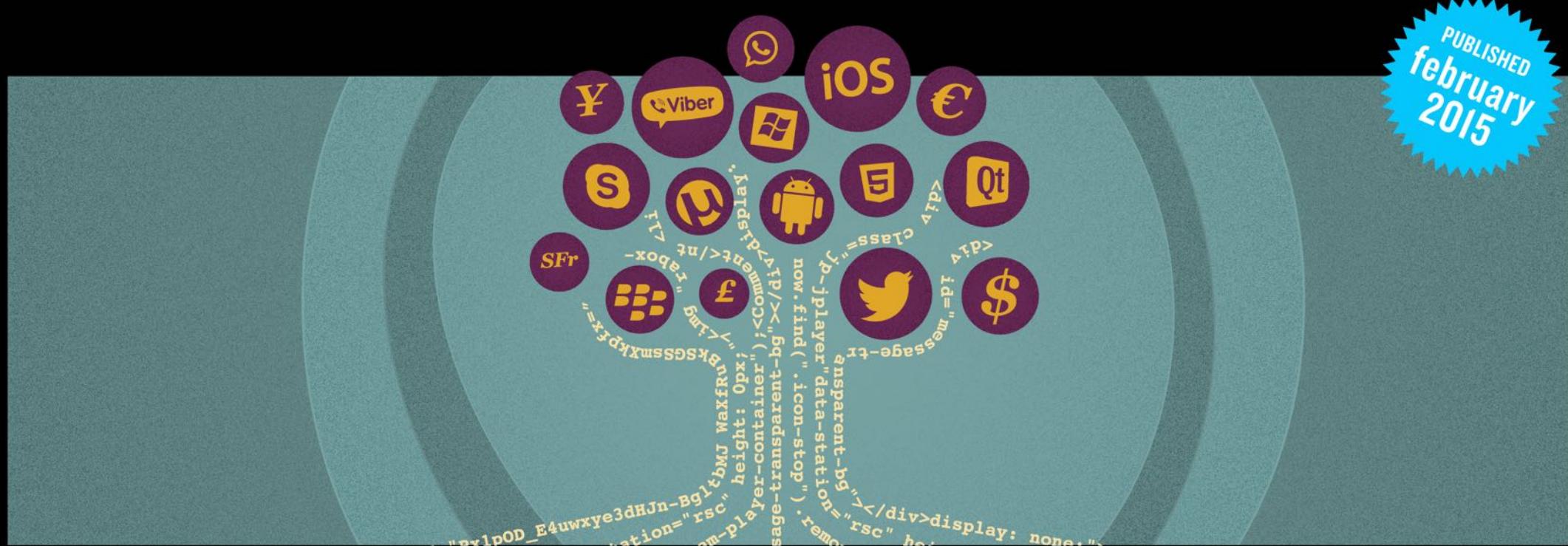


DEVELOPER ECONOMICS

STATE OF THE DEVELOPER NATION Q1 2015

Tracking the latest mobile & IoT developer trends across platforms, regions, and revenues



The 8th edition State of the Nation report features all the latest trends in mobile and IoT development, including platform and language prioritisation, revenues and revenue models, tool adoption and more. It also examines the rise of Swift, the most popular IoT markets, enterprise vs. consumer apps, as well as presenting the most important sources for developer monetisation.

vmob.me/DEIQ15

Supported by:

by GLOBO™

About VisionMobile™

VisionMobile™ is the leading research company in the apps economy and mobile business models. Our research helps clients track developer trends through the largest, most global developer surveys.

Developer Economics is our semi-annual industry research series, tracking app developer trends, attitudes, experiences and monetization by region.

Our mantra: distilling market noise into market sense.

VisionMobile Ltd.
90 Long Acre, Covent Garden,
London WC2E 9RZ
+44 845 003 8742

www.visionmobile.com/blog

Follow us on twitter: @visionmobile

Terms of re-use

1. License Grant.

Subject to the terms and conditions of this License, VisionMobile™ hereby grants you a worldwide, royalty-free, non-exclusive license to reproduce the Report or to incorporate parts of the Report (so long as this is no more than five pages) into one or more documents or publications.

2. Restrictions.

The license granted above is subject to and limited by the following restrictions. You must not distribute the Report on any website or publicly accessible Internet website (such as Dropbox or Slideshare) and you may distribute the Report only under the terms of this License. You may not sublicense the Report. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Report you distribute. If you incorporate parts of the Report (so long as this is no more than five pages) into an adaptation or collection, you must keep intact all copyright, trademark and confidentiality notices for the Report and provide attribution to VisionMobile™ in all distributions, reproductions, adaptations or incorporations which the Report is used (attribution requirement). You must not modify or alter the Report in any way, including providing translations of the Report.

3. Representations, Warranties and Disclaimer

VisionMobile™ believes the statements contained in this publication to be based upon information that we consider reliable, but we do not represent that it is accurate or complete and it should not be relied upon as such. Opinions expressed are current opinions as of the date appearing on this publication only and the information, including the opinions contained herein, are subject to change without notice. Use of this publication by any third party for whatever purpose should not and does not absolve such third party from using due diligence in verifying the publication's contents. VisionMobile disclaims all implied warranties, including, without limitation, warranties of merchantability or fitness for a particular purpose.

4. Limitation on Liability.

VisionMobile™, its affiliates and representatives shall have no liability for any direct, incidental, special, or consequential damages or lost profits, if any, suffered by any third party as a result of decisions made, or not made, or actions taken, or not taken, based on this publication.

5. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by you of the terms of this License.

Contents

1. Stalemate in the Platform Wars?
2. The Rise of Swift
3. App Economy Revenues are Polarising
4. Developing the Internet of Things
5. Not All Tools are Created Equal
6. Enterprise vs. Consumer
7. Platforms appeal to different motivations
8. The App Economy in 2015: e-Commerce Dominates

Also by VisionMobile

Find out more visionmobile.com/reports



[App Economy Forecasts
2014-2017](#)



[North American App
Developer Trends 2014](#)



[Developer Segmentation
Q3 2014](#)



[App Profits and Costs
2014](#)

ABOUT THE AUTHORS

Mark Wilcox
Business Analyst



Mark has over 13 years of experience in mobile software across a variety of roles, however, he got his first computer at 4 years of age and wrote his first program at 7. He will probably always be a developer at heart. A growing interest in economics and business models has led him to work with VisionMobile as a Business Analyst, whilst still keeping his development skills up to date on development projects.

You can reach Mark at:
mark@visionmobile.com
[@_MarkW_](https://twitter.com/_MarkW_)

Christina Voskoglou
Director of Research and Operations



Christina leads the analyst team and oversees all VisionMobile research and data projects (big or small!), from design to methodology, to analysis and insights generation. She is also behind VisionMobile's outcome-based developer segmentation model, as well as the Developer Economics reports and DataBoard subscription services. While at VisionMobile, Christina has led data analysis, survey design and methodology for the ongoing Developer Economics research program, as well as several other primary research projects.

You can reach Christina at:
christina@visionmobile.com
[@ChristinaVoskog](https://twitter.com/ChristinaVoskog)

About Developer Economics

Welcome to the State of the Developer Nation Q1 2015 report, the 8th edition of Developer Economics - the leading research program on mobile developers and the app economy, tracking developer experiences across platforms, revenues, apps, languages, tools, APIs, segments and regions. The Developer Economics program investigates the latest trends in mobile development via developer surveys reaching up to 10,000 app developers in over 130 countries.

This is the 8th edition Developer Economics: State of the Developer Nation report and presents the key findings from the most global developer survey to date with over 8,000 respondents from 143 countries. This research report delves into the key mobile developer trends, as identified in our survey, and discusses platforms, languages, revenues, consumer vs. enterprise, developer tools and segments & more!

The report focuses on eight major themes – each comes with its own infographic:

1. Stalemate in the Platform Wars? - Global mindshare and priorities for full-time professionals
2. The Rise of Swift - Language mindshare and the adoption of Swift
3. App Economy Revenues are Polarising - Revenues by platform and region
4. Developing the Internet of Things - How mobile developers are getting involved
5. Not All Tools are Created Equal - Tool mindshare and platform variations
6. Enterprise vs. Consumer - Categories, revenues and platforms by audience

7. Platforms Appeal to Different Motivations - segment splits by platform
8. The App Economy in 2015: e-Commerce dominates - revenue forecasts

We hope you'll enjoy this report and find the insights useful!

If you have any questions or comments or are looking for additional data, you can get in touch at matos@visionmobile.com. You can also find an online version of our report at www.DeveloperEconomics.com/go

Mark, Christina V, Matos, Alex, Andreas, Dimitris, Vanessa, Chris, Michael, Nick, Stijn and Dinos at VisionMobile.
@visionmobile

Thank you

We'd like to thank everyone who helped us reach 8000+ respondents for our survey, and create this report:

Our Marketing and Research Partners – Globo, Intel, Microsoft and Mozilla.

Our Leading Community and Media Partners, who are too many to number here – you know who you are! Also, the developers and mobile insiders that took the time and interest to share their experiences with us.

COMPANIES WHO CONTRIBUTED TO THIS RESEARCH

Supported by:



Research Partners



Lead Community Partners



Media Partners



PRIORI DATA
Quantify the app economy.



KEY TAKEAWAYS

The 8th Developer Economics survey has once again achieved an industry leading scale, including responses from more than 8,000 app developers across 143 countries. This State of the Developer Nation report brings you the most important findings from the survey, showcasing the latest trends in the platforms, languages and tools that app developers are using and how much money they're making. We also look at how mobile developers are exploring the Internet of Things and take a deep dive into enterprise versus consumer app development.

Platform Wars

The platform wars have ended in a stalemate. Apple has an increasing lock on the high-end with iOS and Android dominates everywhere else. Windows Phone is still growing, now at 30% mindshare, but not generating enough sales to break through the app-gap.

The split of developer platform priorities amongst full time professionals best illustrates the stalemate. Android has 40% of developers, iOS has 37%, whilst Windows Phone and the mobile browser have just 8% and 7% respectively.

With the massive growth of mobile apps it's important to remember that the desktop and mobile web combined is still the most important digital channel for the majority of businesses. The web is definitely not dead.

The Rise of Swift

Our development language rankings show absolutely unprecedented growth for Apple's new Swift language. 20% of mobile developers were using Swift just 4 months after it was introduced to the world.

For comparison, Google's excellent Go language doesn't make it onto our new top chart for server-side programming languages, having reached just 5% mindshare amongst mobile developers after more than 5 years.

23% of Swift adopters were not using Objective C, a sign that Swift may succeed in attracting a much wider range of developers to build native iOS apps.

Revenues

Growth in direct revenues from the app stores is slowing. As these direct revenues are preferred sources of income for the Hobbyists, Explorers and Hunters that make up around 60% of the mobile developer population, competition for them is becoming more intense.

17% of developers who are interested in making money generate no revenue related to apps at all. A further 18% of developers make less than \$100 per month and the next 17%, bringing us to a total of 52%, make less than \$1000 per month.

Of those that prioritise iOS, only 37% are below the app poverty line, making less than \$500 per month on iOS. On the opposite end of the revenue scale, 39% make more than \$5,000 per month on the iOS platform.

The revenue distribution for Android-first developers is not much different than for those targeting BlackBerry 10 or Windows Phone. In fact, developers that go iOS first actually earn much more revenue on Android than those that prioritise the platform.

Internet of Things

Despite the immaturity of IoT platforms, mobile developer interest is high. A massive 53% of mobile developers in our survey were already working on some kind of IoT project.

Smart Home was the most popular market with 37% of mobile developers working on IoT projects targeting it. Wearables were a close second with 35% mindshare.

The majority of mobile developers involved in IoT development are doing it as a hobby (30% involved at this level) or side project (just under 20%), whilst working on mobile apps in their day job.

Tools

Tool awareness is increasing. The fraction of developers not using any third party tools at all has fallen to an all time low of 17%.

The second most popular category of tool is ad networks, with a 31% adoption rate. Unfortunately this is the one category of tool that's negatively correlated with revenues.

Cross-platform tool adoption is on the rise. The percentage of developers using these tools has grown from 23% to 30% over the last 6 months.

Enterprise vs. Consumer

43% of enterprise app developers make more than \$10K per month versus 19% of consumer app developers reaching the same revenue level.

A typical game is giving a third of gross revenue to the app store provider as a cut of in-app purchases and spending half of what's left on ads to acquire new users.

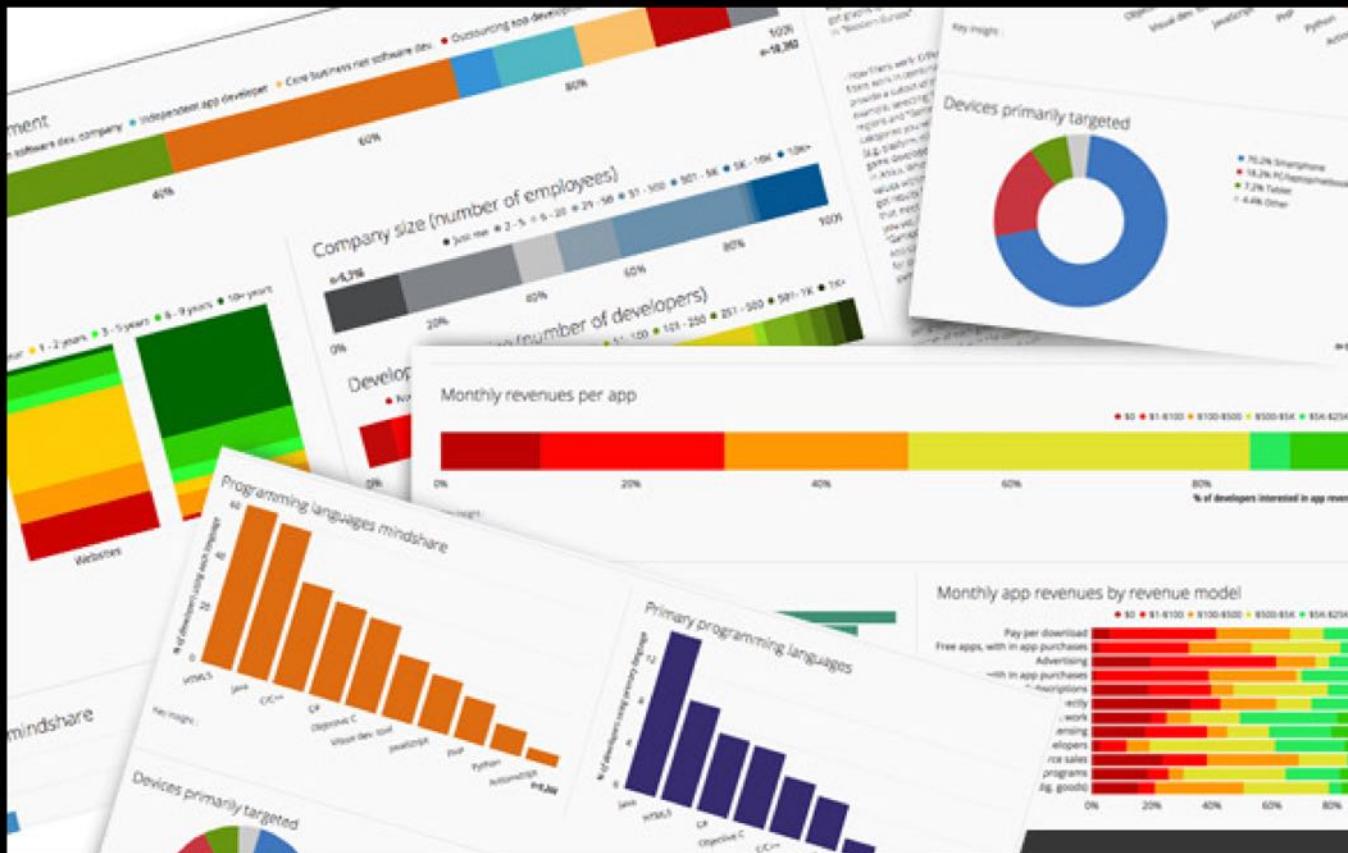
Forecasts

We estimate that there are 5.5 million mobile developers actively maintaining or building publishable apps in 2014.

We predict that mobile e-Commerce will account for 2.5 times as much revenue as all the other sources put together at \$300 billion.

Looking for the data behind this report?

VISIONMOBILE DATABOARD SERVICE



An annual subscription service that allows you to slice and dice thousands of app economy data points, via dashboards and interactive charts

Export graphs in multiple formats to use in your presentations

Download the aggregated data tables behind each graph to use in your analysis

Make sense of the data and leverage the expertise of our analyst team

1 STALEMATE IN THE PLATFORM WARS?

In our last State of the Developer Nation report six months ago, we noted that the platform wars had shifted to the local level, with the battle for local developers and content key to winning a share of device sales in each market.

Since then we've had several signs that the battle may already be effectively over.

Apple owns the high end

At the premium end of the market, Apple is growing their share of device sales. One of the strongest markets for local Android content, China, is Apple's second largest market. Samsung is the main contender in premium Android but Apple finally made the jump to larger screens and removed the Korean giant's largest differentiator.

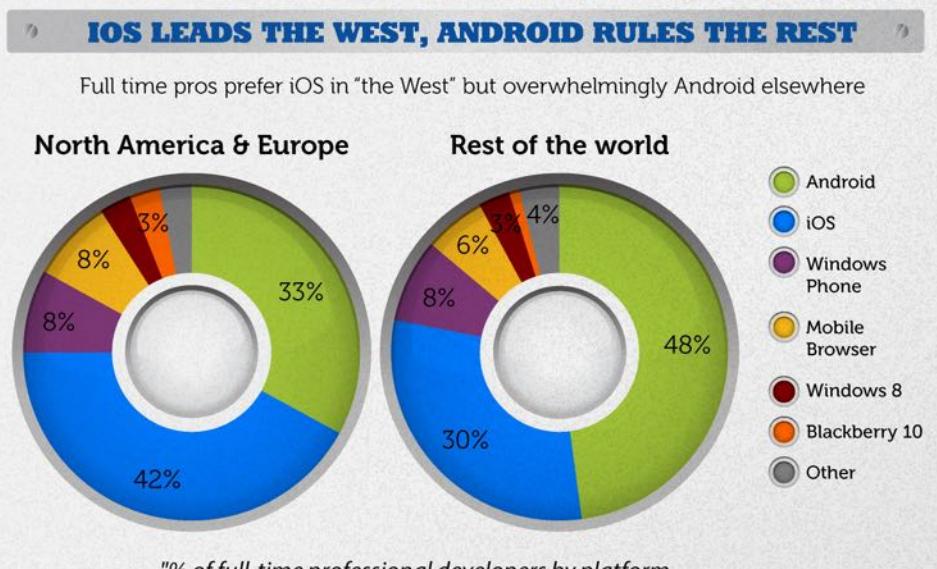
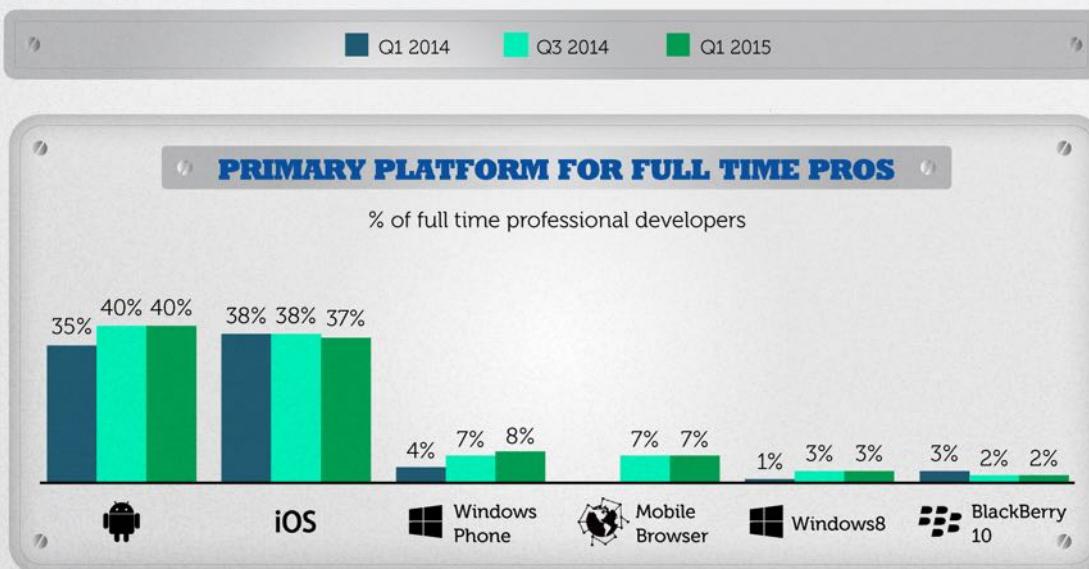
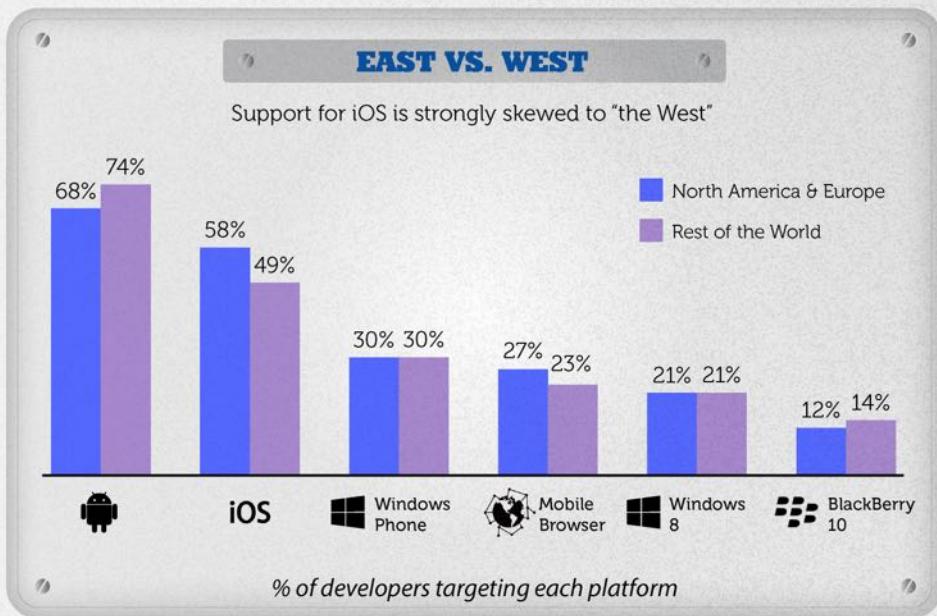
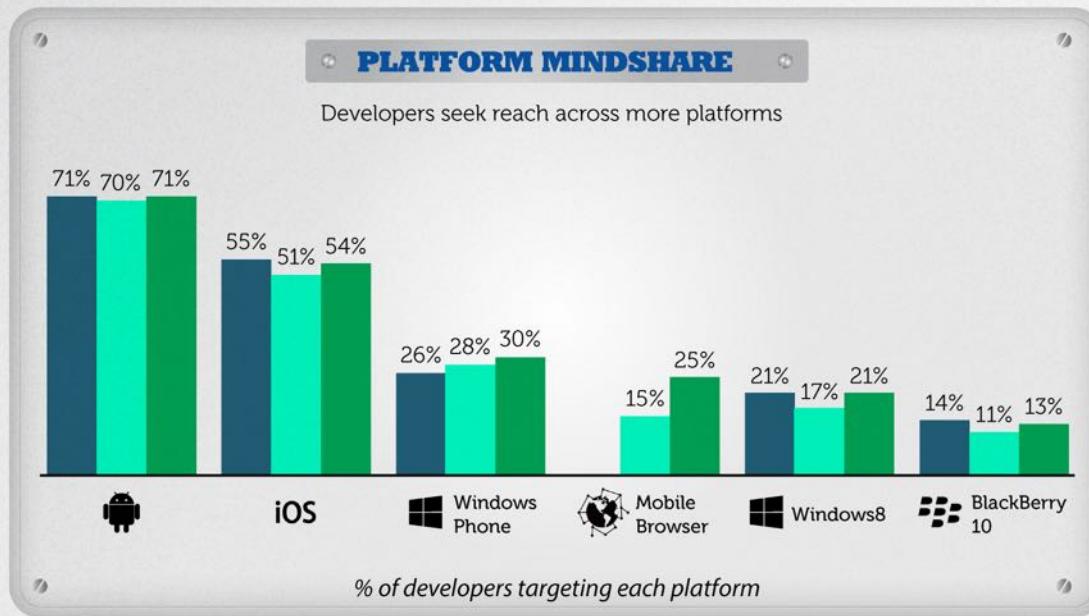
The iOS ecosystem appears to have a lock on the high end that will be hard to break. Slow but steady, iOS developer mindshare declines over recent years have reversed; they're now back up to 54%. iOS also remains the primary platform for 37% of the full-time professional developers, creating the quality apps that keep their ecosystem in front. In North America and Europe, iOS commands a greater mindshare (58%) and by far the largest share of professional developers, with 42% prioritising the platform.

Android commoditising – both good and bad

In the rest of the market, Android will continue to dominate to the point where everyone and their dog owns a smartphone. At the same time, more premium handset makers like Samsung are struggling with competition from local manufacturers. Companies such as Xiaomi and Lenovo in China as well as Micromax in India, are offering better value with the same app ecosystem in the mid-to-low range device segment. This fierce competition helps Android to reach lower price points and thus a wider audience. It's great for users. At the same time inability for manufacturers to differentiate significantly from one another prevents them from competing in the premium market. Even without much of the high-end, Android represents such an enormous global market that it retains 70% developer mindshare and the priority of 40% of full-time professional developers. Outside North America and Western Europe, almost half (48%) of full-time professionals are prioritising the platform and almost three quarters (74%) target it.

STALEMATE IN THE PLATFORM WARS?

iOS owns the premium segment, Android almost everything else, Windows and the browser fight for the scraps



Windows Phone – bigger but not breaking through

Windows Phone is highly differentiated and keeps steadily growing developer mindshare (now at 30% of mobile developers) but is still not making any significant headway with device sales share, which is stuck at 3%. Some major apps, including the top two banks in the US - Chase and Bank of America, have already lost patience and decided to remove their apps from the Microsoft platform.

Meanwhile, some smaller developers on the platform are seeing that there's less competition on Windows Phone. This is particularly true in categories where big brands have left a gap, like music and video. The big brand app gap enables indie developers to get greater visibility and make more revenue than on Android. As such, the number of professional developers prioritising Microsoft's platform continues to climb slightly to 8%. Interestingly, the platform seems to have fairly uniform adoption amongst developers across regions, despite this not being the case amongst users.

Unfortunately, lack of scale means that the top new apps come to the Windows Phone platform quite slowly, if at all, and many others are not updated. This leads to the vicious cycle of self-selection also known as the app-gap – most users who care about apps don't adopt Windows Phone and so in turn the platform isn't as interesting to the top developers. This leaves Microsoft fighting with its latest Lumia models for cost-conscious smartphone buyers who want something different to Android. This market is big enough to sustain a decent sized developer population but is it ever going to be big enough to sustain Microsoft's development of the platform and devices? It currently looks unlikely. If not, how long will Redmond continue to subsidise it? Microsoft is trying once again to leverage their desktop developer base by combining desktop and mobile under Windows 10 but also acknowledging the reality by aggressively building out their services with over 100 apps on Android and iOS.

Stalemate

These factors combined lead to a stalemate in the platform wars. The positions of the platforms are becoming entrenched. Apple cannot move down-market without cannibalising their high-end sales. They're extremely unlikely to risk this, particularly while they are still able to increase their average selling price and grow unit volumes. Android handset makers are increasingly unable to compete effectively for the premium customers. Those are the customers that are most interesting to well-funded developers, as well as advertisers, retailers and various service providers. Meanwhile, a decent Android experience has moved far enough down-market to fend off any potential competition from below. The result is that most developers who are primarily interested in revenue target iOS first and those who prioritise reach over revenue target Android first. In both cases a large fraction of developers also target the other platform as well. Windows Phone is still fighting to build a sustainable niche but faces a vertical mountain climb before challenging the top two.

Will the web developers fight back?

The only possible challenge to the status quo is if cross-platform approaches to building apps begin to dominate. Six months ago,

With the massive growth of mobile apps, it's important to remember that the desktop and mobile web combined is still the most important digital channel for the majority of businesses. The web is definitely not dead.

developer support for the mobile browser was at an all-time low. In this survey we saw a significant inflection point and the mobile browser bounced back strongly, with 25% of developers now supporting it.

This was in conjunction with a reversal of the steady trend towards developers supporting fewer platforms. The average number of platforms developers supported had fallen to 2.2 in our Q3 2014 survey but was back above 2.5 in the latest. As the market matures and is no longer dominated by attempts to monetise apps directly, developers are realising that at least iOS and Android are here to stay, both markets are obviously big enough to be worth supporting. Getting a new product or service discovered and downloaded via an app store is difficult though. Users have some app fatigue; they're not going to download every new thing that comes along to try it out any more. To let users quickly see or try a service on their mobile device, the best way to deliver that experience is through the mobile browser. With the massive growth of mobile apps, it's important to remember that the desktop and mobile web combined is still the most important digital channel for the majority of businesses. The web is definitely not dead.

"Our web projects typically have at least 10x bigger budgets than mobile specific ones. It's difficult to find companies inclined to spend even £100k on building an app, even when that will be returned in a week through additional sales, but plenty happily spend £1 million or even £10 million on their websites."

– Tim Ocock, Executive Technical Director, VML

The web presence for your product is likely to be the first experience users will have with your service. If you have to deliver an experience in the browser and make it as good as possible, then it becomes more tempting to try to leverage that in the native apps. Although web technologies have not historically delivered many great mobile app experiences, they are improving, as are the mobile browsers that host them. The sort of hybrid app technology that emerges may also not follow the traditional web app with native wrapper model. Facebook's newly announced React Native is a glimpse at a possible future where full native app performance can be achieved whilst the UI is declared much more like a web app and the code is mostly written in JavaScript.

2 THE RISE OF SWIFT

Learning a new programming language represents a significant time investment for a developer. It's not as significant as learning the APIs for a new platform but still not something to be taken lightly.

Of course there are some developers that collect programming languages – learning as many as they can for fun, or to improve their understanding of language design and the art of programming itself. These developers are in a small minority though; adoption of a new programming language is a relatively rare event in the life of a typical developer. It's for this reason that it's fascinating to be able to see what happens when, as with the case of Swift, a leading platform introduces a new fully supported language.

Recent comparisons

Fortunately, we don't have to look very far back for comparisons to judge the results by. Objective C has steadily worked its way up to the current 39% mindshare amongst mobile developers over the last 7 years. When the iPhone launched, it was a relatively unpopular language used only on the Mac, which was near an all-time low in popularity. By contrast, Google used Java for Android because it was already one of the most popular programming languages and heavily used in early mobile development. It's noteworthy that even today, less than half (42%) of developers that prioritise iOS are primarily using Objective C. Of the rest, 2% primarily use Swift and the others use some cross-platform or hybrid technology.

Objective C may not have been very popular when the iPhone launched but it was a mature and stable language. It was also the only choice to target the platform at the time. An alternative new language for comparison is Go. Google announced the Go language at the end of 2009. Like Swift, it's a language that's suitable for systems programming (i.e. including low-level, high performance code). However, rather than being designed for building mobile apps, it's ideally suited for developing highly parallel applications on the server side. The kind of thing you might want for a backend for tens or hundreds of millions of mobile clients, or at least to process the data they generate.

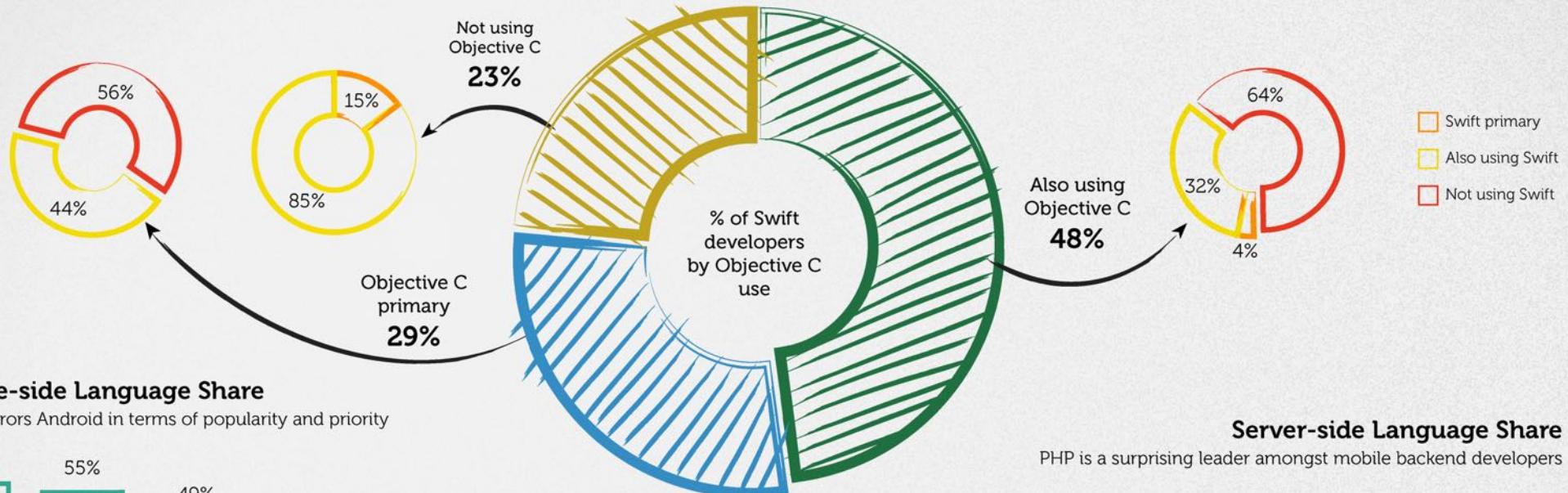
Go doesn't make it onto our chart of top server side programming languages for mobile developers. In fact, after more than 5 years, its mindshare is just 5%.

THE RISE OF SWIFT

Apple's new language jumps immediately to 20% mindshare, attracting more than just the loyal core of Objective C developers

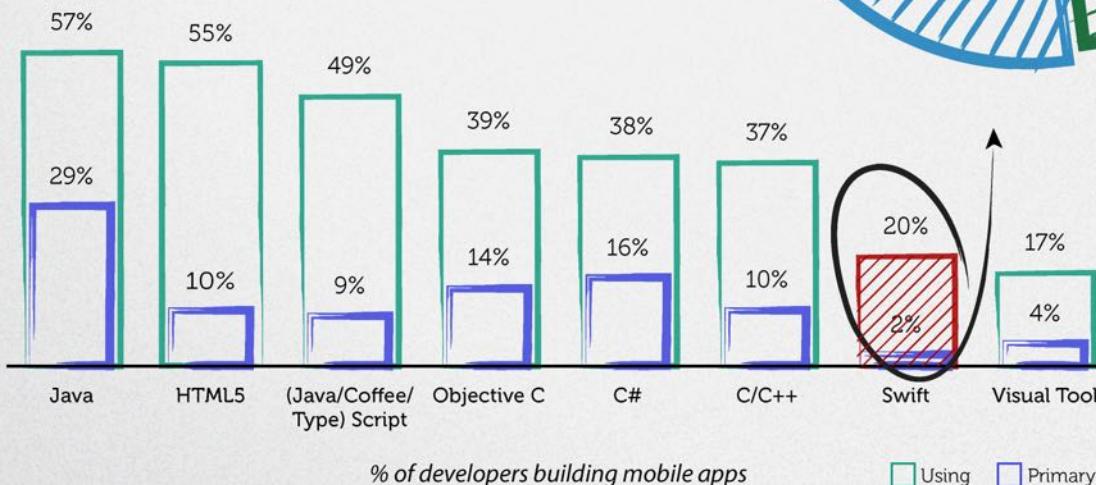
Where the Swift developers are coming from

23% of Swift developers were not previously building true native iOS apps



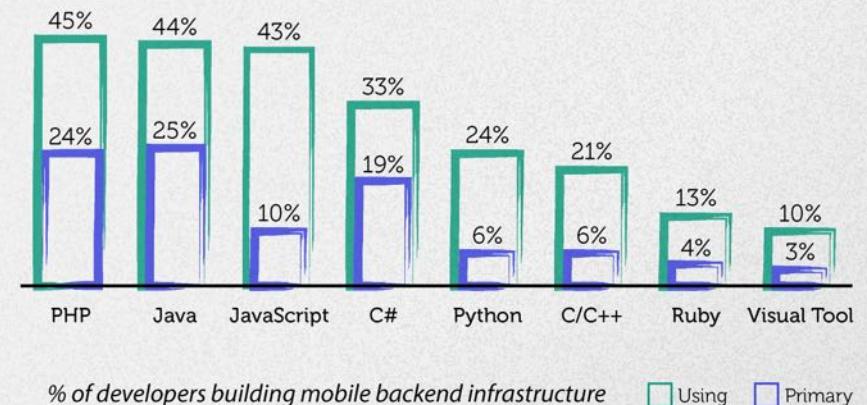
Device-side Language Share

Java mirrors Android in terms of popularity and priority



Server-side Language Share

PHP is a surprising leader amongst mobile backend developers



Unprecedented adoption

Our survey of 8,000+ developers ran just 4 months after Swift was introduced to the world and only 1 month after it was possible to submit an app written with it to the iOS App Store. **With this background it's fair to say that adoption levels are totally unprecedented. 20% of all mobile developers were using Swift and 2% as their primary language.** For a language that's still evolving and for which the tools are not yet mature (Apple just released an updated compiler that crashes a lot less) this is highly remarkable. The fact that Apple clearly signaled that Swift is the future for their platforms obviously helped, and enabling mixing with Objective C makes it very easy to try it out. However, Objective C is not going away any time soon. Objective C was a rather elegant language but with the various additions to modernise it, the syntax was starting to get messy. There was clearly demand from developers for a more modern and flexible language.

Even if we look outside of just mobile development, where iOS clearly makes Swift a big deal, it's clear that no language has seen adoption like this before. RedMonk, who track the popularity of all languages via activity on GitHub and StackOverflow, in their latest ranking update in January had this to say: "the growth that Swift experienced is essentially unprecedented in the history of these rankings. When we see dramatic growth from a language it typically has jumped somewhere between 5 and 10 spots, and the closer the language gets to the Top 20 or within it, the more difficult growth is to come by. And yet Swift has gone from our 68th ranked language during Q3 [2014] to number 22 this quarter, a jump of 46 spots."

Where are the Swift developers coming from?

It's tempting to think that with 14% of developers using Objective C as their primary language, the majority of Swift adopters would come from this core of loyal Apple developers. However, only 44% of those developers are already using Swift, making up just 29% of all Swift

adopters. Many have extensive Objective C experience and projects to deliver. They don't want to take the productivity hit of learning a language without a stable specification or well-established usage patterns. The largest group of Swift adopters, 48% of them, come from the ranks of iOS developers who use Objective C but not as their primary language. The fact that there's a lot of interest in Swift from these developers is a good sign for Apple – it should lead to more truly native apps being created in the future. 23% of Swift adopters were not using Objective C, a sign that Swift may succeed in attracting a much wider range of developers to build native iOS app.

For those new to mobile development, being able to get in early with a major new language must be attractive. A counterpoint to this is that the platform frameworks and a lot of useful third party libraries are already there and written in Objective C. For at least the next few years it seems that practically speaking it'll be necessary to learn both languages to be an accomplished iOS developer.

"We learnt Java and C# in school but I'm teaching myself Swift. I love Apple's products and their platforms are where all the best customers are."

– Ayush Rawat, Student

23% of Swift adopters were not using Objective C, a sign that Swift may succeed in attracting a much wider range of developers to build native iOS apps.

The PHP puzzle

There's another comparison to Swift that's worth looking at in the light of our server side language rankings. The most popular language amongst mobile developers for building their backend

infrastructure is, somewhat surprisingly, PHP. It's surprising because amongst developers that have used anything other than PHP, it's almost unanimously agreed that PHP is not a great language for doing serious engineering. Indeed a quick search for "Why PHP sucks" will produce an endless stream of results, some from highly regarded computer scientists, going back more than a decade. It's only somewhat surprising because PHP is one of the most popular languages in the world. It's what WordPress is written in and that powers more websites than any other content management system – estimated to be more than 20% of the Internet. It's also the language originally used to implement Facebook. However, Facebook ran into problems scaling their systems in PHP and initially built a compiler to convert it to C++, then created a new virtual machine (HHVM) for running PHP code much faster. They also launched a new language last year, Hack, that improves many of the weak points of PHP and adds some modern language features. They've converted a lot of their code to Hack.

Here's the interesting comparison to Swift. Hack is created and backed by a US tech giant, just like Swift. Hack inter-operates with PHP in a similar way to Swift inter-operating with Objective C. Hack has some obvious improvements over PHP as Swift does over Objective C. Hack was launched a few months ahead of Swift, yet we don't see it in our survey and it doesn't show up in the RedMonk or TIOBE programming language rankings either. Why not? PHP is easy to set-up and learn. It makes it very easy to do a number of common web-related tasks. It helps developers to be productive when getting something working quickly. A better PHP sounds like it ought to be a winning proposition. What has Apple done right that Facebook missed? Is it about developer trust, or does Hack just not solve the problems that most PHP developers have? In any case, it's clear that inertia and caution are the standard developer community responses to new languages and the response to Swift is more positive than anything we've seen before.

3 APP ECONOMY REVENUES ARE POLARISING

The app stores are still filling with new apps at an impressive rate, smartphones set new sales records again and app store payouts continue to grow.

However, there has been a shift. The high end of the smartphone market is saturating and tablet sales have fallen slightly. Apple is still growing device volumes but this is now mostly at the expense of premium Android rather than new-to-smartphone users.

The result is that growth in direct revenues from the app stores is slowing. As these direct revenues are preferred sources of income for the Hobbyists, Explorers and Hunters (see chapter 7) that make up around 60% of the mobile developer population, competition for them is becoming more intense. At the same time, the app economy is maturing and larger companies are finding much better ways of making money with apps.

“We initially focused on how to monetise but now we put most of our effort into adding value for users and reaching more of them. It's working.”

– Jim Vitek, Founder & CTO, AppKey

Total revenues are unsustainable for more than 50% of developers

In our previous developer surveys we asked about the amount of revenue earned per app per month. On seeing the staggeringly low levels of income reported, the most common response has been to

question how many apps developers were making to multiply this up to a living wage. To avoid all doubt in our latest survey of more than 8000 developers, we asked for total monthly revenue from all sources related to apps.

The result does not paint an encouraging picture at the bottom end. 17% of developers who are interested in making money (not including those purely doing it for fun or learning purposes) make nothing at all.

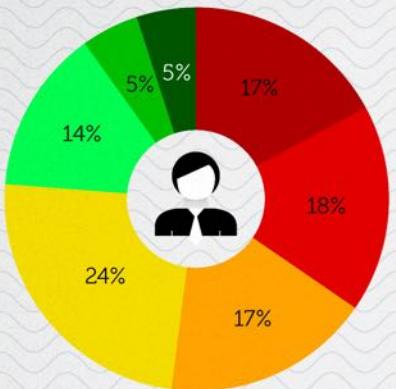
A majority of these are trying and failing to earn some money with a side project, rather than doing this as their full-time occupation. This group also includes those who aren't making any money yet, but presumably don't expect to be. A further 18% of developers make less than \$100 per month and the next 17%, bringing us to a total of 52%, make less than \$1000 per month. This group now includes a lot of fulltime professional developers with mobile apps as their only income. Considering the salary that a mobile developer can earn in most of the world, this does not seem like rational behaviour. Money is clearly not the primary motivator for a lot of these low-income developers.

APP ECONOMY REVENUES ARE POLARISING

An increasing percentage of developers are below the app poverty line, although there are also more big winners

Total monthly developer revenues

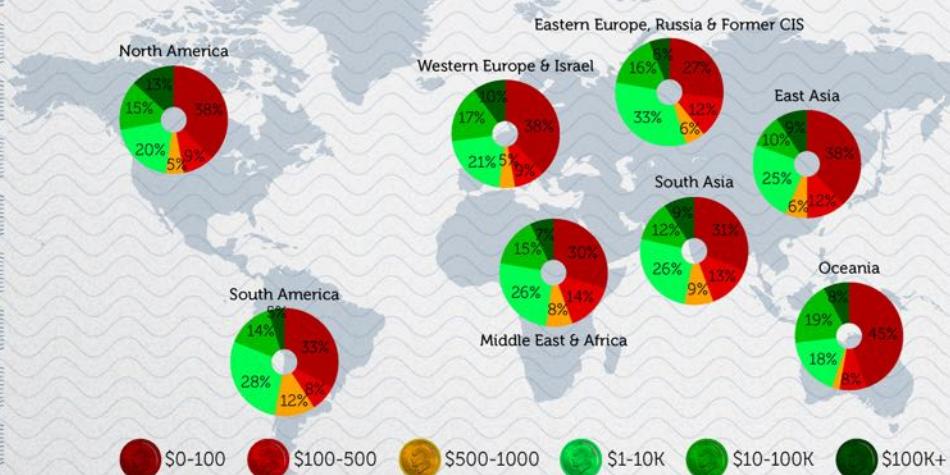
% of developers interested in revenues



Average number of apps each developer is involved in over 12 months: **6**

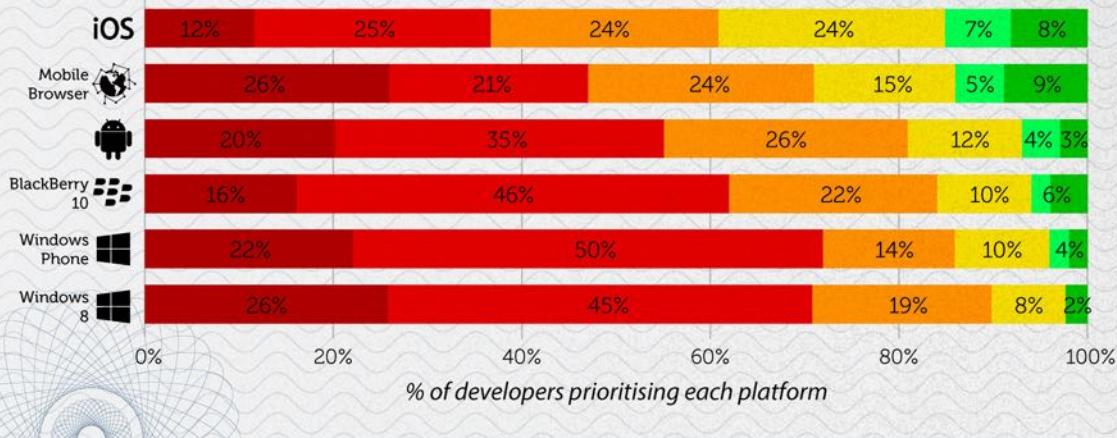
Monthly revenues by region

% of developers interested in revenues in each region



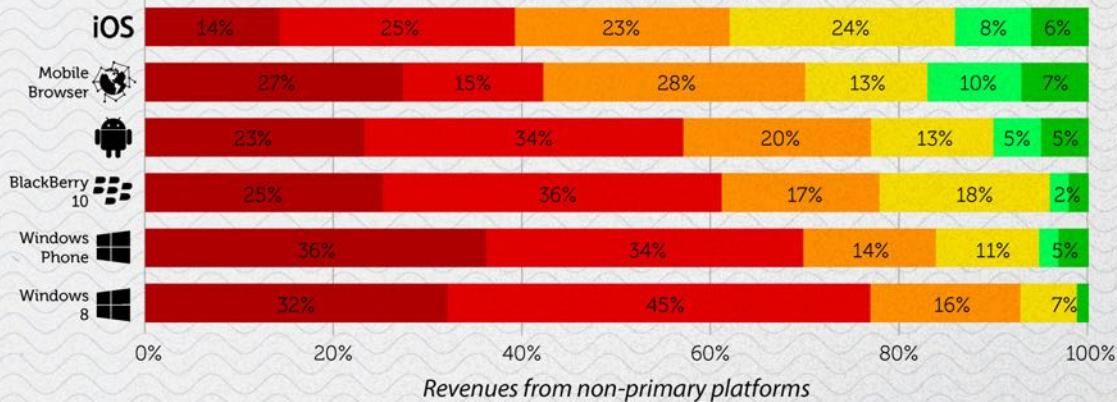
Monthly revenues from primary platform

iOS leads but the Mobile Browser, not Android, is a strong 2nd



Monthly revenues from all other platforms

Developers who prioritise iOS do better on other platforms too



Wide regional variation in the middle-income bracket

Globally, 24% of developers are earning between \$1K per month and \$10K per month. However, in developed markets this group makes up a much smaller percentage of the population. In Oceania, North America and Western Europe plus Israel this is 18%, 20% and 21% of the developer populations respectively. In South America and Eastern Europe, Russia & Former CIS, this is 28% and 33% of the local developer populations. There's no corresponding variation in the \$10K-100K per month bracket above this one, so it isn't just higher living costs in developed countries reflected in higher contract development rates.

In fact, it seems that the developed markets have more developers in app poverty and more big winners, earning more than \$100K per month.

In the more mature markets with higher smartphone penetration, the middle class of small independent app developers is disappearing. This is understandable as they compete with larger and more sophisticated developers for direct revenues from the stores but also for contract work with those in other countries with lower living costs. This is causing revenues to polarise. Developers can either form or join larger organisations to compete for the bigger markets, or find small niches to subsist in. Although there will always be exceptions that thrive as small independents, they're the exception and not the rule.

Revenues are also polarising by platform priority

We have consistently shown that developers who prioritise iOS earn more money than those prioritising Android.

Using our latest data with revenue breakdowns from almost 4,000 developers who reported revenues, we can see which platforms the revenue is earned from. The results show an even bigger revenue gap for Android developers than ever seen before.

Of those that prioritise iOS, only 37% are below the app poverty line, making less than \$500 per month on iOS. On the opposite end of the revenue scale, 39% make more than \$5,000 per month on the iOS platform.

It turns out that Android is not even the second best platform for revenues. Of those prioritising the mobile browser, 47% are below the app poverty line, making less than \$500 and 29% make more than \$5,000 per month. We've previously speculated that those prioritising the mobile browser are more likely to have mature web-based businesses already. Android developers look poor in comparison with 55% earning less than \$500 per month from the platform and just 19% earning more than \$5,000 per month. In fact, the revenue distribution for Android-first developers doesn't look much better than that for those targeting BlackBerry 10 or Windows Phone either. Since the user base for Android is more than an order of magnitude bigger, this is astonishing.

So where are all the Android revenues going? It turns out mostly to those who prioritise iOS and the mobile browser. Developers who prioritise iOS earn far more from their other platforms than anyone else. Most of those iOS developers "other platforms" are just Android, or Android and one other.

So developers that go iOS first actually earn much more revenue on Android than those that prioritise the platform.

This is another aspect of bigger and more sophisticated organisations taking over the market as it matures. Android provided a great open playing field initially but large, serious organisations prioritise iOS because most of them can make more money there. They also target Android and generally out-compete the smaller developers that prioritise that platform.

The flip side to this is that developers who prioritise Android and also target iOS typically don't do as well on either platform.

Speculation that Android's overwhelming market share would eventually tempt top iOS developers to switch their priorities seem to have been unfounded. If anything, the reverse is likely to happen as the app economy continues to mature.

4 DEVELOPING THE INTERNET OF THINGS

The Internet of Things (IoT) is not one product category, market, or technology but very, very many.

The term IoT really refers to the increasing integration of computing and communications technology into everyday objects, from cars to toothbrushes. Adding this digital layer to the physical world over the next decade is almost inevitable due to the ubiquity of smartphones. This is both because the smartphones themselves provide the necessary portable connectivity and computing power to make enhancing individual devices much simpler, but also because the smartphone wars have brought a “peace dividend” in the form of ever smaller and cheaper components.

The IoT's many markets

There are multiple distinct markets where connected devices will be adopted at different rates. Some of the key ones to watch are Smart Homes, Wearables, Retail, Connected Cars, Smart Cities, Medical devices and the Industrial Internet of Things. Most of these markets are immature. There are some early products, a lot of them developed by startups and launched independently on crowdfunding sites like KickStarter. There are systems that have been launched but few or no products yet – the car platforms are a good example – there are some select new models with CarPlay and even fewer with Android Auto. There are subsets of Wearables, like smartwatches that have some products on their second or third generation while the Apple Watch is not yet launched.

The software advantage

Once devices are connected it becomes possible to add value to them in software. This brings the zero-marginal cost economics of software into play in almost every market. Central investment in software or software ecosystems can now improve every individual product. The products with the best software will be the most desirable; hence developers become essential to creating competitive products.

What's better than having high quality software to accompany your product? Having a wide range of high quality software to choose from. Thus the companies with the greatest advantage entering IoT markets are the ones that have a proven ability to build software platforms with vibrant developer ecosystems.

DEVELOPING THE INTERNET OF THINGS

More than half of all mobile developers are already exploring the Internet of Things



53% of mobile developers are working on IoT projects

The **majority** of those involved are exploring the Internet of Things as a hobby or side project

Involvement in IoT development

As a hobby

As a side project
(this is not my main occupation)

As an independent developer (this is my main occupation)

I work for a company and this is their main business

My organisation does this development internally but it's not their core business

My organisation outsources this development to third parties

We provide tools or services to these developers

Not involved



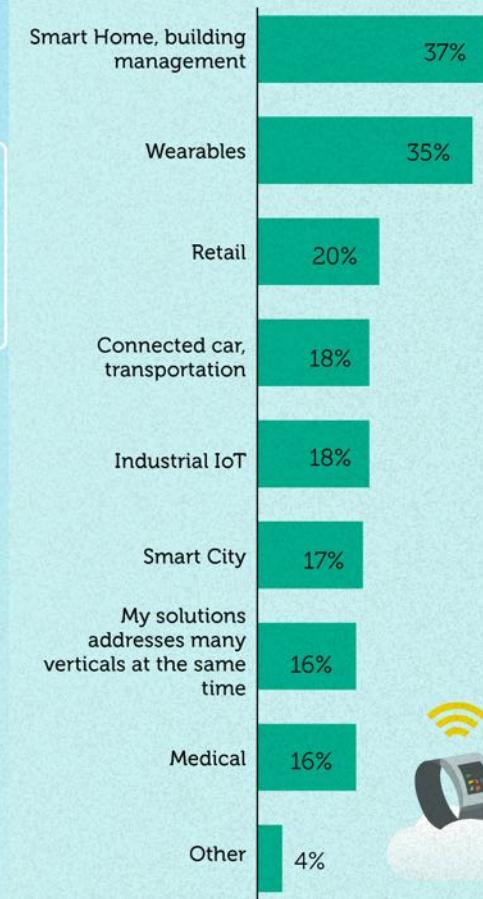
Involvement in mobile development

High % of mobile developers

Low % of mobile developers

Popularity of IoT markets

Smart Homes and wearables by far the most targeted



Smartphone platforms bid for dominance

The major smartphone players are making their bids for several IoT markets by extending their existing software ecosystems. This enables developers to build apps that work across multiple devices, or devices that work with multiple apps.

In the Smart Home vertical we have Samsung's acquisition of early market leader SmartThings versus Apple's HomeKit and Google's acquisition of Nest. For many Wearables and eventually medical devices we have Apple's HealthKit, Google Fit and Samsung's S Health plus Apple Watch, Android Wear and Samsung Gear. In retail, Apple launched iBeacons and ApplePay, with a strong probability of more to come. We also have Google's Android Auto and Apple's CarPlay vying for a central role in the Connected Car. In almost every market there are incumbent owned solutions available or in development and usually a host of start-ups trying to become an early leader too. The key advantage that the smartphone platforms have is the attention and momentum of their existing developer ecosystems. If you're a developer, or a connected device manufacturer, do you bet on Apple and Google's offerings succeeding, or some 10-person start-up with venture capital?

Mobile developers follow the platform's lead

Despite the immaturity of IoT platforms, mobile developer interest is high. A massive 53% of mobile developers in our survey were already working on some kind of IoT project. The most popular markets for those developers to target are the ones where the smartphone platforms have their biggest bets. Smart Home was the most popular with 37% of mobile developers working on IoT projects targeting it. Wearables were a close second with 35% mindshare.

"I'm currently experimenting with several types of IoT project, I think there are a lot of opportunities, e.g. point of sale, parking reservations, bus stops, home automation."

– Navjot Singh, Freelance Researcher

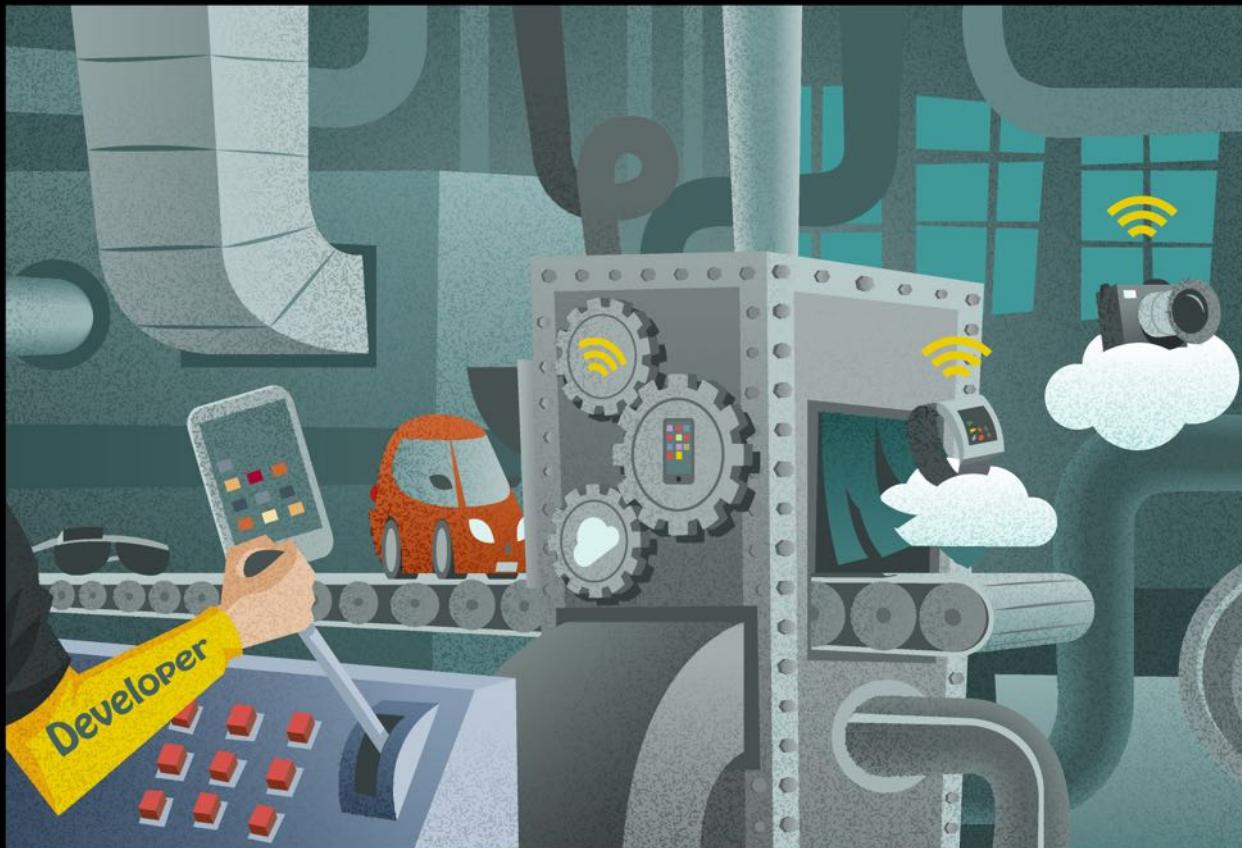
The immaturity of the IoT markets is reflected in the level of involvement developers have with their IoT projects at present. The majority of them are doing IoT development as a hobby (30% involved at this level) or side project (just under 20% here), whilst working on their mobile apps in their day job. The biggest exception to this is amongst developers providing tools or services to IoT developers. As expected with an early market like this, there's a group of developers gearing up to sell picks and shovels to those who come afterwards in search of IoT gold.

Which IoT platforms are developers targeting?

It might also be good to know which platforms developers are targeting within a market, or which ones they plan to target. Perhaps what sort of products they're developing? Or what challenges they face? For more information see our IoT developer trends report at www.vmob.me/IoT15

Looking for IoT trends? Check out our new report!

IOT DEVELOPER TRENDS 2015



4,000+ IoT developers - one report

Which platforms, projects and markets do IoT developers choose, and why? What challenges do they face? What languages do they use and what motivates them?

Plus: mapping the business models of IoT platform vendors to spot future winners.

vmob.me/IoT15

5 NOT ALL TOOLS ARE CREATED EQUAL

As part of our Developer Economics research we track over 900 third-party tools for mobile developers. There are tools to help with almost every facet of a developer's workflow and business.

In our reports, we have consistently highlighted the correlation between tool use and various measures of success for most categories of tool. Sadly the other insight we see consistently in survey after survey is that developer adoption of tools is fairly low across almost all categories. For most types of tool there are options that are completely free, or at least have very generous free tiers that make them effectively free for most apps. This makes it all the more puzzling as to why more tools aren't used. Is it lack of awareness? Maybe it's a reticence to introduce any third party dependencies into their apps? Perhaps most developers just don't think they need third party tools, preferring to build their own or do without?

"How long is it going to be around" and "is it backed by a big corporate" are key considerations in tooling selection."

- Ben Reed, Head of Technology, Mubaloo

Most developers use at least 1 relevant tool

To start on a positive note, the fraction of developers not using any third party tools at all has fallen to an all time low of 17% according to our latest survey of more than 8,000 developers. A lot of those developers are Hobbyists and there's no strong incentive to use third party tools if you're just coding for fun. Another positive is game

development tools. Just over a third of developers build games and 70% of those game developers use a third-party game development tool. Developers building 2D games with Apple's SpriteKit on iOS most likely don't consider that a third party tool since it's part of the platform, so the total percentage of game developers building on a third-party engine rather than re-inventing the wheel is very high. Also, user analytics tools have bounced back to the most popular slot with a decent 47% of developers using them. Why adoption was so much lower in our last survey? It surely can't all have been the ever-changing device ID rules on iOS at the time, although iOS developers are the most frequent users of this category in our latest survey.

The tragedy of the ads

The second most popular category of tools in the survey is ad networks, with a 31% adoption rate. Unfortunately, this is the one category of tools that's negatively correlated with revenues. There's a phenomenal amount of user attention lavished upon those tiny little smartphone screens and a lot of developers trying to monetise it with ads. This produces almost unlimited inventory for advertisers to buy and, as a result, each ad impression is worth very little indeed.

NOT ALL TOOLS ARE CREATED EQUAL

Tool category adoption and tool selection within a category vary significantly by primary platform

User analytics leads tool usage

% of developers using each tool category

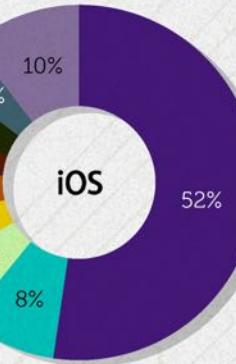
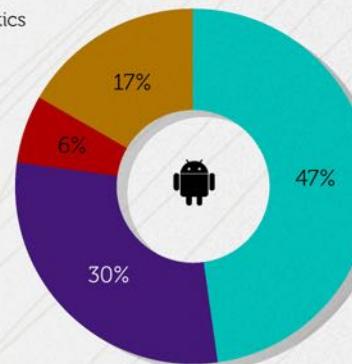


User Analytics is dominated by Google on Android and Flurry on iOS

% of the top 5% of free apps* on Google Play and the iOS app store using each user analytics tool

Legend for User Analytics Tools:

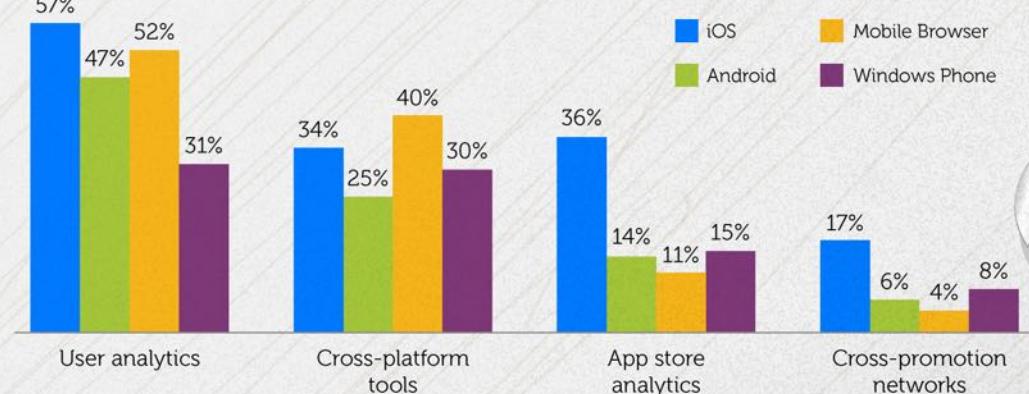
- Google Analytics
- Flurry
- Umeng
- Others



*data gathered by automated scanning of app binaries.

iOS developers are more tool-savvy

% of developers using some tool categories shows variations by primary platform



In order to successfully monetise a mobile app with ads you need an enormous reach and high levels of engagement. Sure, Facebook with more than a billion users spending tens of minutes a day in the app can make more than \$7 billion dollars in a year. Most developers approaching the kind of scale that will make decent money with ads are also big enough to build or buy their own ad technology, rather than use a third-party ad network. Hence the paradox that mobile advertising is a huge market but hardly anyone makes money with ads is solved. Even so, ad networks can provide a decent secondary revenue source as long as they don't detract too much from the experience.

Cross-platform tools on the rise

The percentage of developers using cross-platform tools has grown from 23% to 30% over the last 6 months. Unsurprisingly, developers who prioritise the mobile browser are the most likely to adopt cross-platform tools in order to leverage their web skills building hybrid apps. Xamarin is also increasing in popularity as C# developers need to reach users on iOS and Android. Both web hybrid approaches and Xamarin are increasingly popular with enterprise-focused developers. This has resulted in cross-platform tools moving from being uncorrelated with revenues to having a positive correlation. It's worth noting that this is a correlation though and not causation. Cross-platform approaches tend to have reduced competitiveness for consumer apps. Native enterprise app developers also make more money than cross-platform ones on average. However, there's a lot of demand from enterprises for cross-platform development. The extra revenues available from developing enterprise apps outweigh the native versus cross-platform factor.

iOS developers are more tool-savvy

For most categories of tools, developers that prioritise iOS have higher adoption than those that prioritise other platforms, in some

cases significantly so. App store analytics shows the biggest variation across platforms. This is probably partly because there are more apps making direct revenues from the iOS App Store that are worth tracking across countries and devices. It may also be partly because the interface Apple provided was poor compared to the developer dashboards on other stores. Fortunately for iOS developers, Apple has recently made significant improvements. Another category where iOS developers are the major adopters is cross-promotion networks. Cross-promotion is often used by more sophisticated games publishers. Beyond this, it's not clear why these tools are used so much more on iOS.

Tool selection across platforms

In addition to our survey data, VisionMobile has also developed the technology to detect which tools apps from the iOS App Store and Google Play are using. This gives us more complete data on which tools developers are using most. We can also compare the tools in use by developers of top apps versus others, or look at correlations between tool use and ratings. Looking at this data for user analytics tools shows us that it's not just the level of tool adoption that's platform dependent, it's tool selection too. Amongst the top 5% of free apps on Google Play, just 3 tools make up 83% of user analytics usage with 23 tools making up the remaining 17%. All of the top 3 tools are completely free for the majority of users. A massive 47% share is Google Analytics, possibly not surprising coming from the platform provider. In contrast, iOS user analytics is dominated by Flurry with a 52% share. Another 7 tools have 4% or greater shares with many of these being premium tools. This variation in tooling suggests rather different profiles for the developers of the top apps across the two stores. This kind of data mining in app binaries and metadata across the stores can provide a wealth of information for those with an interest in the developer tools space.



AppZone™
by GLOBO

Cross-platform HTML5
Security Rapid JavaScript
MBaaS CSS IDE

Boost your mobile app development productivity **now!**



Develop



Test



Build



Deploy

Rapid. Cross-platform. MBaaS included.

www.goappzone.com



6 ENTERPRISE VS. CONSUMER

In our Q3 2014 State of the Developer Nation report we suggested enterprise apps would be the next gold rush.

The gold rush is underway with the number of developers primarily targeting enterprises up from 16% to 20% over the last 6 months. Pioneers in the business and productivity apps and tools arenas have already struck gold too. As just a few examples: Microsoft has acquired the Acompli email app and turned it into Outlook for iOS and Android, then recently added the Sunrise calendar app, which will presumably also become part of Outlook on those platforms; Progress Software has acquired Telerik, who have long been successful suppliers of enterprise products and development tools and were early to build strong mobile focussed offerings; Slack, the team chat app with superior mobile apps and sync just announced it has reached \$12 million dollars in Annual Recurring Revenue (ARR) and is currently adding a further \$1 million ARR every 11 days! Slack may be a long way from the valuations of the top consumer messaging apps now but it's only been launched for a year.

Willingness to pay

Slack now has over 500,000 daily active users. This might not seem so big if you consider a consumer chat service like WhatsApp, which very probably has over 500 million daily active users. However, WhatsApp's revenue model is currently* to provide free service for one year and then charge \$0.99 per year after that; there also seem to be a lot of users who don't get charged after a year. In contrast, Slack has a free tier but then starts charging users at \$7 per month, more

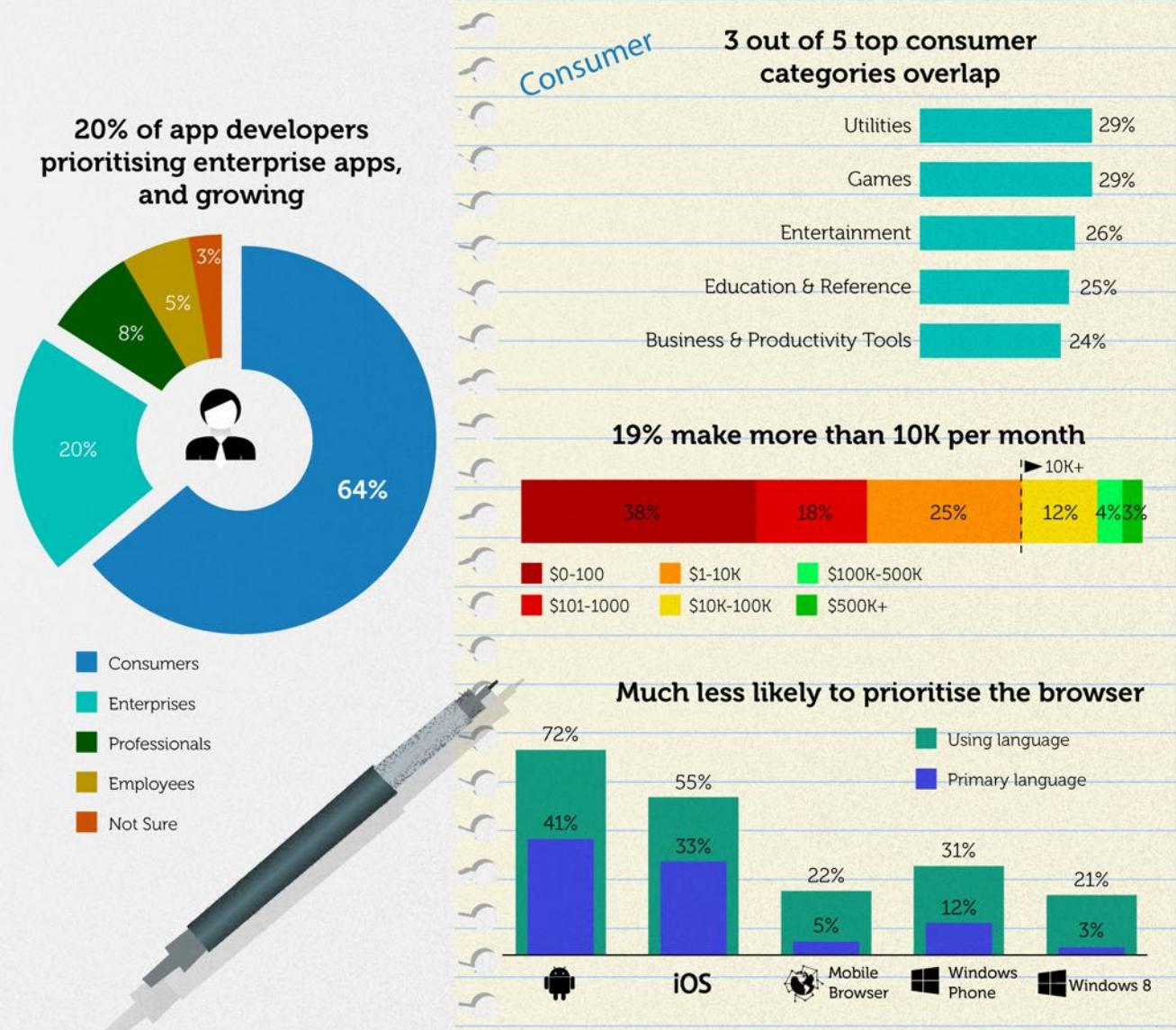
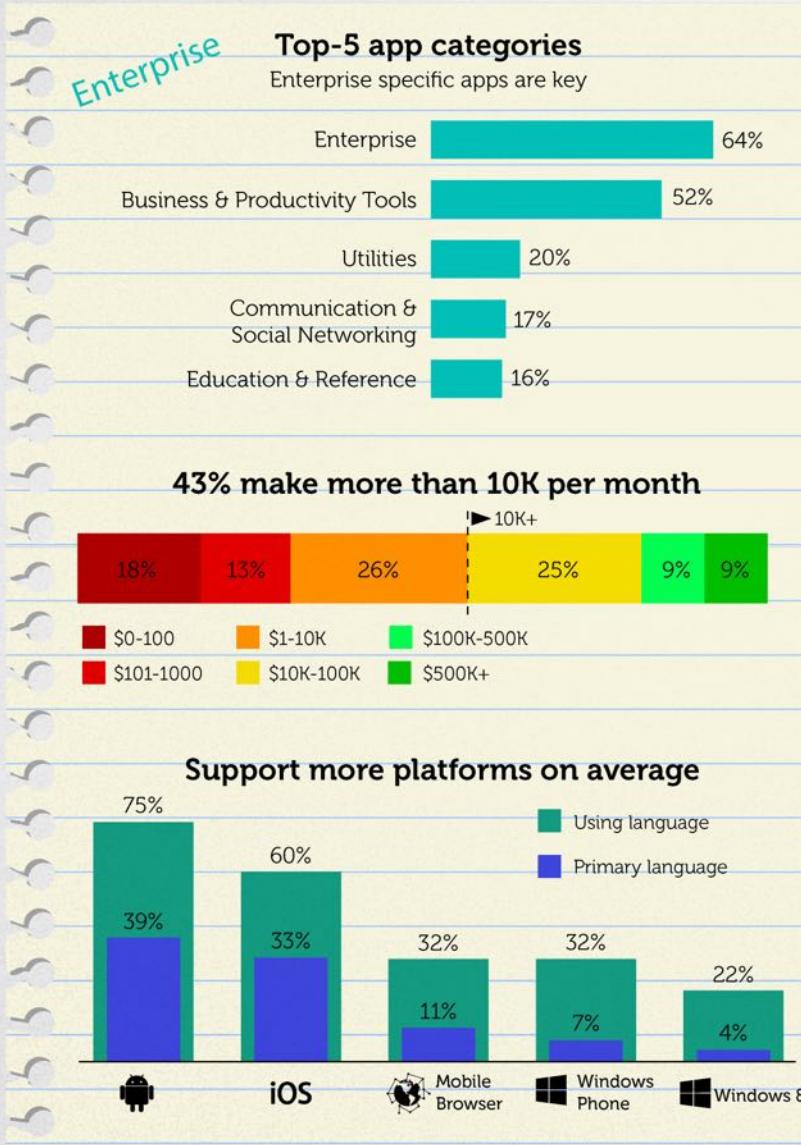
than 100 times as much per year once you take into account the difference in payment processing fees. This is common – businesses are very willing to pay for software that helps them be productive and make money.

It's very hard to convince consumers to part with their cash for software and digital services. We can see this reflected in the revenues of developers who target enterprises versus those who target consumers.

43% of enterprise app developers make more than \$10K per month versus 19% of consumer app developers reaching the same revenue level.

ENTERPRISE VS CONSUMER

An increasing fraction of developers now primarily target enterprises rather than consumers, their tactics differ and they're much more successful



Competition for attention

A common complaint from consumer app developers is that selling to enterprises requires a sales team, which is both expensive and difficult to scale. This is true, although demand for good mobility solutions for enterprises outstrips supply at the moment and really well executed products and services are getting a lot of word-of-mouth marketing. Also, the competition for consumer attention is so fierce that getting a large user base for any app is often prohibitively expensive.

The high-earning consumer apps are mostly free-to-play games. These businesses are now marketing apps much like other fast-moving consumer goods, with advertising budgets to match. A large fraction of Facebook's revenue is coming from mobile app install ads and Twitter, Google and now Pinterest have all rushed to compete for this ad spending. Developers of non-game apps need to generate a high lifetime value from their users to be able to bid for these ad units against the top games. Which poses the question: Is it really that much harder to sell to businesses?

"We spend 50% or more of our revenue on advertising and can predict our revenues 12 months ahead."

– Philip Beck, COO, HappyLatte (Games developer)

* It's very likely that Facebook has some better ideas for making money from the user base without charging them directly – probably by enabling transactions between users and taking a very small cut.

Categories in common

If we look at the app categories that developers for different audiences target, there are obvious differences but also some surprising similarities. It comes as no surprise to see that by far the most common apps built by enterprise developers are enterprise-specific apps. Companies often need software tailored to their existing systems and business processes, such as line-of-business apps or sales tools. This will always be a big part of the market. Similarly games are a very popular category for consumer app developers, as are entertainment apps. Outside of these, 3 of the top 5 categories targeted by enterprise and consumer app developers are the same: Business & Productivity Tools, Utilities and Education & Reference. Developers who are building these types of app for consumers should consider how they might build similar products for businesses. Developers targeting professionals directly also have these common category preferences and should at least consider a way to target their products to companies that employ the professionals, as well as going direct. The remaining developers in our survey are split between those targeting employees in their companies and those who aren't sure who they target. The former group are already serving enterprise needs and the latter group, given that they're interested in making money, have plenty of data here to suggest where they should focus their efforts in the future!

A typical game is giving a third of gross revenue to the app store provider as a cut of in-app purchases and spending half of what's left on ads to acquire new users.

Table stakes

Apart from the need for more direct sales activity discussed above, the main argument against doing enterprise app development is that the technical barrier to entry is higher. Consumer app developers can start on a single platform and see if their idea gets traction with users before committing to other platforms. It's much less common to find a business customer with all of their users on a single platform. Businesses want solutions that work across multiple mobile platforms, including tablets, usually the web and/or desktop computers too.

We can see in our data that enterprise app developers are more likely to support and prioritise all the major platforms, with the exception of Windows Phone. To achieve this, enterprise app developers often turn to cross-platform solutions. They are more than twice as likely to prioritise the mobile browser and also use related cross-platform tools like PhoneGap/Cordova. Also, despite the lack of enthusiasm for Windows Phone, enterprise app developers are more likely to use C# via Xamarin to reach iOS and Android. Even with cross-platform tooling, unless it's a pure contract development strategy, selling to enterprises requires more up-front investment than targeting consumers.

7 PLATFORMS APPEAL TO DIFFERENT MOTIVATIONS

Software really is eating the world. Developers have important skills that make them a very valuable part of the economy.

Only very rarely are individual developers able to realise the true financial value of their own skills; it usually takes an entrepreneur with a vision to organise and focus developer effort in order to create and capture real value. There are a lot of entrepreneurs whose visions require software and so the market for developer jobs is extremely vibrant. Whilst discussing app developer revenues (Chapter 3), tooling (Chapter 5) and enterprise versus consumer audience selection (Chapter 6) we repeatedly question the economic rationality of common developer decisions.

At VisionMobile we've created a developer segmentation model based on motivations and outcomes that provides a very useful tool for understanding developer behaviour and predicting future adoption patterns. If we look at the composition of the developer population targeting each platform based on our developer segments we can shed further light on some of the insights we've already gained in the previous chapters.

The 8 developer segments

We have developed a psychographic segmentation model based upon the Jobs to Be Done methodology, popularized by Harvard Professor Clay Christensen and which constitutes today's cutting edge in consumer segmentation techniques.

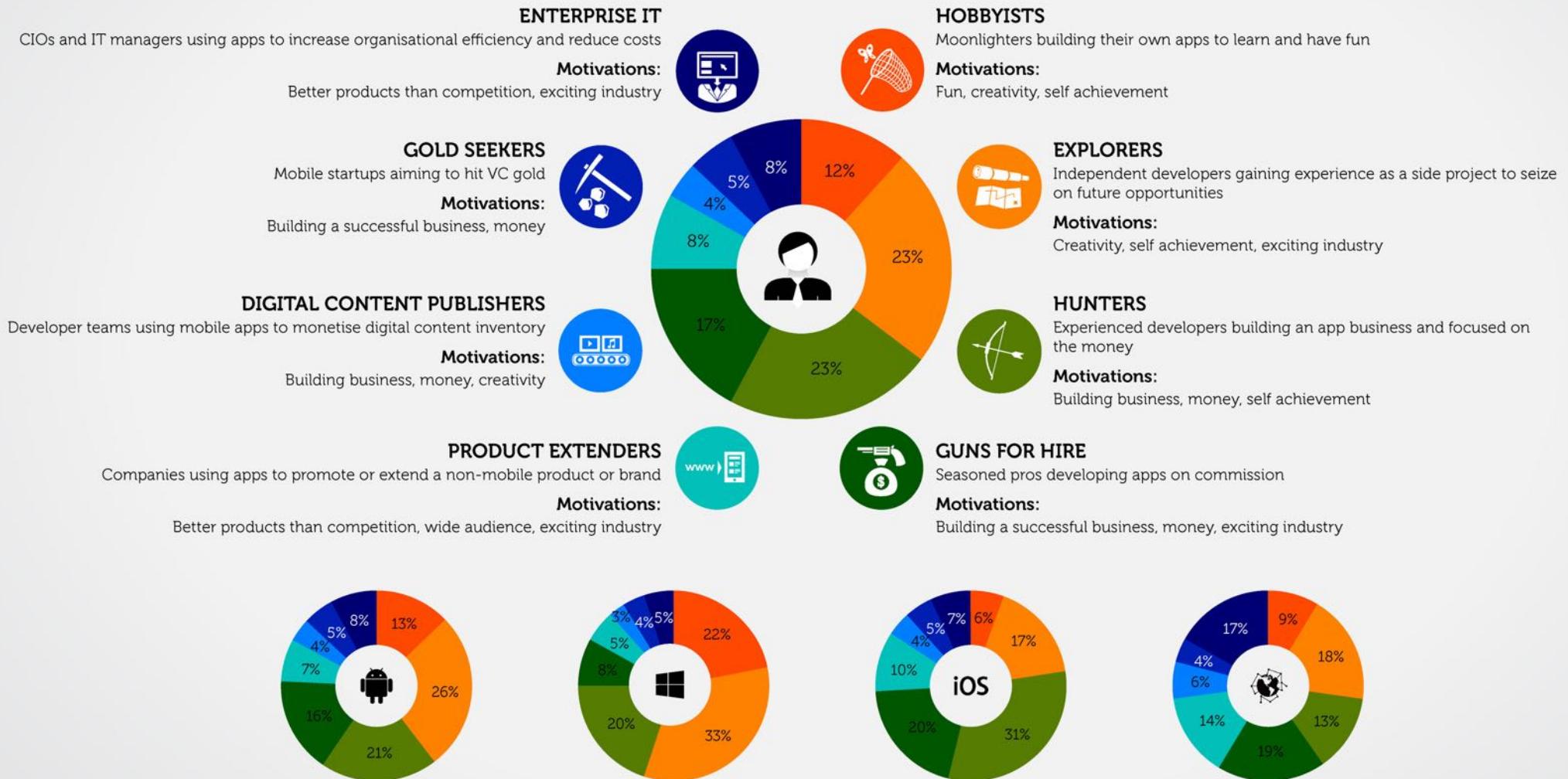
Our developer segmentation model was produced using hard data from over 6,000 developers in our Q3 2013 survey and validated in our Q3 2014 survey with data from over 10,000 developers. The model consists of eight developer segments – the Hobbyists, the Explorers, the Hunters, the Guns for Hire, the Product Extenders, the Digital Content Publishers, the Gold Seekers and the Enterprise IT developers. You can see the definitions of the segments and what motivates them in the infographic, along with how they collectively make up the developer population and their breakdown by platform. These segment sizes have been remarkably stable over the last 18 months as the developer population has grown and individuals migrated back and forth between segments.

PLATFORMS APPEAL TO DIFFERENT MOTIVATIONS

Android has broad appeal, iOS and the Mobile Browser are mostly for serious professionals and Windows Phone is a fun place to start

Developer segments by primary platform

% of all developers



Android - the platform for everyone

The split of developers across segments on Android fairly closely mirrors the developer population as a whole. The only deviations of 2% or more are the Explorers (26%) and the Hunters (21%). Explorers are attracted to Android because it's an easy place to get started. This is both because of the platform language, Java, is one of the most popular in the world outside of mobile development and because setup costs are low. The tools for Android are free and run on almost any desktop computer; a test device can be acquired easily and cheaply and signing up to publish is a \$25 one-off fee. There are presumably slightly fewer Hunters on Android because it offers less of the direct revenues that those developers seek than iOS. Popularity across all other developer segments is still assured because Android retains a dominant market share worldwide and has an installed base of well over 1 billion devices. The greater number of Explorers on Android goes some way to explaining the poor revenue distribution for the platform (see Chapter 3), although it cannot account for all of it.

iOS - the platform for professionals

Unlike Android, iOS has a highly skewed distribution of developer segments in comparison to the total developer population. The Hobbyists on iOS are a rare breed (6%) and there are also far fewer Explorers (17%). Balanced against this, there are many more money-motivated developers, with the Hunters and Guns for Hire together making up 51% of all developers on the platform. Product Extenders are also represented at an above average level.

The lack of Hobbyists and Explorers is easy to understand – Apple actively discourages amateur-looking apps from submitting to the App Store while set-up costs are high on iOS. Where Android is deliberately as approachable as possible, iOS development requires a recent premium desktop computer (Mac only, of course), a premium test device (even the original iPad Mini is still \$249) and a \$99/year

developer program membership that's needed to even test an app on your own device. The greater numbers of Hunters reflect the higher revenues available and more Guns for Hire reflect a contract market where almost every major business wants their app on iOS. The increased proportion of Product Extenders probably just reflects that this split is by primary platform – Product Extenders want reach for their products and target multiple platforms, but more of them want to reach iOS customers first than those on other platforms. A higher percentage of full time professional developers (i.e. not Hobbyists and Explorers) targeting iOS explains some of the better revenue distribution (see Chapter 3) but again, nowhere near all of it.

Windows Phone – the platform for experimenters

The distribution of developers prioritizing Windows Phone is even more skewed from the total population than that for iOS, but in the opposite direction. A massive 55% of Windows Phone-first developers are Hobbyists and Explorers. All other developer segments, are under-represented. The fact that the revenue distribution for Windows Phone-first developers is so close that for Android is all the more encouraging (or damning for Android, depending on your perspective) given this high level of part-time involvement.

That Microsoft has so many developers prioritizing the platform at all is impressive. It's a testament to the strong developer ecosystem they've built around their tools and technologies. Those tools are absolutely first class and many developers used to working with them are unwilling to leave them behind for less polished offerings. This makes Windows Phone an excellent starter platform for those with experience in desktop or server-side development looking to learn about mobile development. There are also many students who've learnt computer science or software engineering using Microsoft's technologies. These are also excellent candidates to get started with Windows Phone. Persuading developers already using other platforms to commit resources to Windows Phone is much harder.

“We like Windows Phone and feel it might help us reach PC users as well, but we don’t plan to adopt before its market share increases.”

– Leon Moodley, Director, Mobiti Interactive

The problem for Microsoft is convincing developers to join, or stay with the Windows Phone ecosystem when they’re developing for mobile devices full time. Our segmentation research shows that when developers migrate to a new segment, they tend to adopt the characteristics of other developers in that segment, including platform preferences. It is a better strategic option for Microsoft to try to retain these developers in their technology ecosystem via Xamarin, or similar cross-platform tooling than to try to keep them on Windows Phone.

Mobile Browser – the platform for reach

The distribution of developers that prioritize the mobile browser reflects the maturity of the web platform. Although there are far fewer than average developers in the Hobbyist, Explorer, and Hunter segments, there are slightly more Guns for Hire and Digital Content Publishers and very many more Product Extenders and Enterprise IT. In the case of Guns for Hire there are just a phenomenal number of websites and web apps out there that need updating to support mobile devices.

Since almost every business has a presence on the web, it’s natural that most of them don’t have in-house developers. Digital Content Publishers are basically what the web was designed for – reaching as many users as possible with your content. While content discovery may happen less on the web these days it’s still mostly consumed via the web, even if that’s an in-app browser embedded in a social networking app rather than the standard platform browser. Product Extenders prioritizing the mobile browser typically have existing web-based products originally built for desktop browsers, adding mobile access just extends their reach (rather than functionality). Finally, Enterprise IT often needs reach for a different reason; they need to reach the broadest range of devices rather than users, so that employees can be empowered to use their services on any device.

The various platforms strengths and weaknesses appeal to different developer motivations and thus influence selection criteria. This is a tiny fraction of what we can explain about developer behavior with our segmentation research. If you need to understand how developers differing goals and motivations influence how they act in the app economy then you should find our Developer Segmentation 2014 report very valuable.

Looking for more on segmentation? Check out our report

DEVELOPER SEGMENTATION Q3 2014



The 8 types of app developers, what motivates them, and how to engage them

An outcome-based segmentation model proven to work in explaining developer choices and investments

In-depth analysis and detailed scorecards for each of the eight app developer segments

vmob.me/DS3Q14

8 THE APP ECONOMY IN 2015: E-COMMERCE DOMINATES

For all the media attention focused on the app stores and the revenue generated from them, they are far from the biggest opportunity in the app economy.

In 2014 mobile e-Commerce generated significantly more revenue than all the other revenue models combined and we forecast that in 2015 that gap is only going to widen.

We predict that mobile e-Commerce will account for 2.5 times as much revenue as all the other sources put together at \$300 billion.

Although app store sales are still going to be second in total revenues for 2015 at \$40.5 billion, they'll be one of the slowest growing revenue models. Mobile advertising (from which we exclude search engine ads) will be growing more rapidly to reach \$34 billion, while subscriptions grow the fastest of all, albeit from a relatively low base, to reach \$9 billion. We expect contract development revenues to total \$18.5 billion for the year and all other revenue models (including royalties, affiliate schemes and selling services to developers) to add another \$18 billion between them.

The app economy revenue figures in this chapter come from our App Economy Forecasts 2014-2017 report. As the title suggests, the report also forecasts these revenues by revenue model out to 2017.

It also estimates the size of the developer population and forecasts its growth over that time period, providing a breakdown by region and platform.

Counting the costs

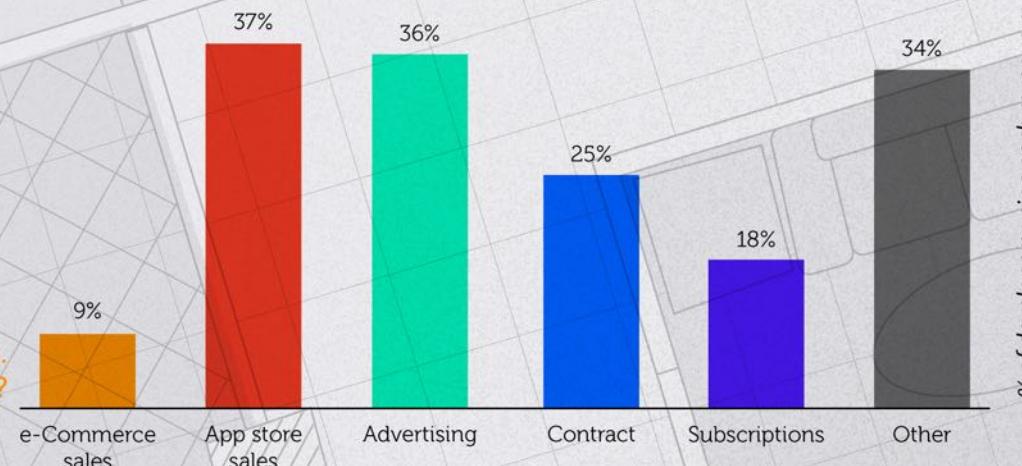
Revenues don't always mean profits. It's certainly true that e-Commerce gross margins are typically lower than other software businesses. However, if we look at the other opportunities in more detail, e-Commerce still looks like a good bet. A typical online retailer might make 15-30% gross margins. A niche mobile retail app might make more. Even with significant customer acquisition spending, this revenue model will generate more profit than any other model generates revenue overall. The app store sales are before a 30% app store cut is taken and as we saw when discussing consumer apps (Chapter 5) these revenues are mostly generated by free-to-play games that spend at least another 30% of the total on ads to acquire customers. Advertising revenues look attractive but the vast majority of these will be earned by a small number of apps with very high user engagement and their own ad platforms.

THE APP ECONOMY IN 2015: E-COMMERCE DOMINATES

Mobile e-Commerce will account for 2.5 times as much revenue as the rest of the app economy put together in 2015



Share of app developers using each revenue model



Only 9% of developers are building apps with e-Commerce.
So where are all the e-Commerce developers?

Contract revenues do mostly end up directly in the hands of developers although it's very hard to scale a contracting business, as this requires hiring a lot of developers and keeping them busy with projects. From a business owner perspective, developer salaries consume most of the revenue. Subscriptions can generate excellent profits, however, in 2015 the bulk of mobile subscription revenue will be for music and video streaming services. These businesses have massive content costs and subsequently fairly low margins. The small but rapidly growing fraction of mobile subscriptions earned by Software-as-a-Service (SaaS) businesses does have high margins. It's typically quite expensive to acquire customers initially but they tend to be very profitable across their lifetimes.

Resource misallocation

The invisible hand of the market is supposed to allocate resources where they generate the best results for the economy over time. Looking at the revenue models that developers are using versus the ones that will generate the best outcomes we seem to be very far from this ideal at the present time. Despite the enormous revenue opportunity offered by mobile e-Commerce only 9% of developers are using this revenue model. Unless there's a lot of e-Commerce related development being done through other models, such as contract work, or "Other" (e.g. selling licenses to some e-Commerce technology) then there's a big gap in the market here. Additionally, although a lot of developers are chasing a decent total of app store revenues, most of them are competing for the small piece of that pie that isn't generated through free-to-play games.

Similarly with ad revenues, far too many developers are trying to monetize relatively low engagement utilities and games with banner ads or interstitials from an ad network. Although subscription revenues appear to be attracting a decent level of developers for the current revenue opportunity, this is actually the fastest-growing model in the app economy (in percentage terms at least) and should probably be targeted by more developers.

What happens next?

The early years of the app economy produced some inflated expectations for independent developers. The open nature of the app stores and the size of the addressable market made it seem as if anyone who could create a decent app could make their fortune, or at least a decent living writing their own apps. As the market has matured it has become painfully difficult for a small developer to be noticed. Also, the quality and functionality bars have gone up to the point where it's hard for a solo developer or very small team to compete in the consumer app market. Developers that want to stay independent will have to do what they've always done – find profitable niches and serve them really well. Most of the rest will either use their new skills to get jobs building the apps that larger businesses need, or move on to try their luck with Internet of Things.

How will the app economy grow in the next 3 years?

APP ECONOMY FORECASTS 2014-2017



Sizing app stores, developer populations and revenues in the app economy

Current size of the mobile developer population and breakdowns by platforms and revenue models

Sizing revenue models and forecasting the growth by revenue model to 2017

METHODOLOGY

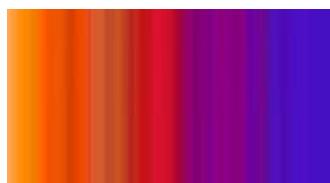
Developer Economics 8th edition reached an impressive 8,000+ respondents from 143 countries around the world. As such, it is the most global ever research on app developers and trends in app development. This report is based on a large-scale online developer survey and one-to-one interviews with mobile app developers. The online survey was designed, produced and carried out by VisionMobile over a period of five weeks between October and November 2014. One-to-one interviews were conducted in December 2014 and January 2015.

Respondents to the online survey came from over 143 countries, including major app development hotspots such as the US, China, India, Israel, UK and Russia and stretching all the way to Kenya, Brazil and Jordan. The geographic reach of this survey is truly reflective of the global scale of the mobile app economy. The online survey was translated in 6 languages (Chinese, Portuguese, Japanese, Korean, Russian, Spanish) and promoted by more than 68 regional and media partners within the app development industry.

To eliminate the effect of regional sampling biases, we weighted the regional distribution across 8 regions by a factor that was determined by the regional distribution and growth trends identified in our App Economy research. The survey gathered responses from developers across mobile platforms including Android, Amazon Fire OS, BlackBerry 10, Firefox OS, iOS, Java ME, Jolla Sailfish, Mobile Browser, Tizen, Windows Phone, Windows 8 and Ubuntu Phone. To minimise the sampling bias for platform distribution across our outreach channels, we weighted the responses to derive a representative platform distribution. We compared the distribution across a number of different developer outreach channels and

identified statistically significant channels that exhibited the lowest variability from the platform medians across our whole sample base. From these channels we excluded the channels of our research partners to eliminate sampling bias due to respondents recruited via these channels. We derived a representative platform distribution based on independent, statistically significant channels to derive a weighted platform distribution.

As we have shown in our Developer Segmentation report, there is no average developer: Our outcome-based segmentation model of eight developer segments shows that the choices and views of developers may vary wildly according to their desired outcomes in the app economy. Hobbyists, who just want to have fun, and Explorers, who are testing the app development grounds, think very differently as compared to professional app developers such as Hunters, who are after direct app revenues, and Product Extenders, who are using apps to promote their non-app products. We have therefore also weighted our results to minimize sampling bias for segment distribution across our outreach channels to derive a representative developer segment distribution. By combining the regional, platform and developer segment weighting we were able to minimise sampling biases due to these factors. All results in the report are weighted by main platform, region and developer segment.



distilling market noise into market sense