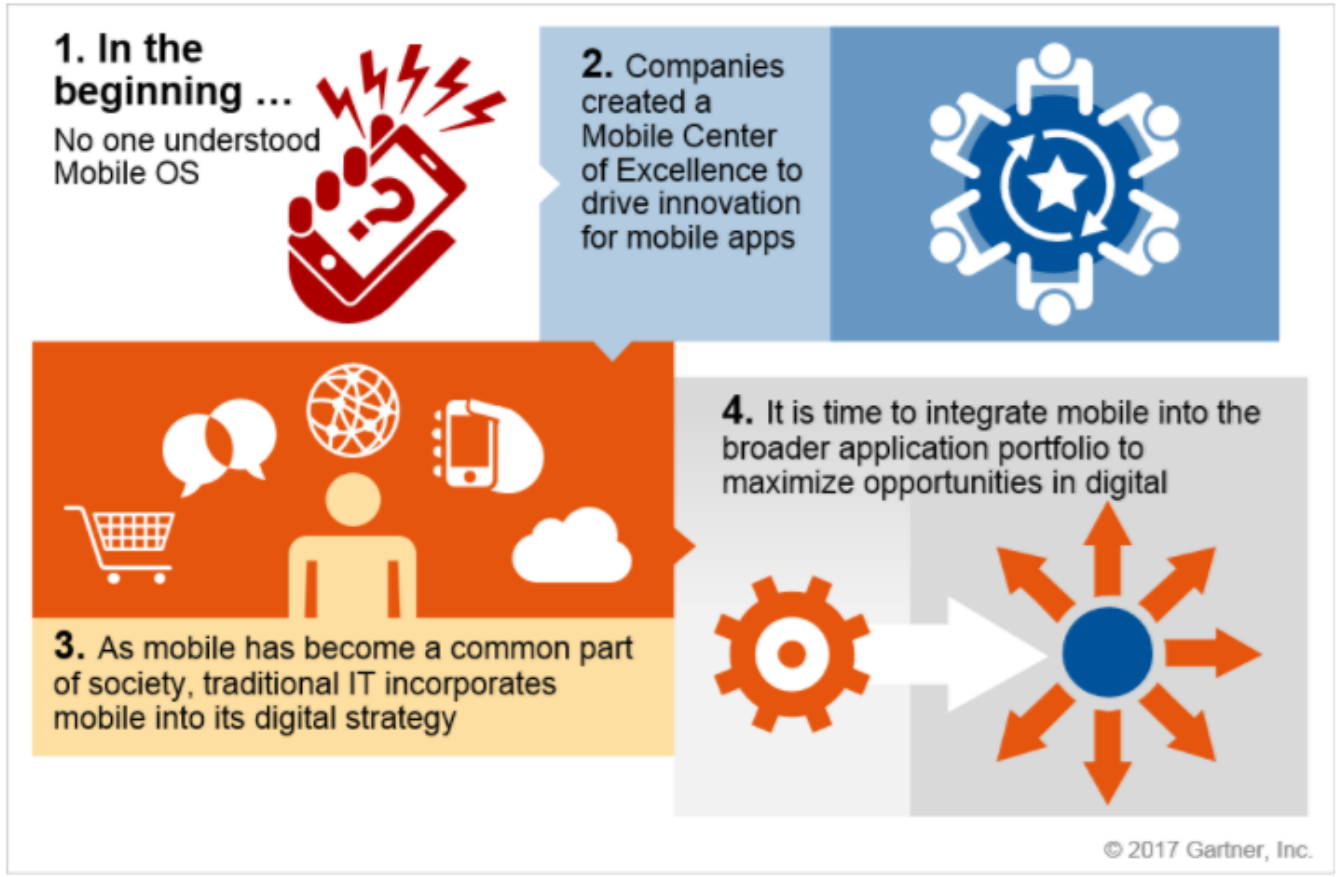
[redacted]
★☆☆☆☆ 5/21/2016

Still....Broken Since December

Connection pictures do not sync to Android, notifications are buggy, interface is busted and support has not address them but rather digs a deeper hole with a half dozen updates that only make it worse. The executive team at [redacted] better take a hard look at their app development department and consider removing all management for talent who can handle the problems or revert back to the old version.

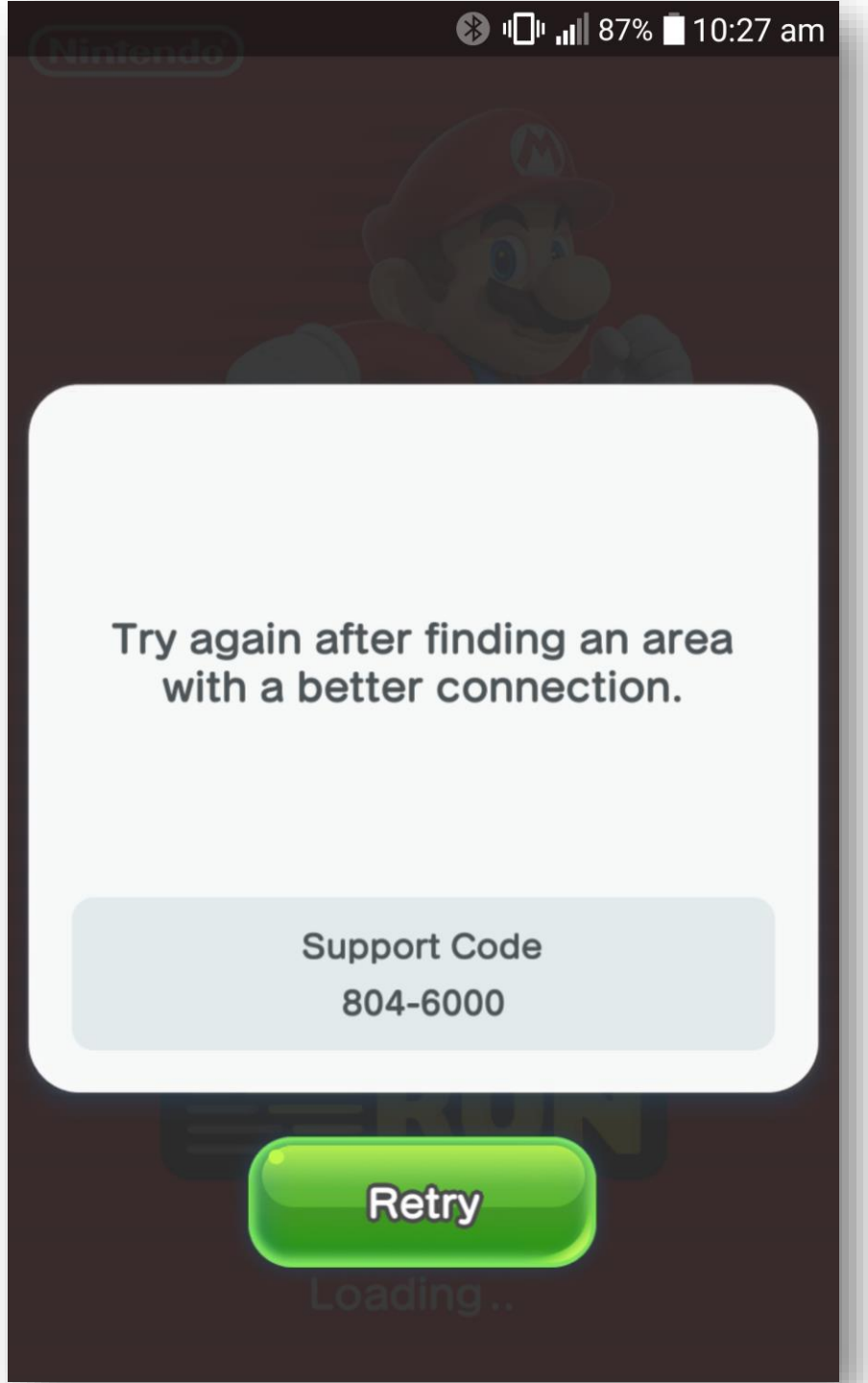
Mobile apps are no longer stand-alone solutions

Figure 2. Evolving Your Mobile Maturity to Capitalize on Innovation at Scale

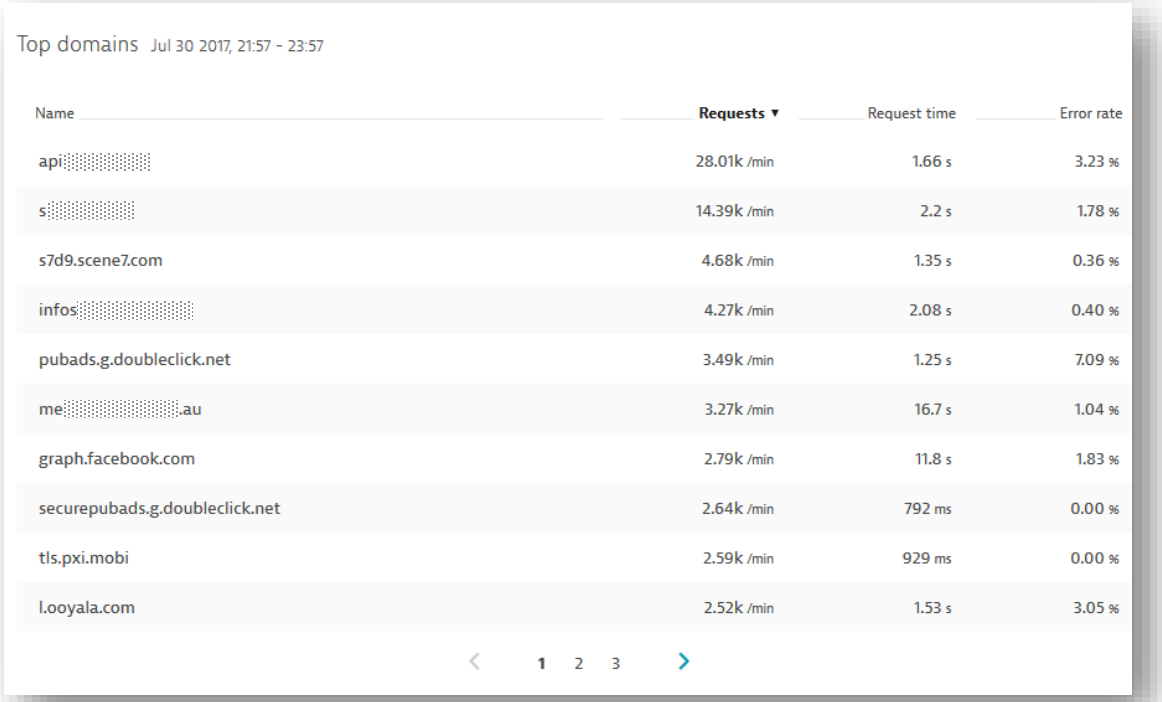


Source: Gartner (July 2017)

Gartner, It's Time to Integrate a Mobile App Strategy Into Your Application Strategy, 27 July 2017 ID: G00327299, Dennis Gaughan, Deacon D.K Wan, G00327299


87% 10:27 am
Nintendo
Try again after finding an area with a better connection.
Support Code
804-6000
Retry
Loading..

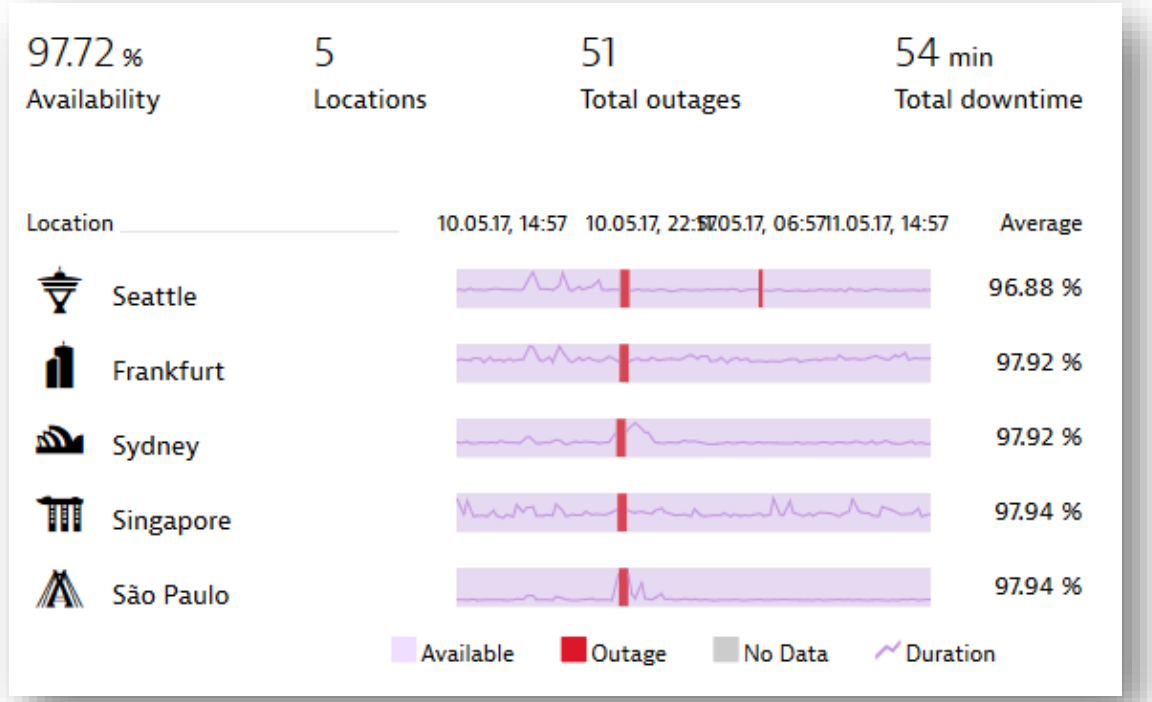
Watch your availability



Top domains Jul 30 2017, 21:57 - 23:57

Name	Requests ▾	Request time	Error rate
api[redacted]	28.01k /min	1.66 s	3.23 %
s[redacted]	14.39k /min	2.2 s	1.78 %
s7d9.scene7.com	4.68k /min	1.35 s	0.36 %
infos[redacted]	4.27k /min	2.08 s	0.40 %
pubads.g.doubleclick.net	3.49k /min	1.25 s	7.09 %
me[redacted].au	3.27k /min	16.7 s	1.04 %
graph.facebook.com	2.79k /min	11.8 s	1.83 %
securepubads.g.doubleclick.net	2.64k /min	792 ms	0.00 %
tls.pxi.mobi	2.59k /min	929 ms	0.00 %
l.ooyala.com	2.52k /min	1.53 s	3.05 %

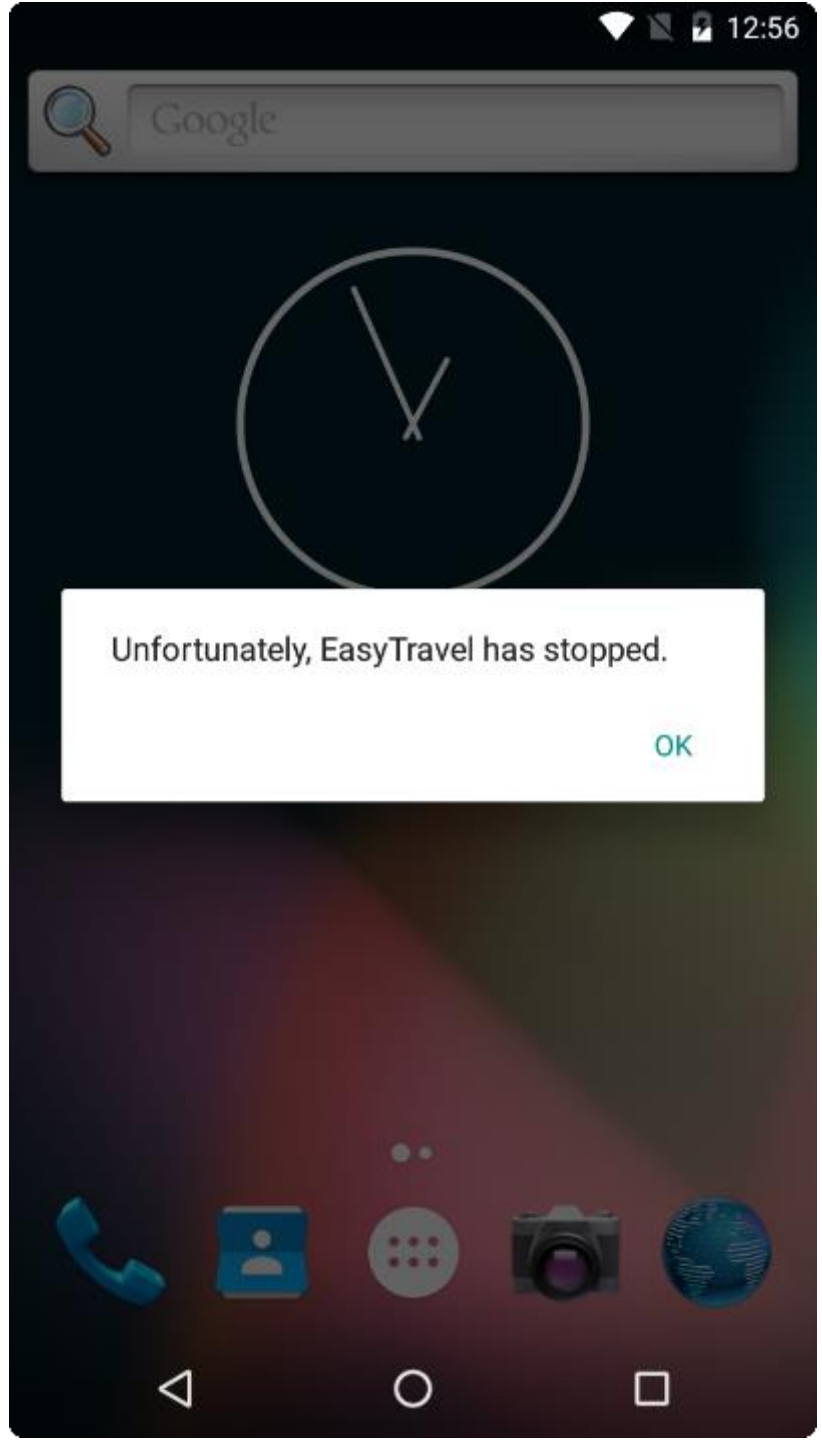
1 2 3



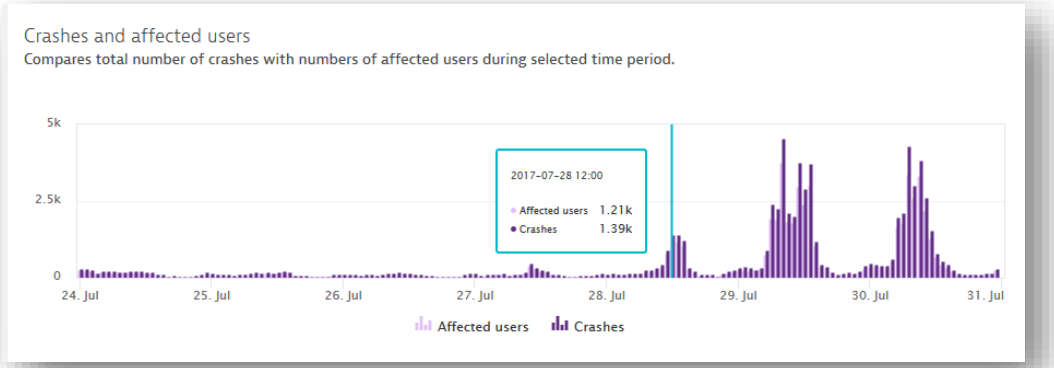
97.72 %	5	51	54 min
Availability	Locations	Total outages	Total downtime

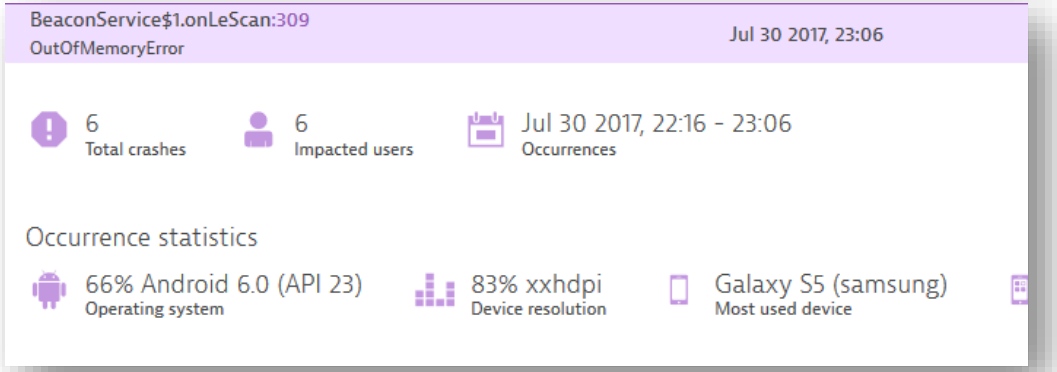
Location	10.05.17, 14:57	10.05.17, 22:57	11.05.17, 06:57	11.05.17, 14:57	Average
Seattle					96.88 %
Frankfurt					97.92 %
Sydney					97.92 %
Singapore					97.94 %
São Paulo					97.94 %

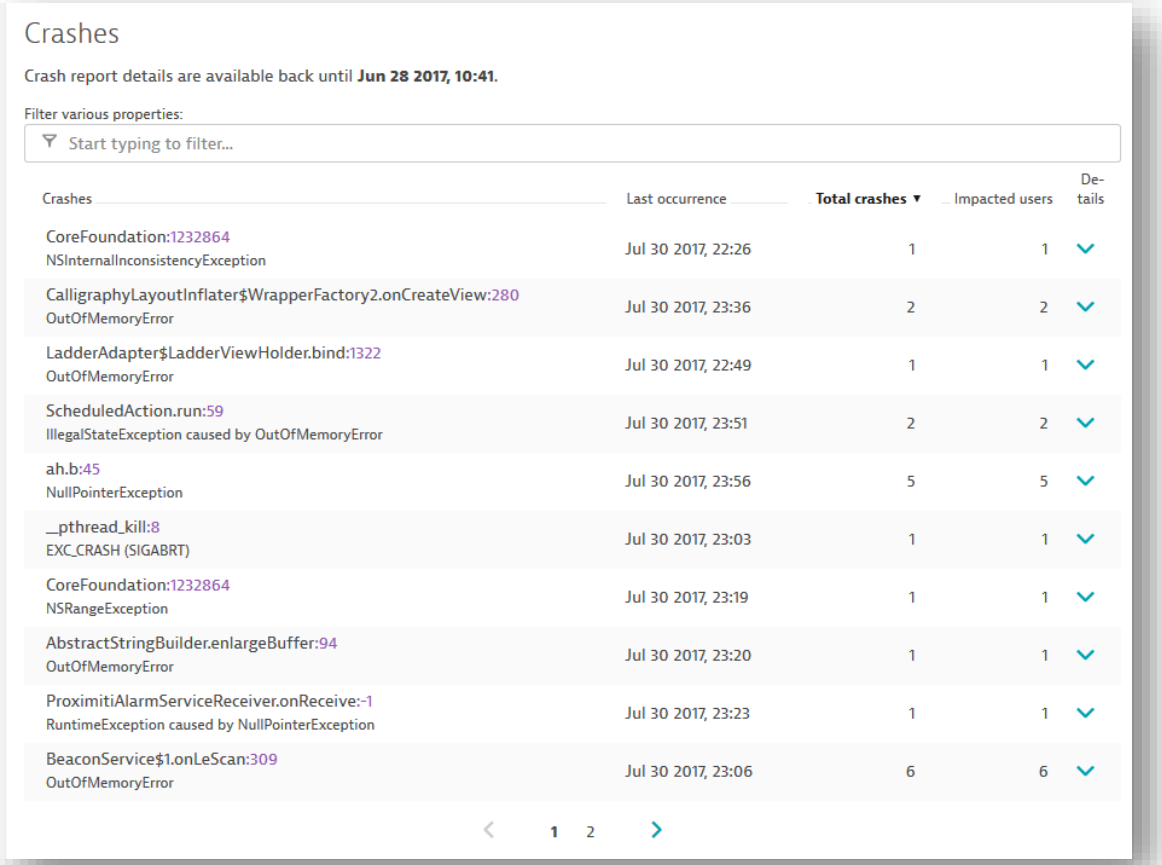
Available Outage No Data Duration

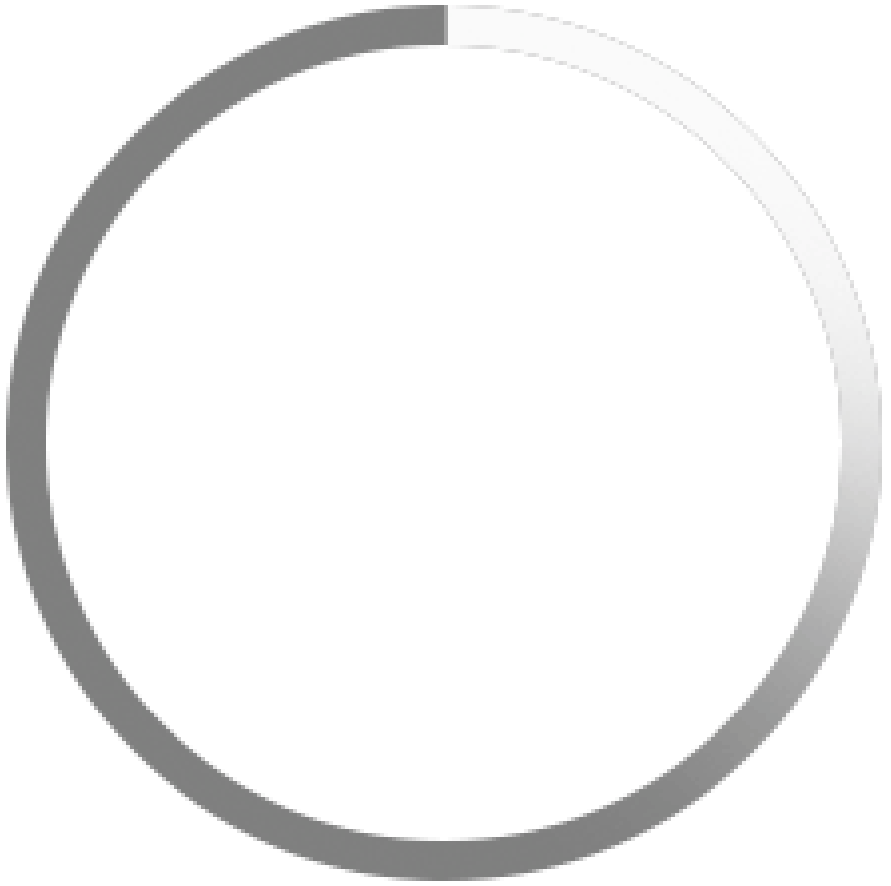

12:56
Google
Unfortunately, EasyTravel has stopped.
OK

Watch your crashes









Watch your performance

Top user actions

Shows the top user actions and their contribution to the overall app performance

Type	User action	Actions	Duration	Details
	Touch on Search	22.93 /min	7 ms	
	AppStart (easyTravel)	9.47 /min	23 ms	
	Touch on User Account	8.00 /min	7 ms	
	Display SearchJourneyActivity	7.64 /min	27 ms	
	Touch on Amsterdam	6.13 /min	7 ms	
	Touch on Rome	6.00 /min	7 ms	
	Touch on Paris	5.73 /min	7 ms	
	Touch on New York	5.64 /min	7 ms	
	Touch on Vienna	5.42 /min	7 ms	
	Touch on Berlin	4.98 /min	7 ms	

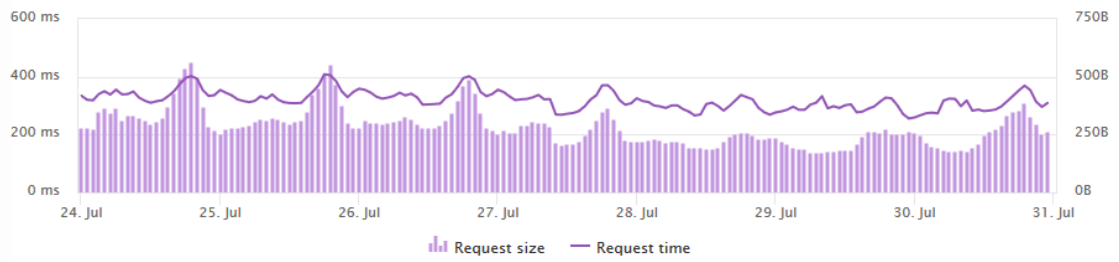
1 2

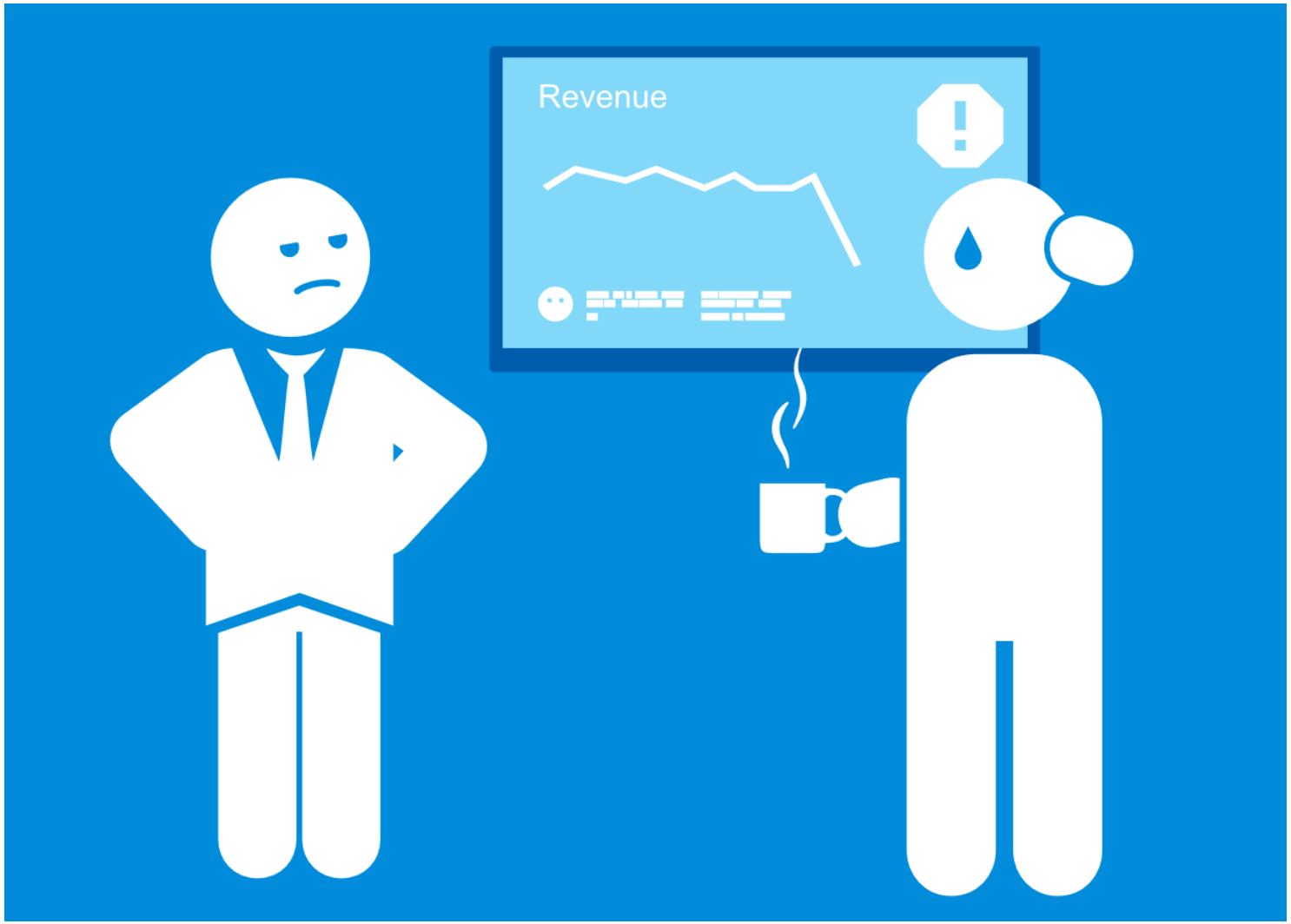
Top HTTP Requests

Name	Requests	Request time	Error rate	Details
/services/JourneyService/findJourneys	12.89 /min	466 ms	0.00 %	
/services/JourneyService/findLocations	10.80 /min	339 ms	0.00 %	
/services/AuthenticationService/authenticate	8.89 /min	231 ms	0.00 %	
/services/AdService/	4.98 /min	25.5 ms	100.00 %	
/services/BookingService/storeBooking	4.71 /min	4.77 s	0.00 %	

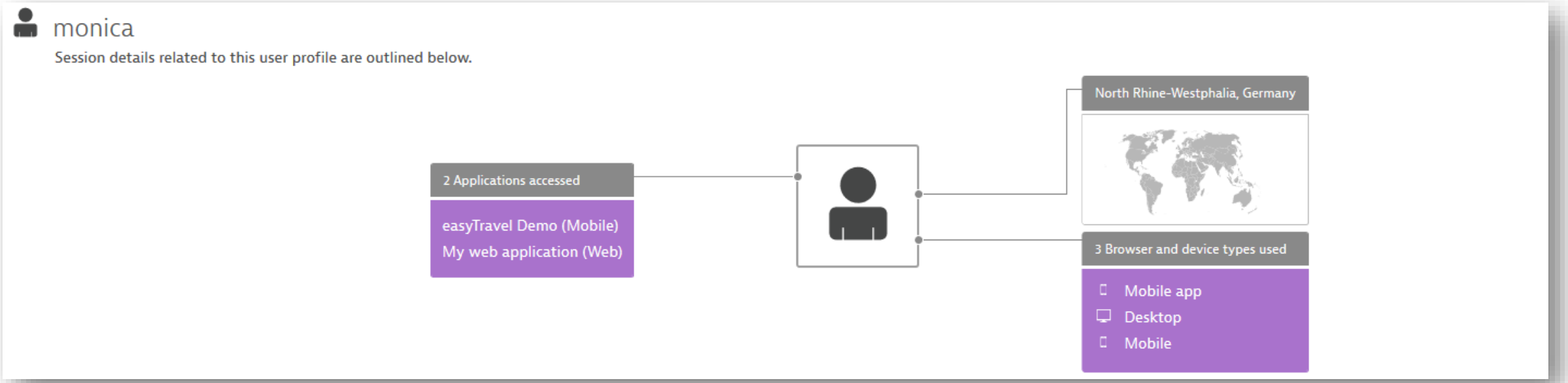
Web request size and performance

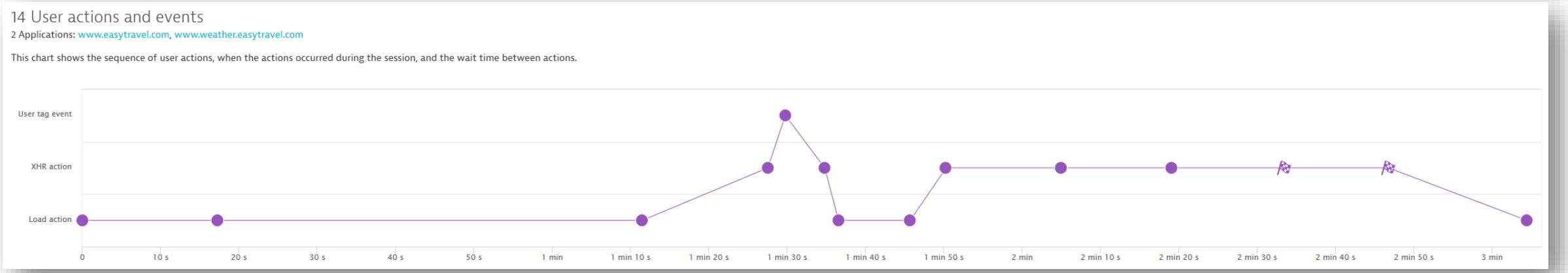
Compares the web request size with client-side performance during selected time frame.

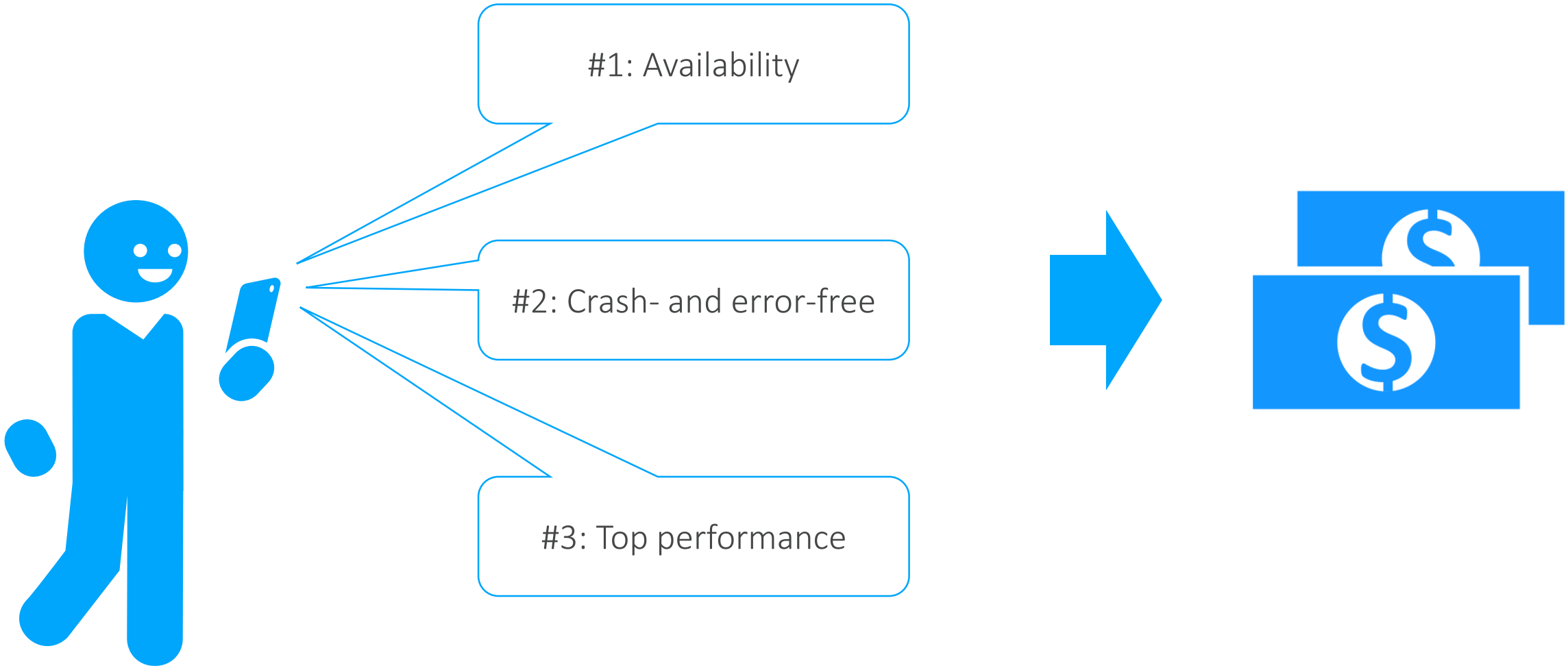


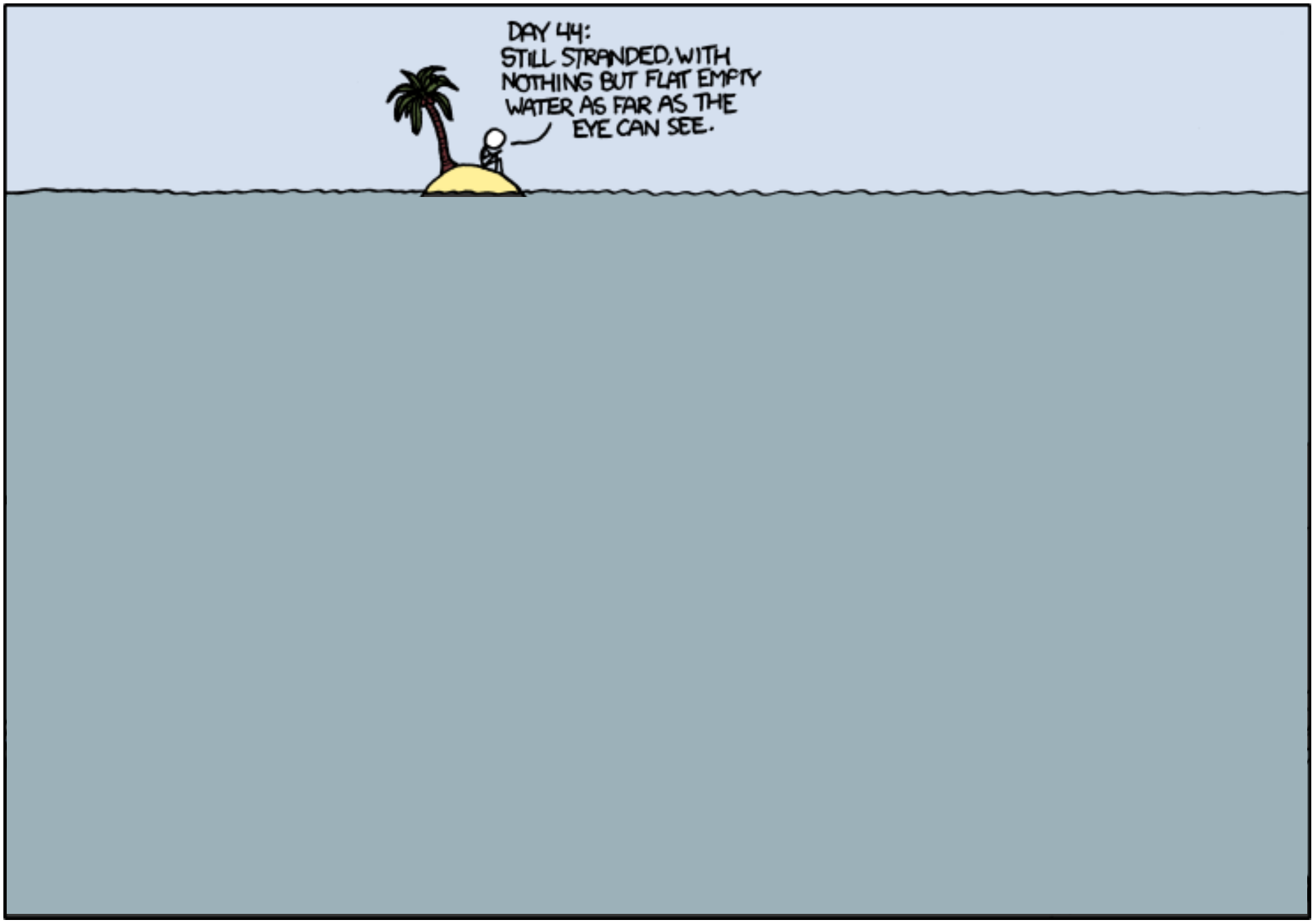

Revenue

Monitor your business



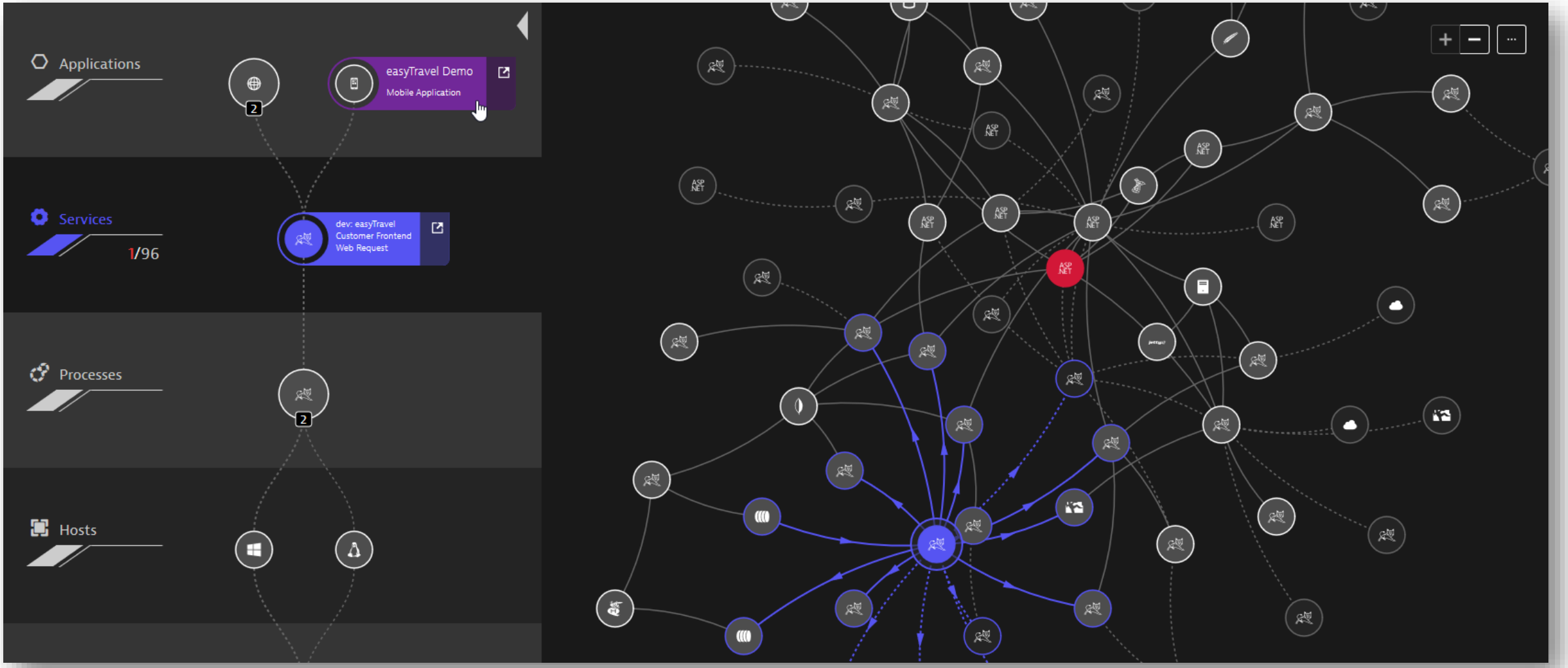


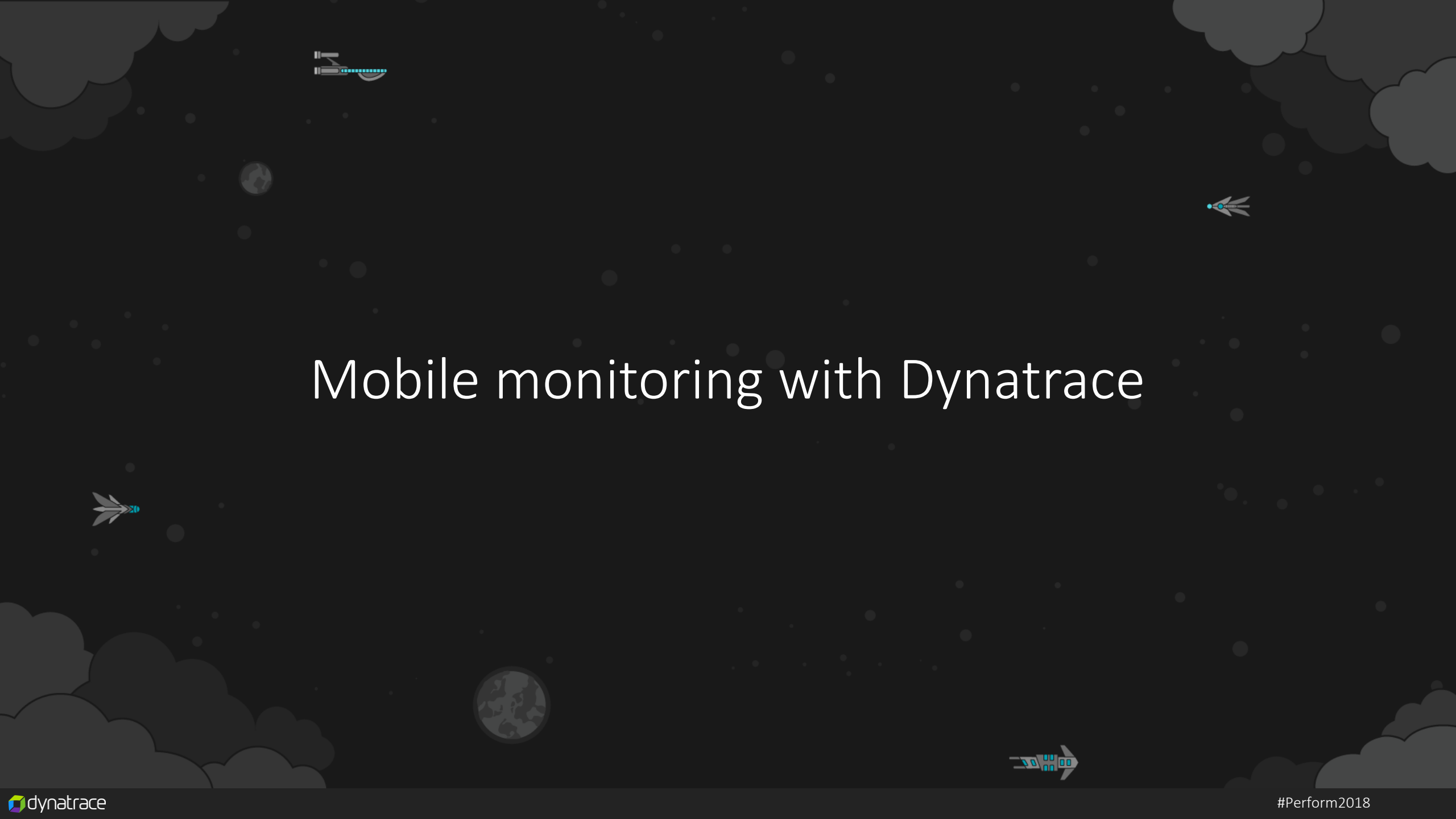

#1: Availability
#2: Crash- and error-free
#3: Top performance



https://xkcd.com/731/

Mobile is an essential part of the bigger picture



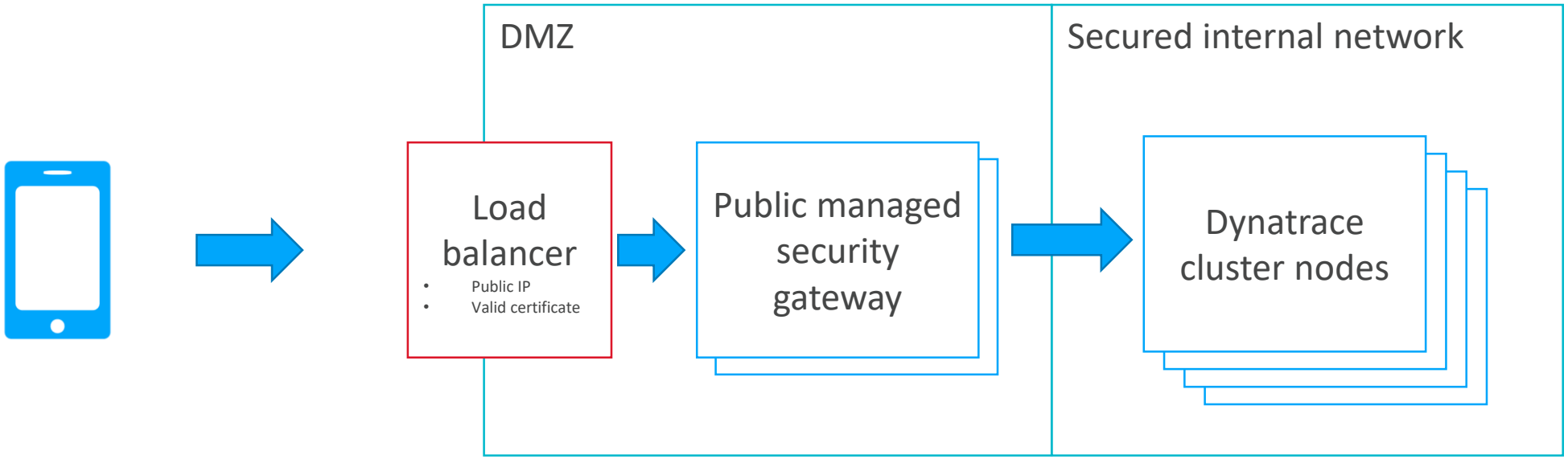

Mobile monitoring with Dynatrace

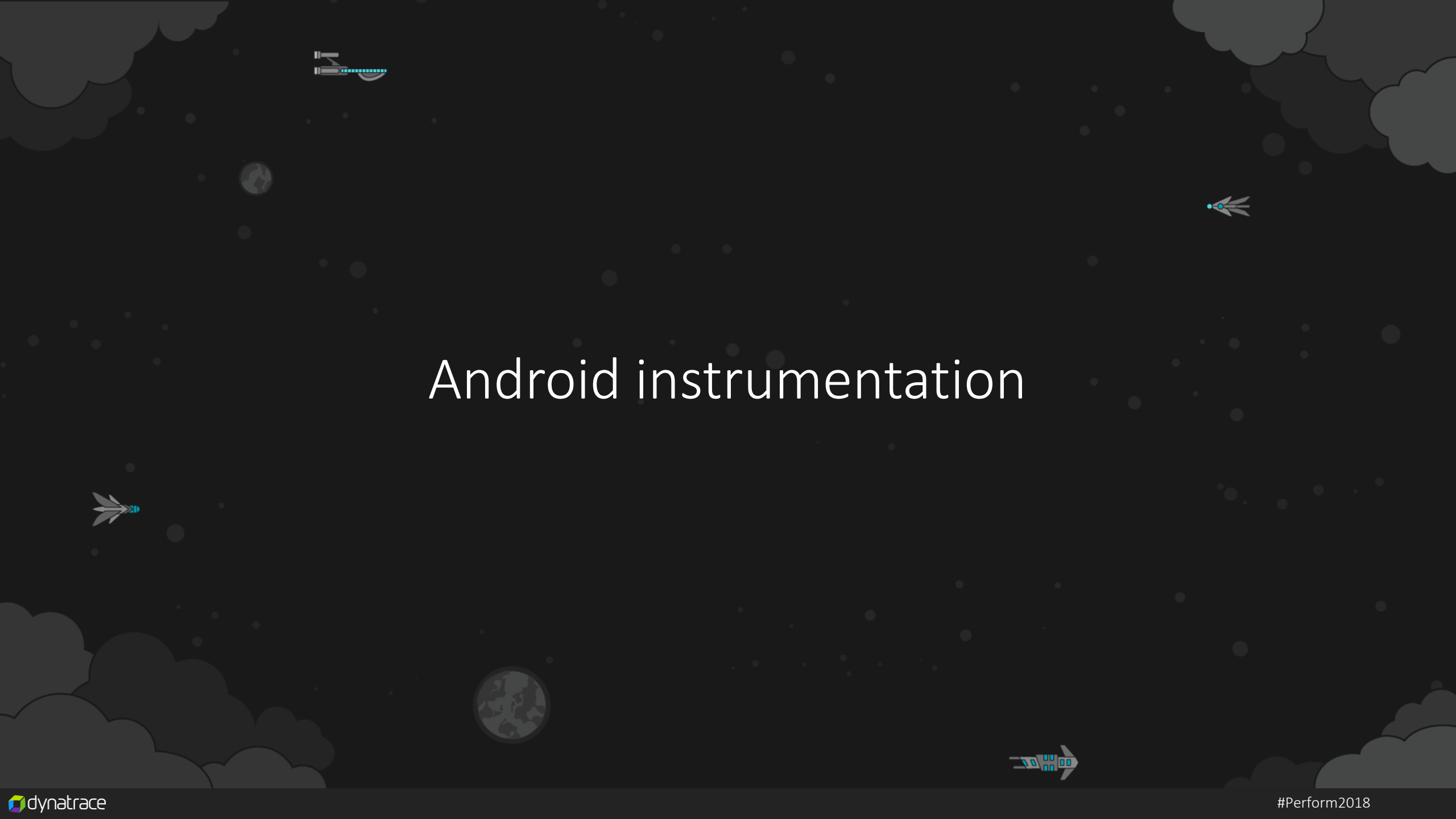
Lesson #1 – The demo environment

- Open easyTravel website and book a new travel
- Log in to Dynatrace
- View the environment in smartscape
- View the web application

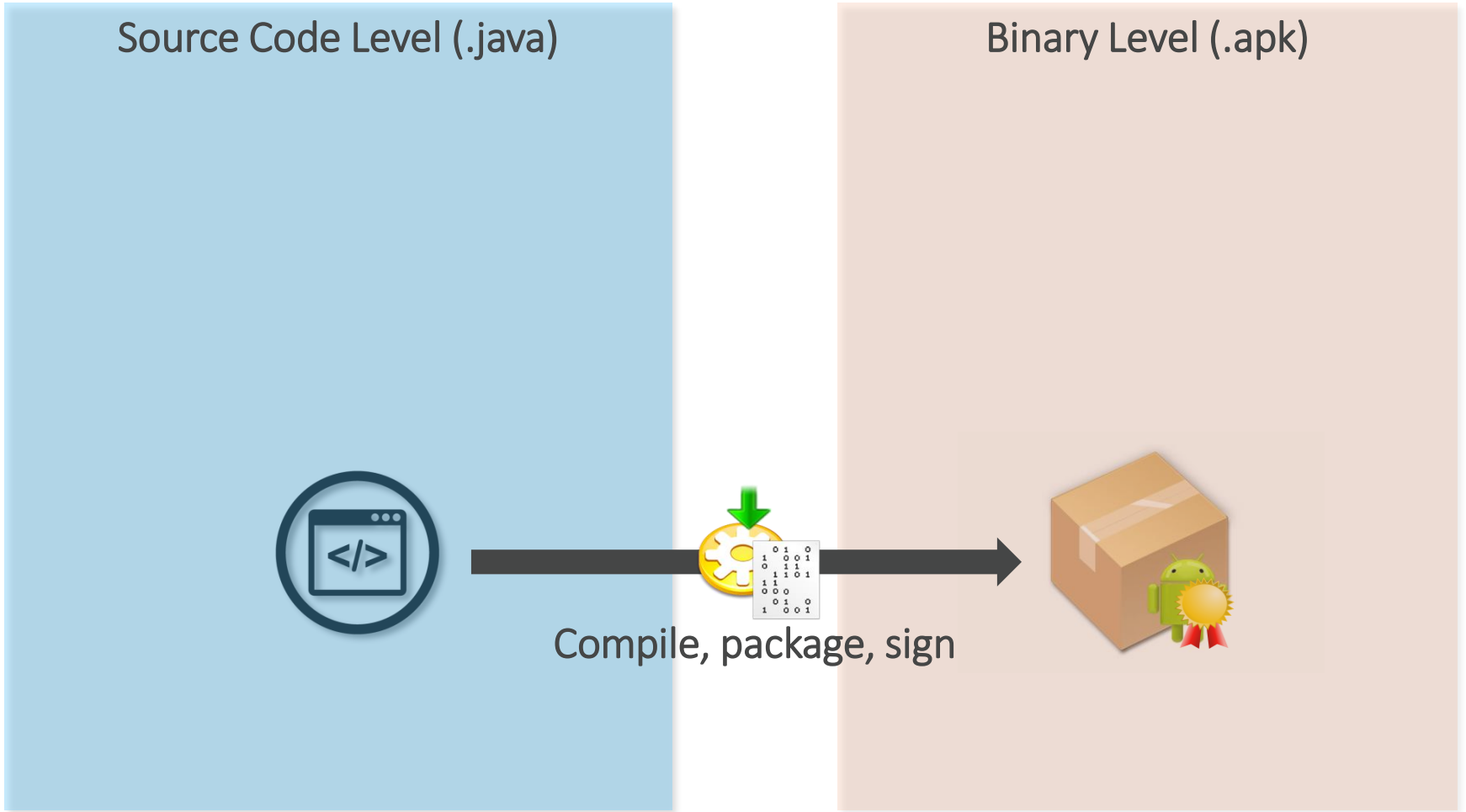
https://github.com/Dynatrace/mobile-hotday-2018

Dynatrace Managed

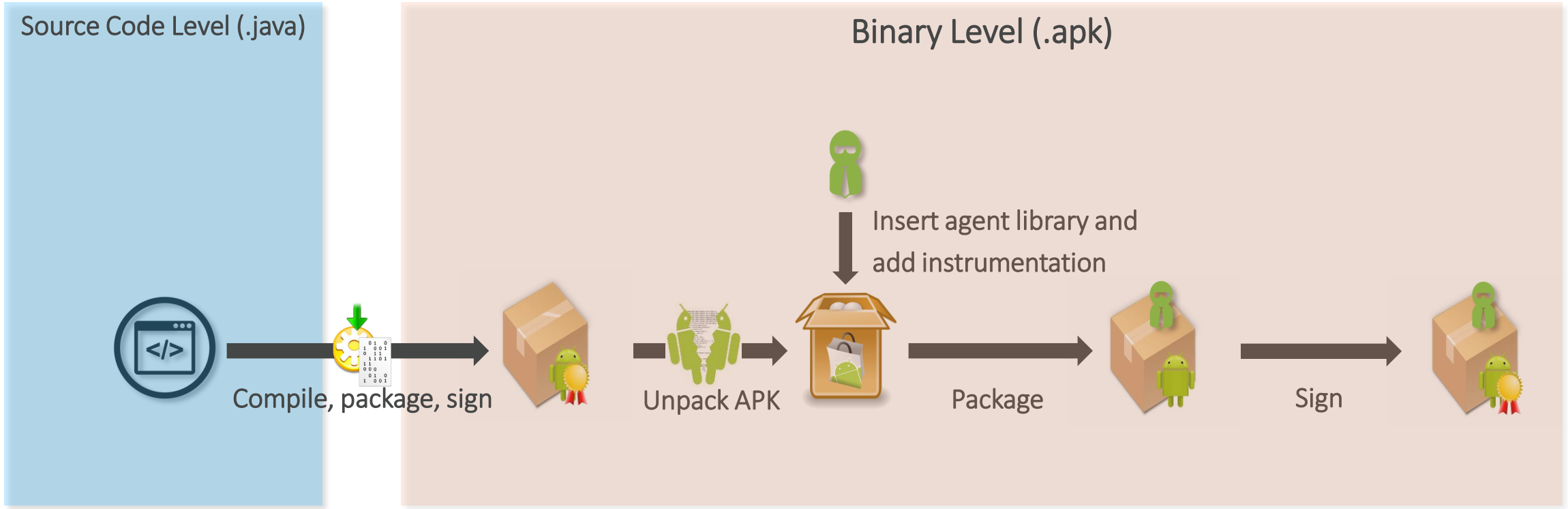



Android instrumentation

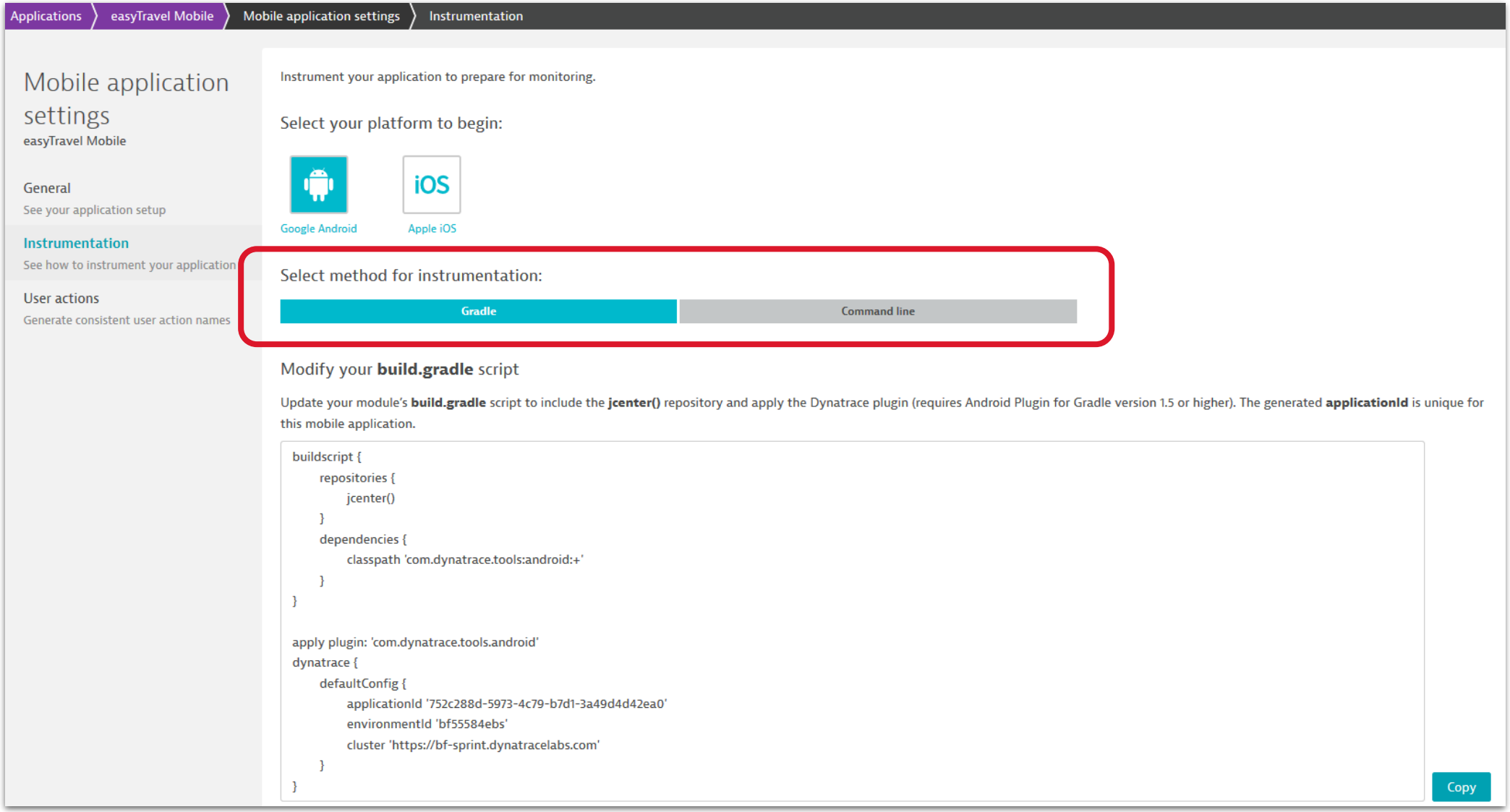
Building an Android app



Android Auto-instrumentation



How to use Android auto-instrumentation

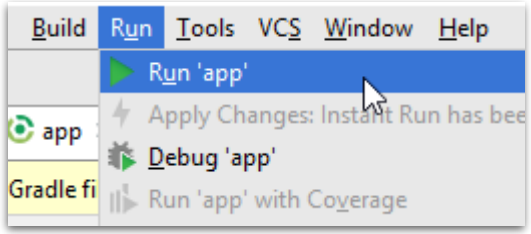


Lesson #2 - Auto-instrumentation in Android Studio

- Check out and configure easytravel-android
  - `git clone https://github.com/Dynatrace/mobile-hotday-2018.git`
- Auto instrument the mobile app
- Crash reporting

Can I disable auto-instrumentation in development?

- Keep the instant run feature enabled
- Avoid to wait for auto instrumentation every time you run the app
- Use a Gradle property to activate or deactivate the plugin
- Run button starts the app uninstrumented



- Add the –Pinstrument property for the release build and the app gets auto instrumented

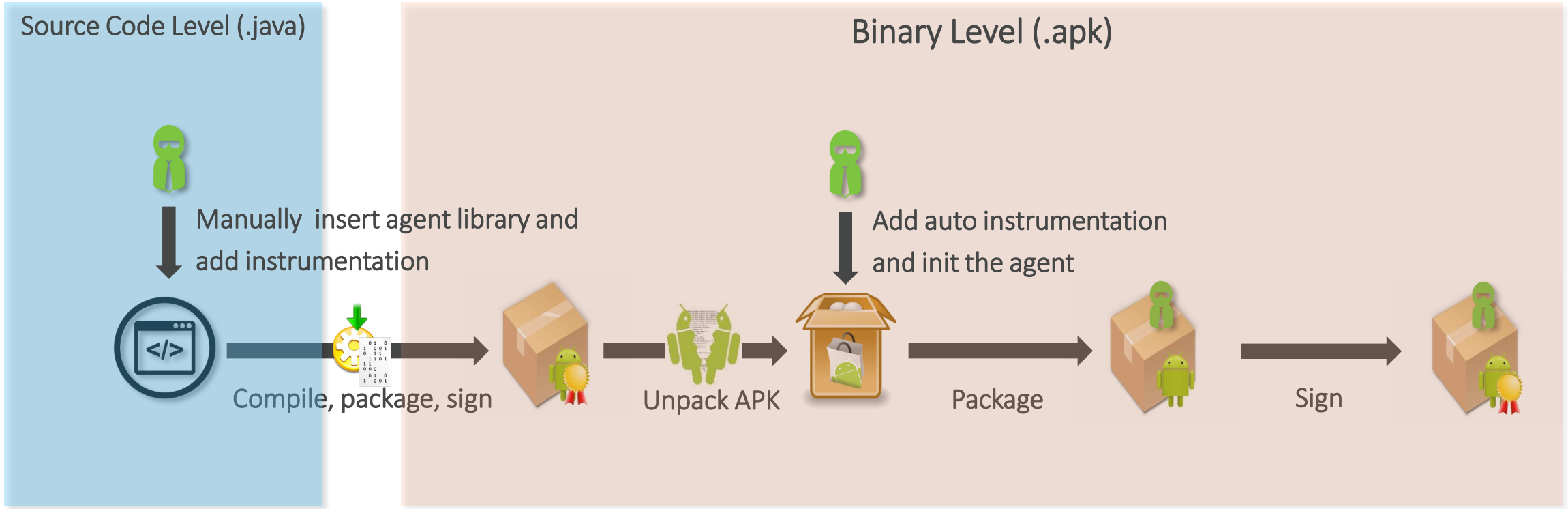
```
>gradlew assembleRelease -Pinstrument
```

```
if(project.hasProperty('instrument')) {
    apply plugin: 'com.dynatrace.tools.android'
    dynatrace {
        defaultConfig {
            applicationId '3fdeefbb-6cfe-45f4-9085-e50f3fb668cc'
            environmentId 'bf32559dsi'
            cluster 'https://bf-sprint.dynatracelabs.com'
        }
    }
}
```

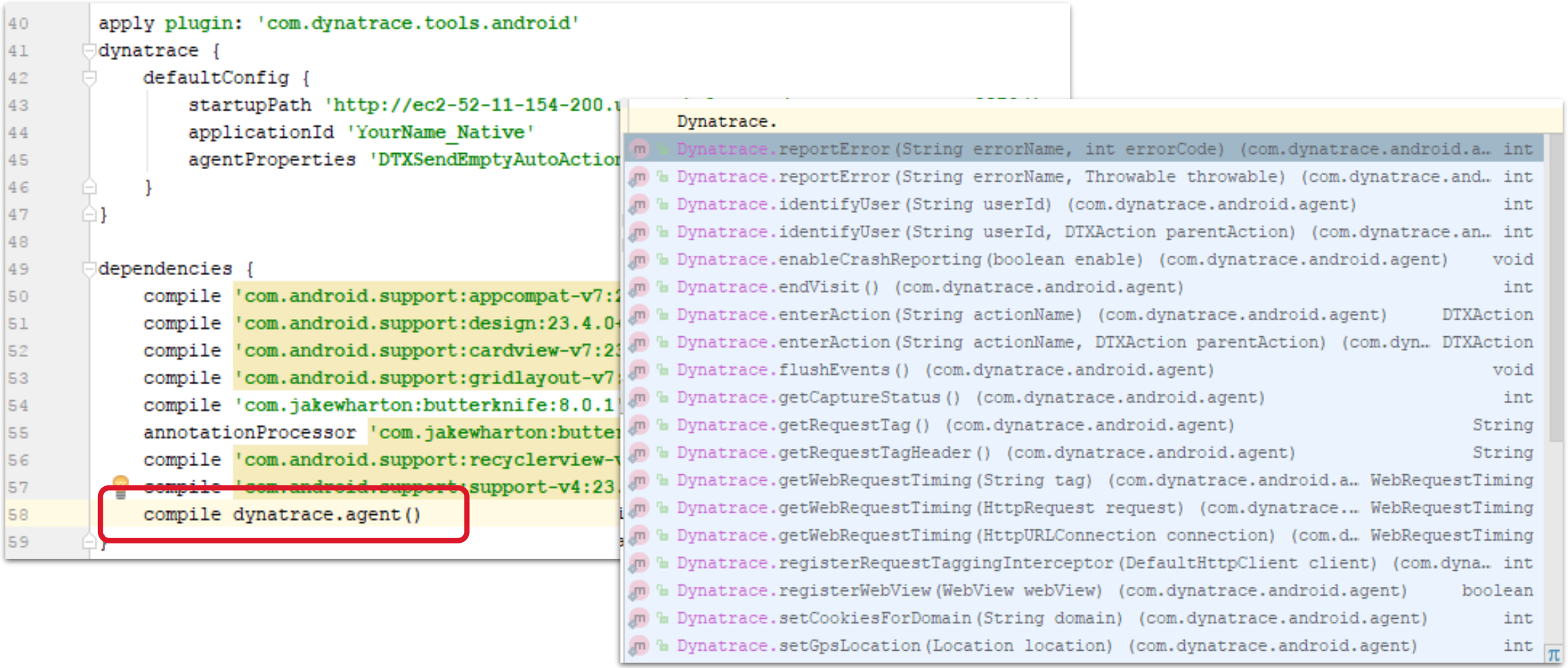
Multidex apps

- Android code is split into .dex files (64K limit)
- Dynatrace agent must be part of the primary .dex file (startup call)
- Auto instrumentor analyzes dependencies and automatically moves classes to make room
- Customization necessary if multiDexKeepFile or multiDexKeepProguard is used
  - `'DTXMultiDexKeepFile': <appmodule>/build/intermediates/multi-dex/<build_variant>/maindexlist.txt`
- If auto instrumentation fails, limits can adjusted manually
  - `DTXPrimaryDexLimit, DTXSecondaryDexLimit`
- Or adapt your build to generate a primary dex with enough room
  - `dexOptions {`
    `additionalParameters '--minimal-main-dex'`
  - `}`
- Test your multidex app on Android 4.4 or lower to verify that the Dalvik VM can handle it

Manual instrumentation



How to add the agent library to your project

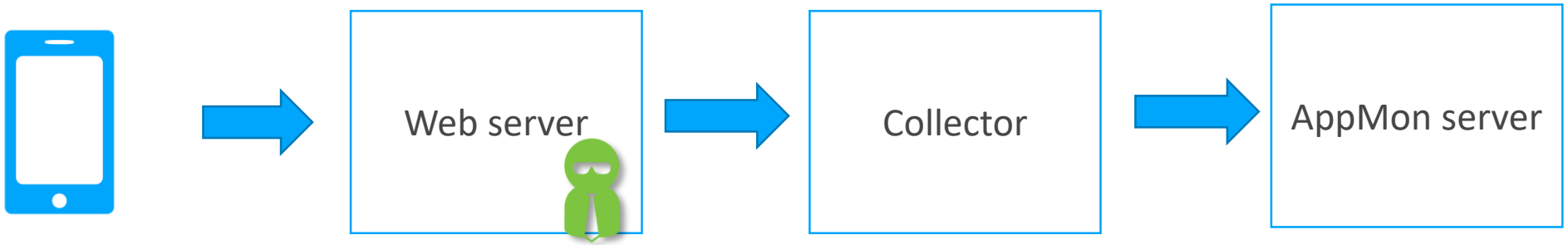


Lesson #3 – Manual instrumentation

- Add dynatrace as a library at compile time
- Identify user
- Custom user actions
- Reporting errors

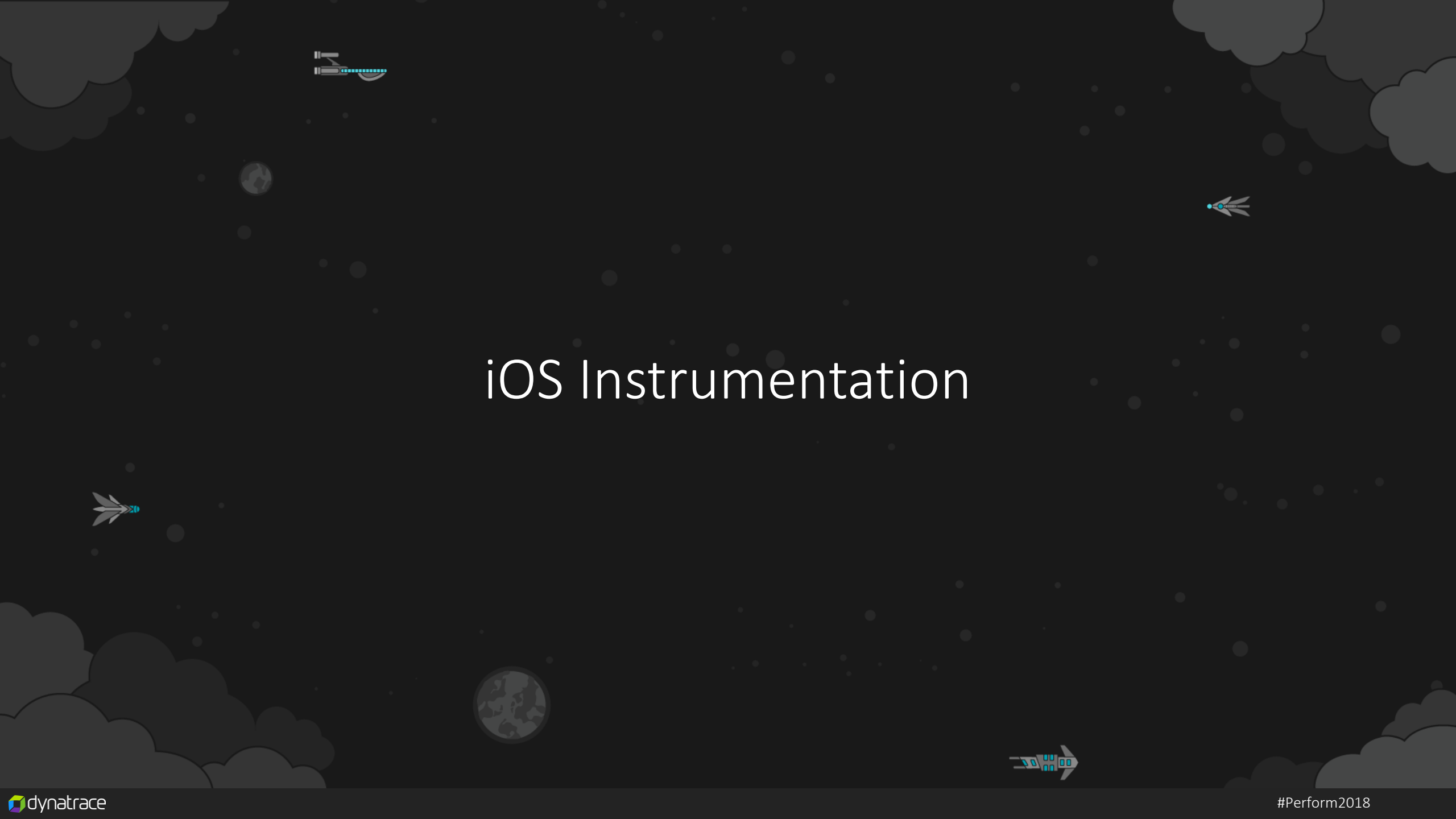
Mobile Monitoring with AppMon

Mobile monitoring with AppMon

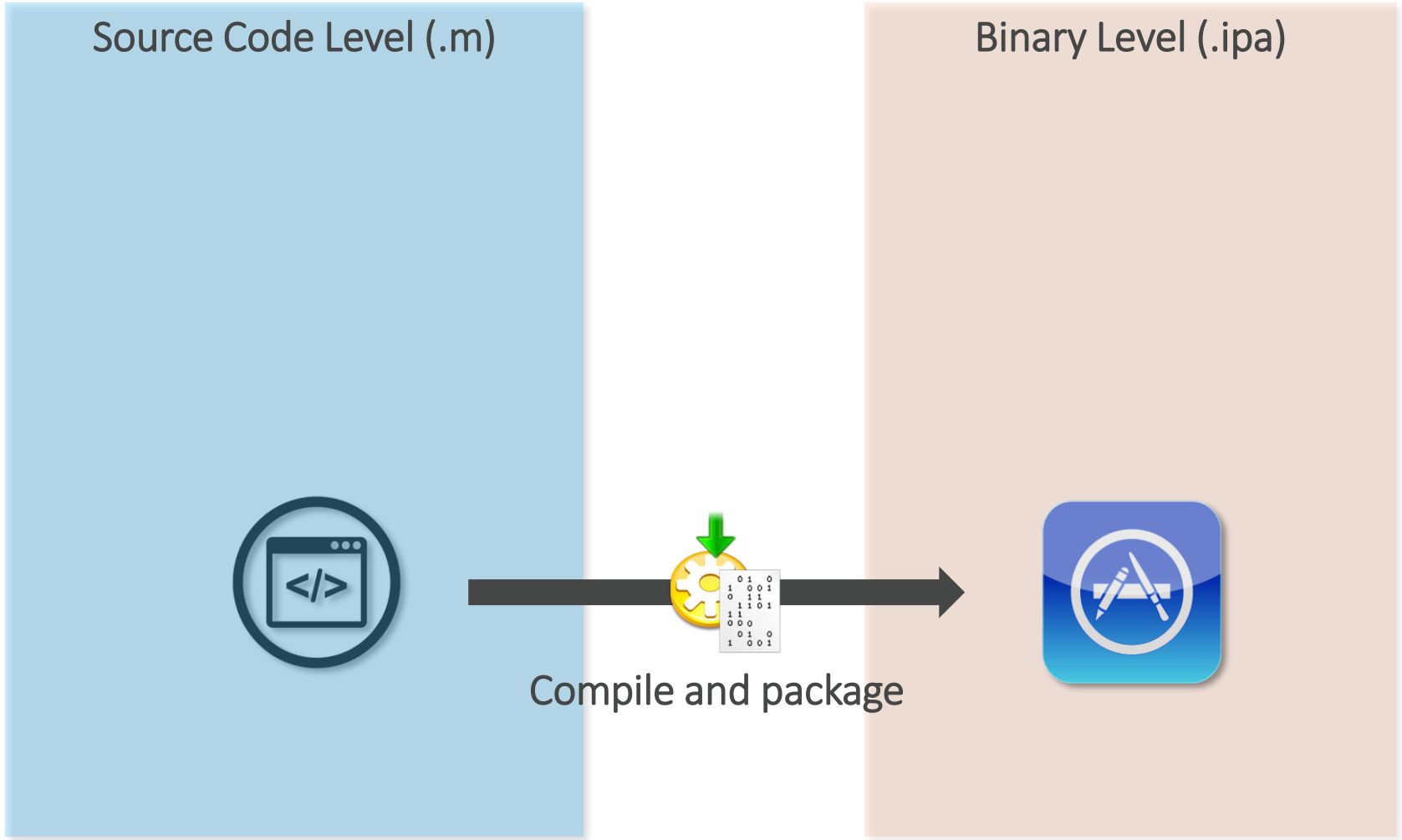


Lesson #4 – Connect the Android app to AppMon

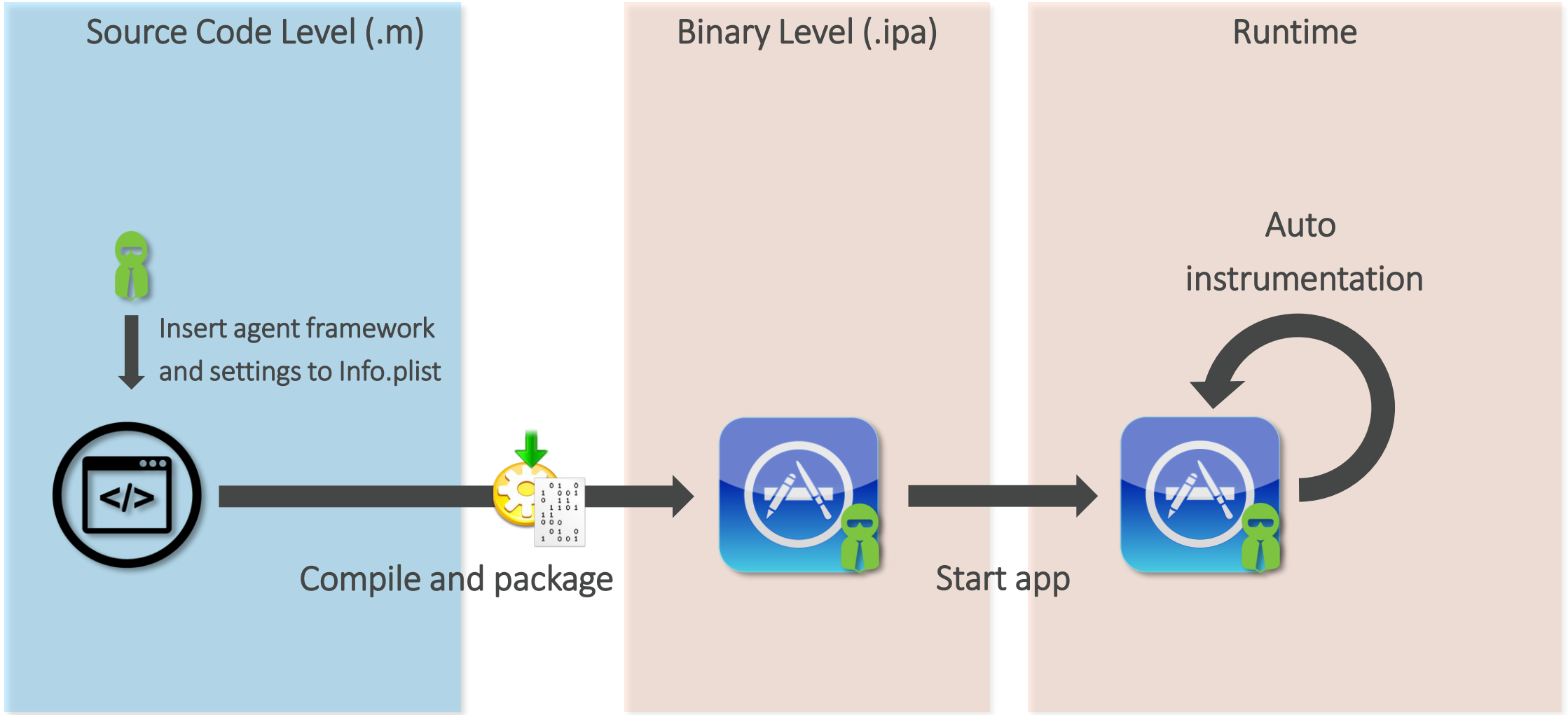
- Change the easyTravel backend URL to another easyTravel instance
- Change the app module's build.gradle to report data to AppMon
- Generate some data and analyze it in AppMon


iOS Instrumentation

Building an iOS app



The Native iOS Agent



Lesson #5 - Auto instrumentation in Xcode

- Import the iOS demo app in Xcode and run it in the emulator
- Use CocoaPods to insert the mobile agent
- Analyze the data in Dynatrace

Hybrid app instrumentation

Native

- Rich user experience
- Platform specific
- Proven path for mobile apps

Native

- Rich user experience
- Platform specific
- Proven path for mobile apps

HTML5

- Fast development cycles
- Cross-platform
- Instant updates

Native

- Rich user experience
- Platform specific
- Proven path for mobile apps

Hybrid

- App-like experience
- Leverages device capabilities
- Multiple platforms

HTML5

- Fast development cycles
- Cross-platform
- Instant updates

Two agents, two applications, one session

- Define mobile app
  - Auto instrument the app
- Define agentless web application
  - Download the inline JS code and place in your apps assets
  - Insert JS tag into all HTML files
- Every agent reports data for its own application
- Joined user session

➔The Dynatrace Cordova Plugin on npmjs.com takes care of everything

Lesson #6 – Instrumenting an Ionic app

- Get familiar with the Ionic CLI and run the app in the emulator
- Use the Dynatrace Cordova plugin to instrument the app
- Analyze the data in Dyntrace

Mobile application settings

test (20)

General
See your application setup

Instrumentation
See how to instrument your application

User actions
Generate consistent user action names

Instrument your mobile application

Please select the platform you want to use

Android iOS Cordova

Follow the steps to setup your mobile application "Test"

1 Configure the Dynatrace Cordova plugin

The Dynatrace Cordova plugin allows to instrument all apps that are built on top of Apache Cordova. This also includes apps using PhoneGap, Ionic and IBM mobile first and others. You can install the Cordova plugin using npm:

```
cordova plugin add dynatrace-cordova-plugin –save
```

Copy

2 Select agentless RUM application

In addition to monitoring the native app, you can also monitor what's happening inside the WebView. You need to select a web application that is configured for agentless monitoring.

You have to select a web application to continue. Choose between an existing web application or create a new one.

Use existing web application

Test Web Application

3 Download config.file to your project

You need to place it in the root of your project, next to config.xml.

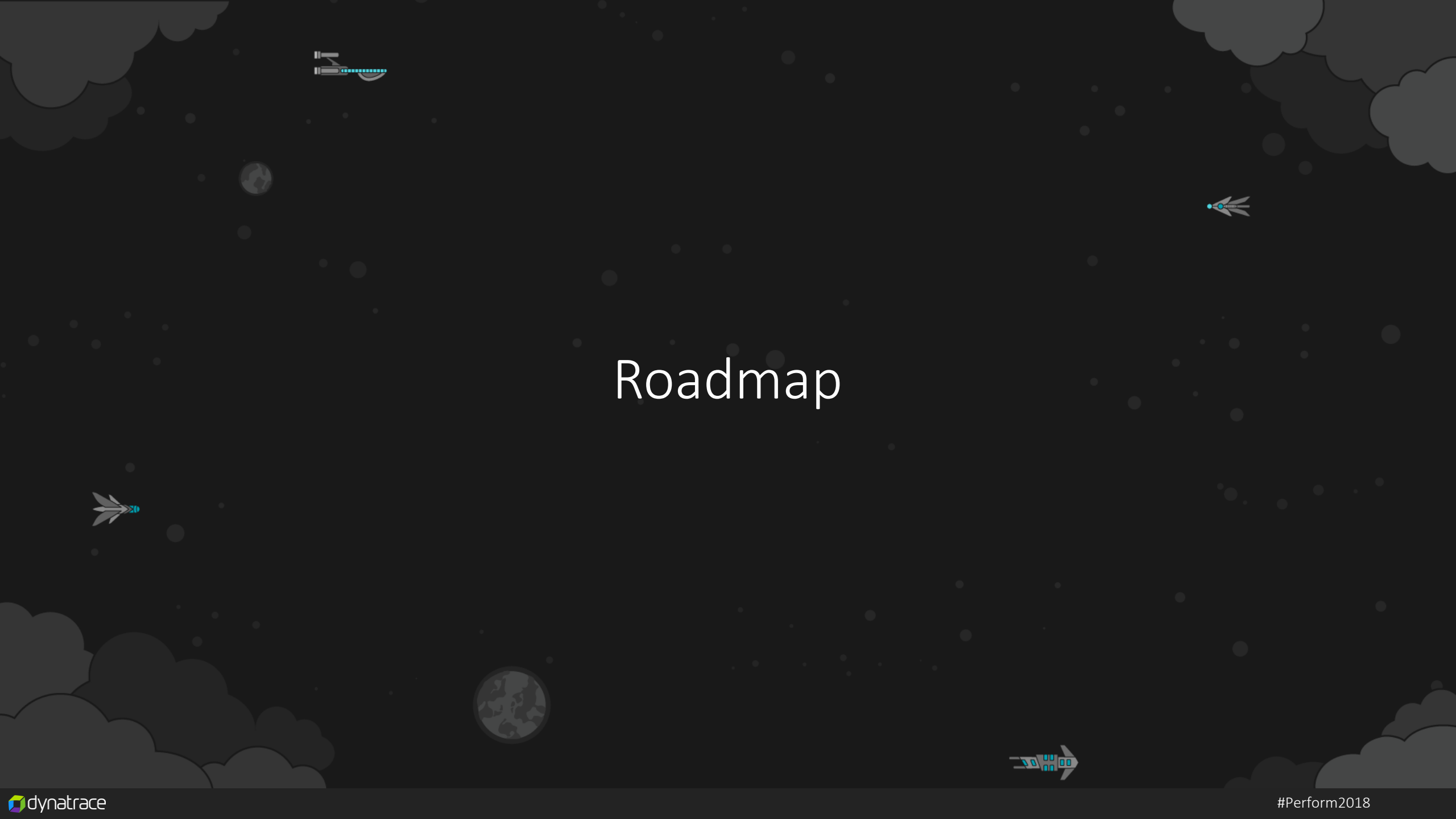
Download dynatrace.config

4 Build your project

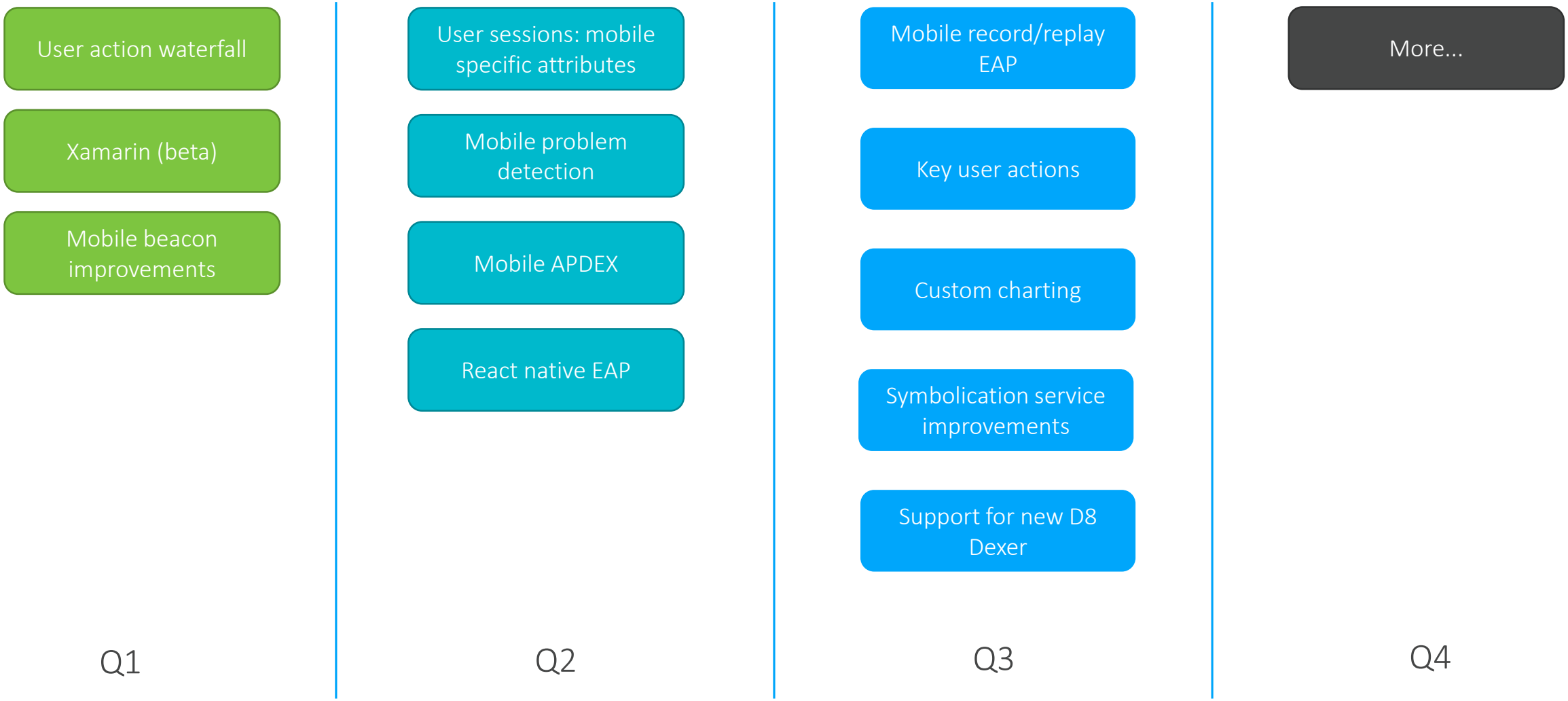
Build your project using **cordova build android** or **cordova build ios**.

What about other hybrid environments?

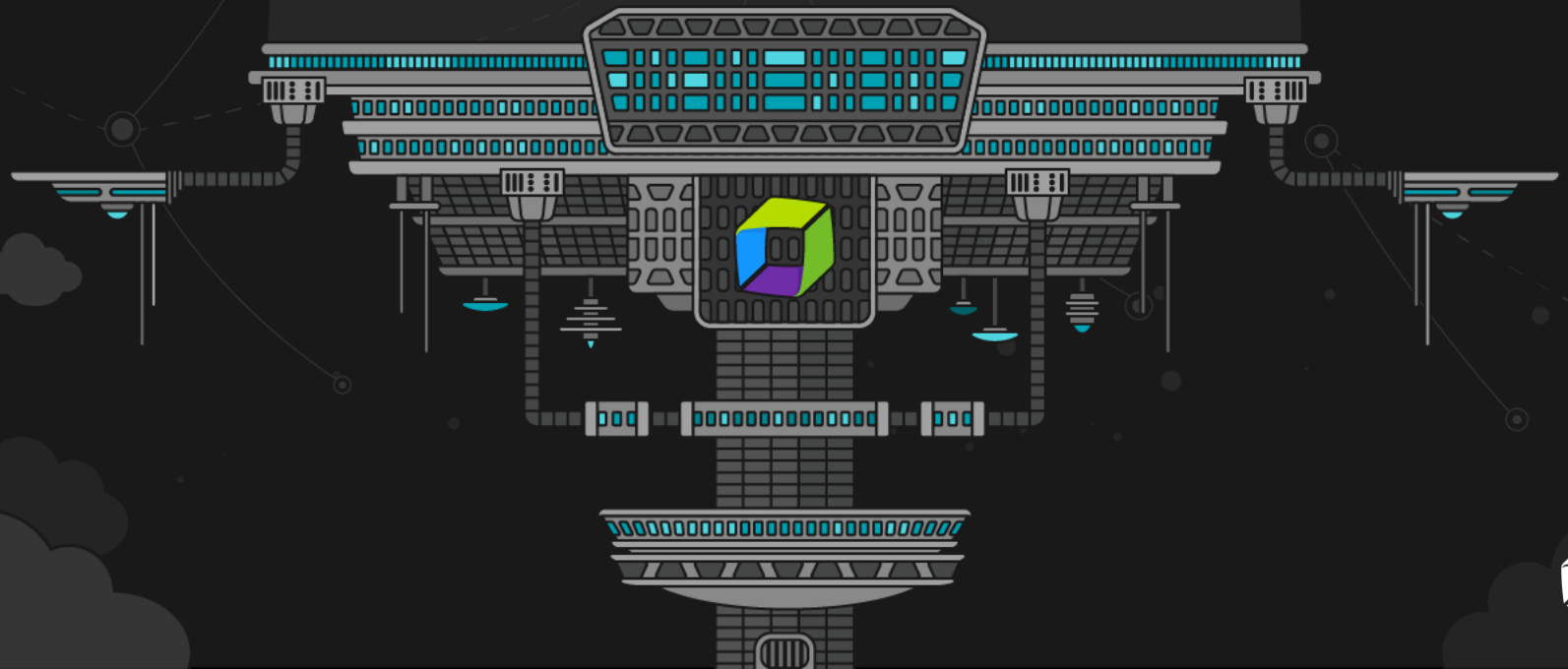
- Webview based (supported via Cordova plugin)
  - Cordova
  - PhoneGap
  - Ionic
  - IBM Mobile First
- Kony (supported via auto-instrumentation)
- Xamarin (EAP)
  - https://community.dynatrace.com/community/display/PUB/Dynatrace+Xamarin+Component+EAP
- Roadmap
  - React native
  - Native script


Roadmap

Mobile monitoring roadmap



Thank you



dynatrace