DEVICE INTELLIGENCE ACROSS ENVIRONMENTS

TYING WEB AND APP DEVICE DATA TOGETHER

DeviceAtlas®

Contents

1. Introduction

One of the most difficult jobs for companies today is reaching their desired target audiences and measuring engagement in such a dynamic landscape. In fact, the "rapid pace of technological change" is considered the single biggest challenge by 71% of CEOs of the top 500 companies in the world according to Fortune's 500 survey, with "mobile computing" listed as one of the technologies most important to their futures. No doubt this rapid technological change has been driven by the increasing adoption of mobile which has surpassed desktop/laptop usage since 2014 and shows no signs of slowing down.

There has been another important shift within the mobile spectrum in recent years in how users are accessing the Internet. We know that mobile is dominating digital usage but what sort of mobile activity is occuring? Yahoo's Flurry observed that **90% of consumer mobile time is spent in apps**. As they put it: It's an App World. The Web Just Lives in It.

With mobile apps having unstoppable momentum, it's time for businesses to develop effective acquisition and retention strategies for mobile app users and subsequently get a better understanding of app usage behavior. The importance of this lies in the fact that with increased mobile app usage, comes increased expectations. Users want to complete mobile tasks efficiently and for this to happen, optimal mobile experiences and real-time personalized content need to be at the core of this. If not, companies are at risk of losing users, conversions and ROI.

One of the most effective ways of getting an insight on your app users is analyzing device data on actual mobile device in use. By getting deep intelligence on device capabilities and properties in the app environment, you can begin to optimize content in real time, use more sophisticated targeting capabilities and enhance your analytics. It is critical for businesses to understand customer engagement at the device level in every environment to ensure that all your systems are addressing customers optimally.

In this paper we will cover:

- **Device detection: what is it and how is it done?**
- **Use cases of device detection for apps**
- **Cross environment analysis between web and apps**

2. Device detection: what is it and how is it done?

Using an accurate device detection solution is crucial for all businesses who want to optimize user experience for mobile users. If done right, device detection can enable you to optimize content in real-time based on a user's device as well as make data-driven choices on mobile strategy. In a nutshell, device detection is a technique for identifying devices accessing your online content.

Device detection, as a concept, initially focused just on detecting devices of users who were accessing websites. This is done by analyzing the User-Agent (UA) string sent by the browser which informs the web server about the visiting device. The UA string is very useful because it tells you quite specific information about the software and hardware running on the device that is making the request. You can then make decisions on how to handle that device by analyzing the request locally using an API and a locally installed data file, or in the cloud. The trouble is that UA strings aren't consistent in terms of their structure. They contain a lot of keywords, many of which are not related to the browser or the device so it's vital to have a robust solution in place capable of detecting a device accurately.

As mobile apps play more of a key role in every day online activity, there is a need to fill the gap of getting insight on app users and what devices they are using so companies can get a unified view of web and app usage. The difficulty with getting device data in apps is that no UA strings are reported as the request occurs in native apps as opposed to browsers. What you can get is the device manufacturer (e.g Samsung) and device model (e.g SGH-T399N) reported in native apps but this does not itself yield enough insight. Trying to convert the make/model string into programmatically useful data is a costly process if approached as a human task, due to both the landscape diversity and the rate of emergence of new devices, as discussed in more detail in our Mobile Web Reports.

The state of device fragmentation

Increasing device fragmentation can be a big headache for businesses as users can access online content on a myriad of devices including smartphones, tablets, laptops, game consoles, wearables and smart TVs. While the world of smartphones is dominated by iPhones and the Galaxy S family, the number of phone models,phone makers, screen sizes, and screen resolutions out there is immense, making it more difficult for businesses to optimize across all devices.

In our Mobile Web Report we found that there are over 2,000 unique phone models actively used for web browsing in the U.S. alone. These devices are made by 243 different makers, have 79 different screen sizes, 24 different operating systems, and 23 different mobile browsers. This underlines how important companies to have some method of reliably identifying and handling traffic from a diverse device landscape.

The best way to retrieve device data effectively within native apps is to use a device detection solution that can do the grunt work for you and work soundly in the background. Take the make/model string as explained above:

```
samsung SM-N9005
HTC HTC One mini 2
Meizu m2 note
OnePlus A0001
LGE Nexus 4
```

Using an API running on your server side, you can obtain a full set of properties for the identified device.

```
String makeModel = “samsung SM-N9005”;
Properties properties = deviceApi.getProperties(makeModel);
```

The type of properties you can retrieve and analyze for a device are wide-ranging and you can find a full list of available properties here: https://deviceatlas.com/device-data/properties

Category	Example properties
Virtual Properties	DeviceAtlas Device ID, Device Vendor, Device Model
Device Name	Marketing Name, Manufacturer, Year Released
Hardware	Mobile Device, Primary Hardware Type, Touch Screen, Screen Width, Screen Height, Display PPI, Device Pixel Ratio, Screen Color Depth, NFC

3. Use cases of device detection for apps

Today, the average person uses up to 4 different devices, more in some cases. The wider the variety of devices used, the greater the risk is in terms of providing an inadequate user experience which can have major impact on ROI. With this proliferation of connected mobile devices, understanding and handling this diversity in real-time gives companies a strategic advantage. Some of the primary use cases for device detection in the app environment include analytics/reporting and in-app advertising.

1. Analytics and Reporting

Learning how users are interfacing with your mobile app is key aspect of getting a deeper understanding of your audience. Information like number of active users, length of sessions, returned visitors and demographics are all core to this. The more information you have the more data-driven decisions you can make so bringing in an additional layer of device analytics will give you even further insight into your audience. There's a couple of ways in which deep device data can enhance your analytics:

• Addition of device Marketing Name to reporting

As documented in the previous section, Marketing Name is one of the properties you can retrieve when using a device detection solution. While the addition of Marketing Name may seem basic, it can provide value for app publishers in two main areas. Firstly, it is the consumer recognised name for a device (e.g. Galaxy Light), and hence your app usage reports become more easily digestible.

Secondly, the marketing name provides a powerful roll-up parameter. The reason for this is that there can be many model numbers that share a single marketing name. For example, there are over 60 model numbers that make up the Samsung Galaxy S6 and accordingly to understand the population of this device, it is necessary to sum the results for all the model numbers. Using a device detection solution makes this very straightforward to do by grouping results by marketing name.

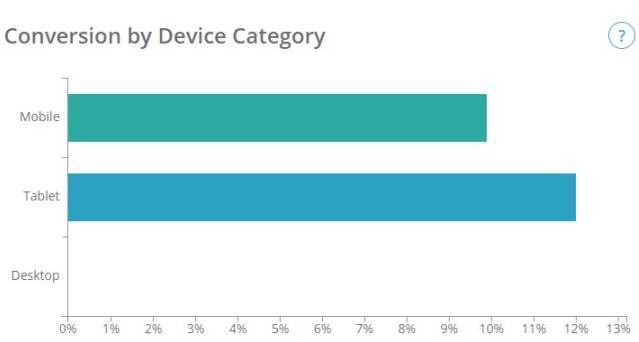
• Device analytics and intelligence

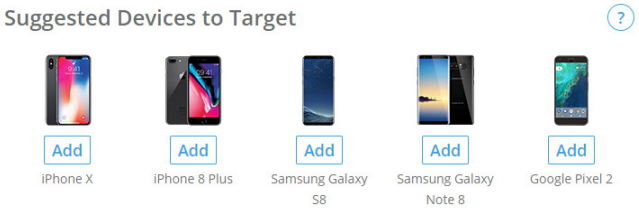
By adding device intelligence to your analytics, you can get a better understanding of user engagement at device level and begin to associate specific behavioural patterns within your app with certain device characteristics - granular reporting at its finest. The availability of a wide range of additional data points about devices permits analysis across diverse variables, yielding deeper insights into app usage patterns.

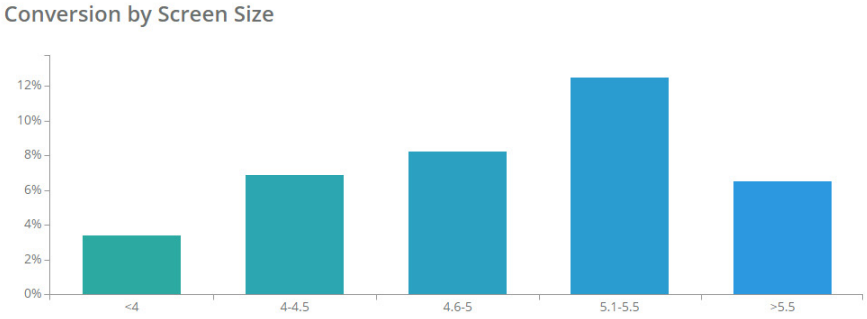
You have the ability to find answers to questions like:

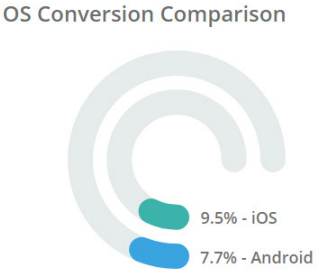
- Does the age of the device (year released) affect conversions?
- Does screen size correlate with time spent in-app?
- Do results vary by brand across nominally similar devices?
- How significant is the impact of device performance (CPU, RAM etc.) on results?

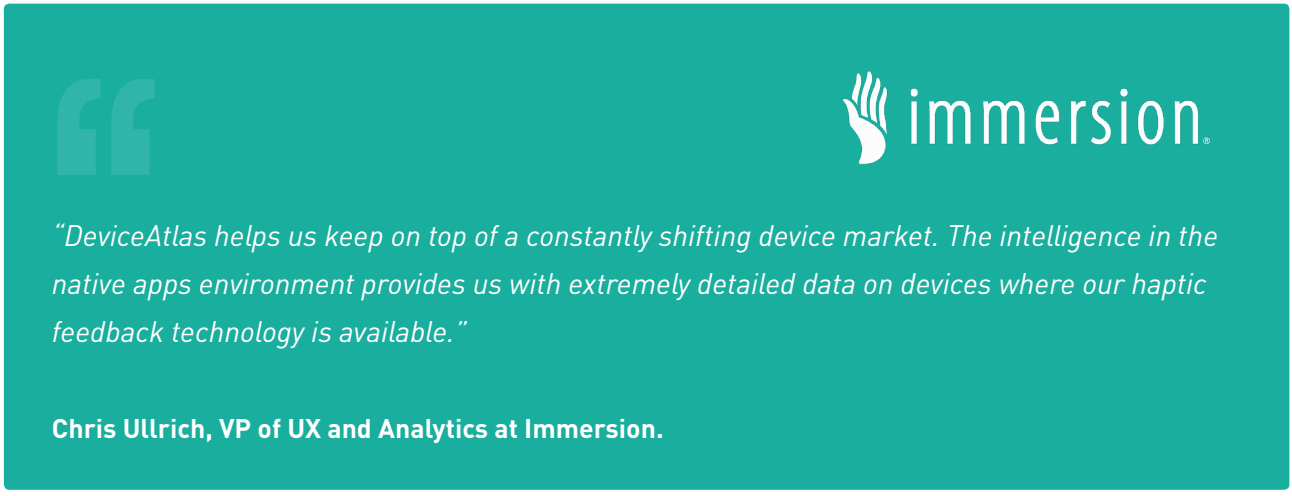
This provides a vital input into delivering the best experience for the device and gives you the power to build custom KPIs and measure success.











2. In-app advertising

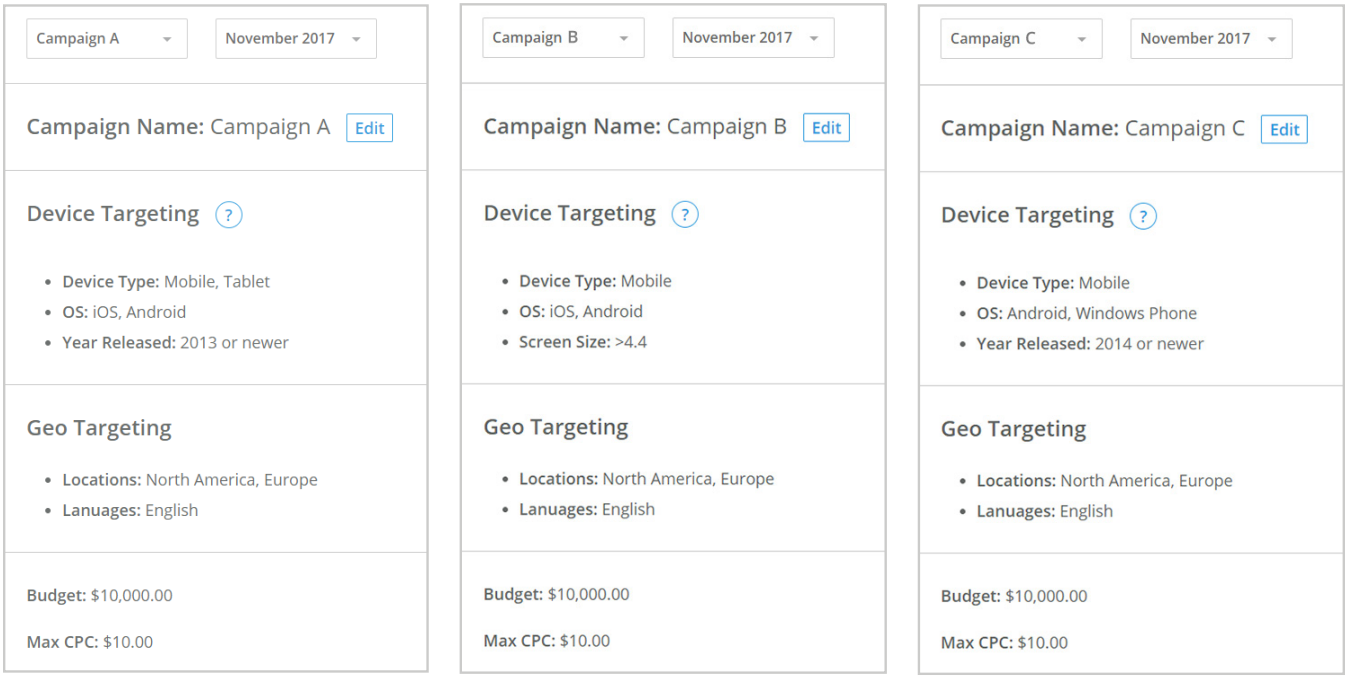
Programmatic advertising is the automated process by which advertisers and publishers buy and sell ad inventory across several channels (including mobile apps) using real-time bidding and is something that will make up 80% of marketing budgets by 2018, according to the IAB.

Device data has a key role to play in the programmatic in-app advertising space. The majority of ad budgets are now spent on carefully targeted rather than wide-reaching campaigns intended for anyone and device detection can play a crucial role here. Not only that but real-time and accurate device data also fits within the advertising remit to measure traffic as accurately as possible and produce consistent reporting.

• Improved campaign management

A device intelligence solution can provide structured inputs that can power campaign management interfaces. Advertisers can then use these inputs to better define their campaigns and enrich their targeting options with various device properties.

For example, listings of OS and browser version, ranked by popularity, as well as listings of device types can be used. The result is that advertisers can reach highly targeted segments at a device level and see which paths lead to conversion.



4. Cross-environment analysis

As mentioned previously, device detection can also be used to recognize devices on the mobile web in real time by analyzing the UA string of the requesting device. With DeviceAtlas' ability to detect connected devices in both the web and app environments, this makes it unique as the only solution to provide intelligence on device usage across both environments (it is also indexed for the mobile network environment).

By using a common device identifier, you can get a full understanding of device traffic across all connected environments. This gives companies a consistent view across channels who can compare results across environments, enabling them to leverage richer reporting and identify characteristics affecting performance.

DeviceAtlas turn these:

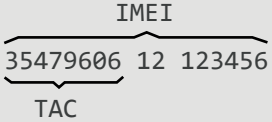
User Agent String(DeviceAtlas for Web)

Example: Mozilla/5.0 (Linux, Android 4.4.2; SGH-T399N Build/KOT49H) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/30.0.0.0 Mobile

Make/model string(DeviceAtlas for Apps)
Example: samsung SGH-T399N

TAC/IMEI (Device Map)

Example:



...into this.

DeviceAtlas Device ID	7536579
Device Vendor	Samsung
Device Model	SGH-T399N
Marketing Name	Galaxy Light
Manufacturer	Samsung
Year Released	2013
Mobile Device	True
Primary Hardware Type	Mobile Phone
Touch Screen	True
Screen Width	480
Screen Height	800
Display PPI	233
Device Pixel Ratio	1.5
Screen Color Depth	24
NFC	False

+150 further data properties

The standard for device detection

DeviceAtlas provides data on all mobile and connected devices including smartphones, tablets, laptops, and wearable devices in real time. The DeviceAtlas API is based on a patented algorithm that quickly and accurately identifies a device visiting a website or app and automatically retrieves the key characteristics of the visiting device from a local data file. DeviceAtlas has a long established track record in delivering technology at Enterprise level and is used extensively as a component in platforms by companies whose products and services need advanced device awareness.

Staying on top of an ever changing device landscape is costly and time consuming, aside from the technical challenges of parsing device information in real-time. DeviceAtlas takes that pain away, so technical teams can concentrate on higher value activities. Not only should your device detection solution be fast and accurate but the memory consumed should be low and predictable so as to not impact the core job at hand, be it serving ads, adapting web pages or analysing traffic. Our API can operate in as little as 15MB of RAM with our data loaded. Even with this tiny footprint, sub-microsecond lookups are achievable on modest hardware.landscape.

5. Conclusion

Increasing device fragmentation can be a big headache for businesses as users can access online content on a myriad of devices including smartphones, tablets, laptops, game consoles, wearables and smart TVs. This development has significantly increased the need to identify how customers are experiencing and consuming content at a device level.

Companies who put in place a strategy to understand how their customers are accessing content and services on mobile and other web enabled devices will enjoy a significant competitive advantage in this increasingly connected environment. By integrating a top class device data source into your website, app or platform, you will get deep, granular information on what devices are accessing online networks and use this intelligence to support key decisions. Those who gamble on "good enough" device detection solutions will risk losing customers, market share and ultimately profitability.