The magic of the mobile numbers
Testing with the power of analytics

Marc van 't Veer
Polteq

Table of content

CONTACT INFORMATION

Name and title: MSc. Christian Marc van 't Veer
Date of birth: 22 February 1974
Home town: Delft, Netherlands
Title: Mobile app test consultant
Mob: 0031 (0) 6 46636148

LinkedIN profile: http://lnkd.in/HqsX7z
Website: www.polteq.com
Blog: http://marcvantveer.niobe.nl
Twitter: @marc_vantveer

ABSTRACT

Analytics is an essential skill to become a more efficient and effective tester. I describe with a case the need for analytics, define analytics, explain an analytics model and the multiple feedback loops. In more detail I'll explain how-to validate the recorded data, the use of dashboards and baselines to signal change and formulating smart questions to guide the search for meaningful patterns. Finally, I discuss how telling compelling stories help the development team to make effective decisions.

PROLOGUE

When I started as a mobile tester 5 years ago, the mobile world felt as a special place separate from the rest of the world. A strange place with different rules. Since then, Mobile phones have grown into an extension of our identity, our gateway to the world.[1,2] As Tomi Ahonen explains, mobile is interactive and has unique features like "it's always on", "built-in payment mechanisms", "augmented reality" and the mobile phone is now for years main stream.[3] Many companies are now following the strategy of mobile first[4], but some have gone to that extent where, they are only present on a mobile phone like eBuddy, Layer, Whatsapp, Shazam, Flipboard, and PicNic.[5]

The ideal situation is that, with the use of analytics; organizations can increase their performance and have a competitive advantage. With the use of analytics, a development team can work more efficiently and effectively. As a tester, you're always dealing with the struggle of finding a balance between two opposite forces: *Time to market*[6] and *the loss of credibility or image damage*[7]. *Time to market*[8] is getting the features on the market as quickly as possible. To miss a timeslot is like missing a flight. It gets harder and harder to keep up with the competitors. *The loss of credibility or image damage*[9] is the consequence for a company if a failure is made. It can be compared with a virus outbreak: if a company doesn't know how to respond directly to the spread of bad reviews and negative news, it grows exponentially. Users stop using the app, delete the app, give bad reviews in the app store, the rating goes down, the ranking goes down, users are ranting on social media and complain in their respective networks. As a tester, we can be of equally great value to for see these situations in advance and take preventive measure to mitigate some/all of these issues. With the use of analytics we can give better and faster feedback to the team, which results in a quicker release with less faults.

The reality is that time to market usually supersedes the loss of credibility. Meaning that, the app is not finished when released to the market. It needs to be finished in close harmony with the end users and the dynamics of the market. Try to imagine that, an app or a feature is not yet in production and we have no idea how the app is going to be used or users are using it in a totally different way than expected. Analytics can give insight in the environment of the user. The more we understand the user experience, the better we will be able to create and execute realistic tests for the mobile app.

The difference between the ideal, happy path flow and the realistic case is, as developers we say: "You are always testing the alternative path, no user is ever doing that". The app should be robust and should be of higher quality so it can cope with the big variation in the production environment. Analytics need to be the direct lifeline to the end user, in order to understand what the risks are and to help us decide what kind of tests we need to perform. Discovering patterns is not an easy task compared to recording, collecting and processing raw data into a dashboard. So how can we discover patterns and use analytics in our daily lives? This paper is the reflection of how I think we, as testers, can come closer to that ideal situation.

Structure of the paper
The paper starts with the IONO tour guide app case. This explains the daily reality and gives a common starting point. From a definition of analytics, and a description of the difference with statistics, an analytics model is described. The model explains the multiple feedback loops in the analytics process. Then the problems with a frequently used form of analytics "the most popular devices of the month" are discussed. Then, we go into the different aspects of the analytics model. First, we start with the core of analytics: asking questions. What is a 'good' question and how to formulate good questions to find patterns. All the pattern-finding and decision-making is based on recorded data. The next step is validation of the recorded data. A further step into the model is, the processing of the collected data into a dashboard and define baselines (benchmarks). The last step is one of the hardest part which is, "to

tell the story". Delivering the message and support the call to action and use the results. I finish this paper with my lessons learned from using analytics on a daily basis. But for now lets start with the IONO case.

IONO AND I: NOT A MATCH

"One's destination is never a place, but a new way of seeing things" - Henry Miller[10]

IONO case is an example from my own end user perspective: as a tourist going on a holiday to France. Spoiler alert: it was not a good experience. The mobile app and me: we were not a match. If you like, you can also download the app and I would love to hear the experience you had with this app, see the link in the sources to the IONO app.[11]

This summer I went on a holiday to France, two weeks with my wife and two children. For the first time by car, we avoided Paris and arrived on a beautiful camping site. Swimming pool, kid's playground, bouncy cushions, restaurant, non-English speaking staff, many other kids and nice weather. We were really enjoying this experience. Since we also had some days with heavy rain, we wanted some alternative places where we could go. In France, there are many castles (château's) from royal times. One of the cities we were looking at was Provins. This is on the Unesco list of most beautifully preserved cities from medieval times.[12] We had many brochures/flyers, a bad Wi-Fi connection to explore website information and a whole evening to decide where we could go to.

We looked at the different flyers and found out that most castles were closed on Tuesday. As it was Monday evening this was a natural selection. We still had two castles left: Provins and Château Vaux-le-Vicomte.[13] I looked at Provins and the other one with my wife. This is also how it felt: the other one. So somehow my expectations were high. I found in the Provins flyer, that the city had a website in English with opening hours, ticket pricing, activities, route information and that it was not one castle but a whole city. It was a surprise that, I could download a tour guide app on my smartphone.[11] This made it easier to visit the city with kids, cheaper and with more flexibility. The app is called IONO, which was made by a French company. They knew the cities and their history well.[14] So far so good.

With all the information I discussed above, you may have started to create a picture, a persona, in your head. A persona is a profile of a group of users that is based on real data and validated by real users.[15] Within most mobile projects, I work with multiple persona's. For the IONO app, the personas could be a tour leader, a student and a tourist. For my situation, the persona is a "tourist": It is a male, his name is "Marc", he is 43 years old, a family man, has kids, is married, has a car, likes history, does not speak French, likes to learn and loves city trips. But also has a bad network connection, uses the app during the evenings and uses his mobile phone in other locations than the visiting city. He speaks a different language and from a different time zone (from his home country), has little preparation time, wants to do everything in one day, has limited battery life and has multiple smartphones (iOS and Android).

Issues with the IONO app
So where did it all go wrong with the IONO app? Let's start with languages. Where is this tourist visiting Provins coming from? He is not from France and doesn't speak French. Based on analytics the common countries, regions and popular languages can be determined. The Provins website was available in French, English and Spanish but it redirected to the French App store. The whole app was only in French, no translation or language choice whatsoever was provided to the user. If I wanted to continue

downloading and using the app I needed Google translate to understand how to use it. I think most users would have stopped and deleted the app by this point. This is where Analytics can help by giving information like - under which network conditions the app is used, what is the network type and network speed. This also gives information about max download time. The IONO app is a general app from which many tour guides can be downloaded. The tour guides are placed in a catalogue as a long (very long) list of city names. For Provins there where 17 guides: for every language, but also multiple versions and variants for the same city. There was no explanation about the differences between the versions. A tour guide can easily be a large file to download, because of many pictures, movies and audio files. Why would we not warn the user about the download size of the tours? An individual tour consisted of 12 packages, which were downloaded one by one. After each download it was installed and then the next one was downloaded. If the process was interrupted I had to start all over again: search the city tour in the long list, start the download for package one and wait for everything to start over again, all on a bad network connection. Why create a feature to install a tour in 12 packages that all can fail during download and installation? Combining analytics from network connection type and speed with successful tour downloads gives insights in problems with the chosen installation mechanism. The biggest problem was that the working of the tour guide is based on the current GPS locations. We must be in the city itself to use the tour guide, we can't start it before, to see the content or the way it works. GPS uses a lot of battery, so if we walk around for a couple of hours, my battery will be empty. This makes the tour guide useless because it's hard to recharge a phone in a city from medieval times.

In the beginning I really wanted to go to the city Provins, but after fighting with the app for an hour I was totally done with it, and also with the city of Provins. We went to the beautiful Château Vaux-le-Vicomte and we were all dressed up like princesses, queens and kings. If I wanted to use the app I had to take at least eight hurdles. Hurdles that could have been removed if the user and the market were properly analysed.

WHAT IS ANALYTICS?

"The Only Thing We Have to Fear Is Fear Itself" - Franklin D. Roosevelt[16]

If we think about analytics, some people only see numbers and more numbers. So what if you're not good in statistics or calculations? I for one, I am not good at math but I love puzzles. Analytics is closer to understanding how systems operate than it is to pure mathematics. Also, it is good to fear. Fear alerts us, it keeps us safe from the unknown and protects us from danger. Think about analytics as alerts to learn something new. Is learning a new and unknown skill something to be afraid of? The first step in controlling your fear is understanding it.

Definition

Analytics[17] is defined on Wikipedia as the discovery, interpretation and communication of meaningful patterns in data. Or more simply put it: "talk about the trends in big numbers". Using analytics means that we can transform raw data into meaningful patterns for decision making and performance enhancement. To do this we need to use statistics to collect, organize and present our data. In summary: With statistics we reduce the big numbers into single parameterization, with analytics we interpret those patterns. To give an example: in the last six months, 20 out of every 100 visits were on a tablet (statistics), compared to the last three months where this was 3 out of 100, therefore we should start testing the IONO app on tablets (analytics) as well. If you want to read more about the difference between analytics and statistics read this article from Jeffery Strickland.[18] When we talk about mobile

analytics; it is the discovery, interpretation and communication of meaningful patterns in collected data from the mobile device and mobile app. Within the mobile app, is a library incorporated to record the data and send them to the mobile app collection. Examples of recorded data are installed app version, device name, network type, app crashes, installed and deleted app, country and language.

Analytics is not an application or product but a process where the recorded raw data is interpreted and transformed into analytics. Like Grigoriy Kogan[19] points out: "If we want to get actionable insights from analysis, look at the application as a funnel instead of a hierarchy of pages." Visitors are the input, they perform actions and some percentage becomes (returning) users or customers (the output). With the funnel concept we are looking for data and start thinking about what is making visitors drop-off?

How to use analytics

According to Julian Harty the patterns found can be used in three ways.[20] The first way is to describe the current situation, to see what happened and to understand why it happened, learn from it and try to prevent the problems from happening again. Secondly, patterns found in historical data, can be used to give a prediction for the future. These predictions is not what will happen, as Harty points out, but what cán happen. With this prediction, there's the possibility to prevent the problem from happening again. The third and last way to apply analytics is to recommend an action as prescription. This makes analytics a decision support system. In summary, analytics can help us to: a. understand the description of the current situation, b. if we can see what can happen as a prediction and c. if we have a prescription on how to act in a certain situation. It can feel like too many if's and this is "too good to be true". Analytics is not a miracle pill ensuring there is no more pain, nor is it a false belief that we will never fail again because we made a wrong decision. Analytics is not a prediction on what *will* happen, but on what *can* happen. This is a big difference. Analytics will help us but we need to keep thinking for yourself because many things in the process can go wrong. For every decision, we make assumptions, we have ideas, biases, experiences and we take short cuts. We may misinterpret the data, the numbers can mean something else than expected, the calculations can be wrong, the data can be wrongly recorded or we mixed up cause and effect. So the more we dare to question these assumptions and possible faults, the better we are able to use them to analyse the results.[21]

THE FLOW FROM DATA VIA ANALYTICS TO TESTING

> *"Of course it is happening inside your head, Harry, but why on earth should that mean that it is not real?" - J.K. Rowling, Harry Potter and the Deathly Hallows*

The figure below (Figure 1) shows an analytics model I created to learn how all the elements in the system work together and to gain insight. It is a personal model and not fully validated or complete. It helps to see that there are multiple feedback loops. If you look at the model in figure 1 then the *first loop* is the creation loop within the development cycle. The *second loop* is the recording loop from the push to production, the data records are saved in (cloud) database collections and the loop finishes back into the development cycle with an update of the recording configuration. The *third loop* is the processing loop, which picks up the raw data from the collections and transforms it into a dashboard and communicates the alert message back into the development cycle. The *fourth loop* is the asking questions loop, which starts at the intersection of the system model with the dashboard and continues with assumptions, asking questions and finishes by searching and selecting data from the collections. The asking question is the discovery part of analytics. This loops determines that, we can come to

discover patterns and depending on the type of question we will get a specific type of answer, in a specific type of pattern.
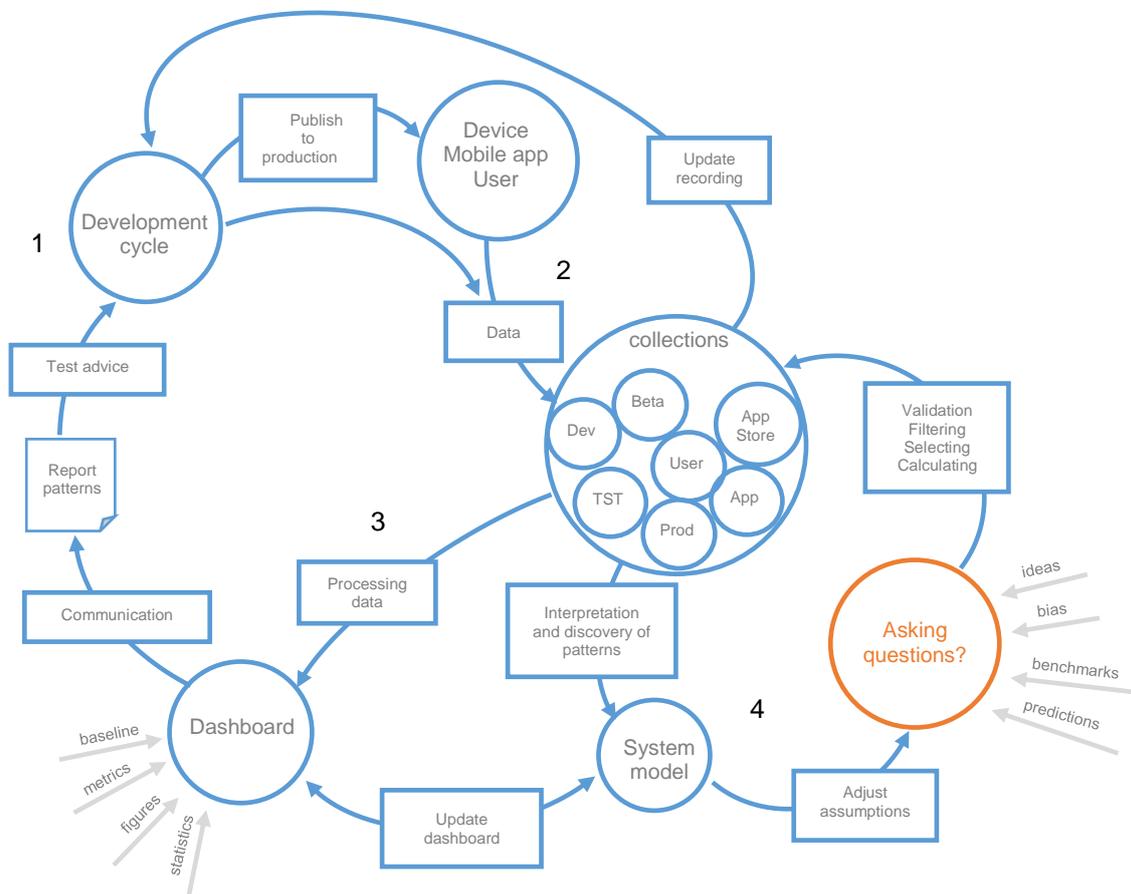
Figure 1 - An Analytics Model

Before we go into the analytics model, in the coming paragraphs, we first need to discuss a frequently used form of analytics "the most popular device of the month". My problem with this popularity list comes down to the question: How does it make my testing better for a tourist of the city Provins, if I know that, the iPhone 7s Plus is the most popular device in the USA?

MY PROBLEM WITH THE MOST POPULAR DEVICES OF THIS MONTH

> *"One size fits all cloths" versus "Customers prefer custom-tailored clothing"* [22]

A first reflex a tester can have is to use the top ten most popular devices from the last month as analytics. Using a top ten of most popular devices from this month, like the Techradar buying advice[23] or the

buyer's guide of GSM Arena[24], is a one size fits all approach. I use these guides to understand which devices *can* be bought, not what my current users have in their hands. This 'most popular device list' is a reduction of all the diversity in the market and between apps and users. It only looks at the device, where are all the other dimensions? My problem is that it is an oversimplified view on the mobile world. If we look at analytics of OpenSignal from 2015 about the fragmentation of the Android platform then we see the diversity of the mobile market.[25] This diversity is a weakness but also a strength. The user has a choice in every direction: manufacturer, brand, price, screen size, screen type, number of cameras, network type, OS, OS version, processor, memory size, disk space, etc. The user decides what to buy, when to buy, when to upgrade and even if to upgrade. This is what Ahonen calls "Communities dominate brands".[26]

If we look at the IONO case: How does it make my testing better for a tourist of the city Provins, if I know that the iPhone 7s Plus is the most popular device in the USA? The assumption is that this is also a popular device among the end users of the tour guide app. But how do I know that for sure? Does it include the growing market of students studying history or a market for elderly that enjoy their retirement? A tester needs to do his own analysis, go through data and have the skills to incorporate analytics into his testing.

Another argument for not using a popularity list is that it's not about the most used device. I was involved as a mobile coach in a two-year-old startup company in the Netherlands, that is now active in more than 25 countries. The CEO provided a pitch at a conference in San Francisco where a Japanese journalist wrote a beautiful article and the next day, Sony, Sharp and Kyocera devices were instantly popular in the download data. In the early adopter countries, the user numbers are still very small and their devices are not yet seen in general popularity lists, but it's very important for a company to support those devices to enable growth of their popularity. The analytics that I require as a tester need to be relevant. It should say something about the users that have the app in their hands. By relevant; I mean representative of the type of market, speed of change or dynamics of the market, actions of competitors, type of users and comparable apps. What is the end user expecting and what is important to him or her?

Now we come to the core of the analytics: asking questions. This step is the discovery part of analytics. This feedback loops determines that we can come to discover patterns.

THE CORE OF ANALYTICS: ASKING QUESTIONS

> *"Asking questions is the difference between people who do things and people who dream about things" - Steve Jobs*[27]

For me, the hard part of analytics is not learning to use a tool, but start using it to search for a specific goal. It is like when I was driving for the first time alone. With my just acquired driving license, I was driving in my parents' car. I was just driving around and really had no idea where to go. I was lost within an hour, 200 km away from home and had to find my way back home on my own. There was no one to tell me where to go, or what to do: turn left or right. No instructor or my parents to give me directions (and there was no such thing as Google maps or Tom Tom). I had to figure out how to get home myself, by asking questions. Having your drivers licence means you are allowed to ride in a car, that you know how to control it and what to do with in on the road. But we only really learn how to drive, by just doing it a lot; by driving a lot of miles, and learning how to find your way around. If we start with analytics, it is just like learning how to drive. We request an account of the implemented tool in the mobile app, for

example Google Analytics, Adobe Omniture, Mixpanel or any other platform[28], and just start somewhere. We need to learn to control this analytics car and try not to get lost in all the numbers. Then we need to start driving and begin asking questions. But what are the right questions to ask? I will discuss several approaches to formulate these questions.

Creative discovery approach
The first approach is what Lutz Finger and Soumitra Dutta in the book Ask-Measure-Learn[29] call the "creative discovery" approach. If we follow this approach then we are trying to find something that we don't know now (yet). Creative discovery is asking yourself the question: Why is the data like this?" The data we're looking at is the resulting answer. In this approach, we need to work backwards. We can start with statistics and calculate the average number of visitors, number of sessions, unique visitors, page views, top used devices, top used OS version, number of touches per page, total time spent in the app, errors per session and number of sessions that end in a crash. After that we will try to combine these calculated statistics and try to relate them. For example - how much time is spent in the app for each user scenario per platform, or the number of errors per session per used devices. The next step can be to add time to these metrics, and plot the "spent time in the app per platform" on the y-axis and on the x-axis the "hours per day" or "days of the week". Visualisation of data in charts sometimes reveals relationships that a single metric doesn't show. There is a catch though. It is hard to find something that we are unaware even exists. Data doesn't tell us if there is a problem nor shows us what is "normal" or "not normal" behaviour. When we start using Google analytics (or any other tool), you'll find soon enough that these tools in itself are not saying anything like: "and there is the problem!" We have to figure out what we want to know, so we can ask the right questions. And interpret the data.

Business sense approach
The second approach to finding questions is, as Finger and Dutta point out is, we formulate the question using our business sense, also often called domain expertise.[29] With this approach we need experience, examples or oracles to formulate questions for our analytics. For example, a domain expert for the IONO app, knows that a tour leader downloads all the guides and spends hours in the app to prepare for a visit. This in contrast with a tourist who spends minutes per step in one tour. So time spent per sessions can't be averaged because this is over all different user groups.[30]

Combine approaches
In general as Finger and Dutta describe, finding questions will be a combination of both approaches. As we try to find meaningful patterns we will combine both approaches. In this way the data can be validated and placed in context. So it starts with the ability to filter, select or calculate averages and other statistics as metrics (the creative discovery). But these results only give us meaning if we have the context: What are we looking for? The resulting questions determine the goal of the analysis.

Example questions
There is a big chance that you already have some questions. Let's take a step back and look at some example questions and see if they are usable for finding patterns. Here below, and in the reference[31], are some examples of incomplete questions that you may recognize from your own organization:
- When are they using our mobile app?
- What is the average time a feature is used in the app?
- Where are they using it from (country, languages, network type and conditions)?

Let's look at the example: What is the average time a feature is used in the app? This question is incomplete because it fails the "So-what" test.[32] "What does this mean for me or why is it important for me?" If there is only a single number like: The average time a feature is used in the app is "*42 sec.*" This 42 sec. does not make much sense. This is a statistic not a pattern with meaning.

Transform into useful questions for analytics
As Finger and Dutta further explain, there are two types of questions: benchmarks (how did we do compared to …) or predictions (what will happen if …). For example, a measurement like number of downloads doesn't tell anything. The IONO tour guide app from the Google Play store[33] has been downloaded between 5,000 and 10,000 times. This is not helpful as an analytic but if it is compared (benchmark) to the previous year when it was between 0 and 1,000 then it shows growth. Is this a useful pattern? No, because it is not yet complete. What is missing is the reason why it is growing. Is it because of new features, bug fixes, marketing or new content? Instead of a benchmark to the own growth it can also be compared to another app like TourPal Travel Guide.[34] This app has been downloaded between 500,000 and 1,000,000 times. Then the question becomes: "How can we grow from 10,000 downloads to 100,000 in the coming year?".

Smart question
Thus the right question asked about data is a question that is SMART: *Specific*, *Measurable*, A*ctionable*, R*elevant*, and *Time bound*. Specific explains which data is used. Measurable means that there is data available to calculate the result. Actionable says that it is clear what to do with a positive or negative result. Relevant requires a direct link to the problem that is analysed and Time bound states for which period the question is answered.

An example of a smart question:
> *"To keep our users on our platform (relevant) we need to analyse the crash rate after each release: What is the "crash rate", number of crashes per 100 started sessions (measureable) per platform (specific) after each release to production, if we compare last 24 hours (time bound). If its higher then the last three months, then everything is* analysed*, till the cause of the crash is found and fixed with a new production release (actionable)."*

The crash rate is the KPI[35] or statistic parameter that needs to be calculated by the team. The crash rate statistic parameter is a chosen bench mark. If the crash rate is now 3%, than with this max bandwidth of 1%, it cannot be raised to 4%. Examples of a crash rate could be: % crashes per platform (number of crashes/number of sessions), API error responses (404/500), one star review rating from the app store or number of service calls with the category error.

DATA COLLECTION AND VALIDATION

> *"72% of the users were accessing an application from the city Groningen (Netherlands)"*

When I was analysing the data for a customer I found out that 72% of the users accessed the application from the city Groningen (Netherlands). This looks like almost everybody was working in one city. It felt strange because the company offices were in The Hague, Amsterdam and Amersfoort. During the analysis, I didn't get it at first, but the moment the scrum master told me that everybody accessed the network through a Citrix[36] connection on a server in Groningen, it clicked. The data clearly needs context

to understand the meaning of the 72%. In this example the number said nothing about the working location of the users but everything about the connection type.

Multiple data collections
When they hear about analytics, most testers immediately think of Google analytics. Google analytics or any other platform is only one of the many data collection tools.[37] With these platforms we can collect the events and user interaction with the application, but not the review remarks from the app store or the number of service calls to the help desk. To discover patterns we need multiple data collections. Every collection has different data, a different size and a different quality. Data can be collected from the development phases (build and test phases), from beta testers (crash reports, usability sessions), from production (operational monitoring), from the app store (review forms), from apps (performance, crashlytics), or from users (social media). Combining the different collections is the area of business intelligence and of data warehouses.[38]

For me analytics is not about tooling, but about understanding what the data is trying to tell us. Most of the times when I'm testing a mobile app, there is a minimal set of data and the source is more an opinion than a fact. Should I be glad that I have some data? Or should I question them? If the data are not based on facts but stem from the opinion of an "oracle", they might be right, based on years of business experience, but they can also be false and merely used because there was a lack of time to collect real data. So, before we start using the data, we always need to validate it first, as all later decisions are based on this set of data.[19,20]

Validating data collections
When validating the collection process it helps to have a reference. The data as a set of isolated figures can mean everything, and therefore not tell me anything. We can only interpret it if we know the context. And asking questions, to others but surely also to yourself, is the only way we can figure out the context.

In the next paragraph we are going to talk about a dashboard. The dashboard is the "output" from the processed collected records. We have to look forward to the data output in the dashboard, and then validate the data backwards to the recording process. Examples of validation questions are:
- What is automatically collected by the tool, and what should be done manually?
- Where is the data stored and how is the data stored?
- Which data is actually recorded?
- Is there sampling of data and can it be fine-tuned?
- Is the data rounded, topped, calculated, approximated?
- What happens if there is an error?
- How much data is saved, for how long and how is it cleaned up?
- Who has access to the data and does it need to be encrypted?
- What happens if there is no internet connection?
- How is the data synchronized?
- Is there any latency and how does it reflect the recording?
- Which clock is the reference for all the records?
- What device specifics can be measured (gestures, memory, battery, CPU, network, location, language, crashes, install/re-install/un-install, app killed, country)?
- Can we make a difference between development and production environment and compare them?
- Can we follow a user over multiple channels?

- Is there a definition, description or standard of the data recorded?
- Are there automatically metrics calculated?
- When and how can we change the recording configuration?

For every tool that we might want to use, take a trial period that is long enough to cover a whole release cycle. And use every function to experience the day-to-day usage. When using a tool in the trial period there are several tests that can be performed to validate the collected data. Examples of these tests are:

- Perform tests to compare results (if we do a series of tests using the same tool, we expect the same results) between record sets after a user scenario or after a min/hour/day/week usage
- Compare two different tools to validate if the recorded data is equal
- Implement a library but don't use the app, to validate that there are no records collected (is zero really zero?)
- Create a validation model to re-calculate the collected data
- Dry-run the process from collecting the data into a dashboard and into a report, to see if we have the right data for the report, or more basically: is the data we have of sufficient *quality* enough to make a report? Or perhaps is cleaning or updating of the configuration needed.

When we start to apply analytics we gain insight into the effectiveness and the value of the analytics. As Harty explains, this meta-analytics[39] can help inform us, about the value of feedback paths and what we do with the data received. One way to measure the value is the concept of signal-to-noise ratio. This ratio expresses how good the information is coming through. Another relevant measure is the latency – how long it takes for information to flow through the particular path, and then how long it takes us to act on the information when we have received it.[20]

DASHBOARDS AND BASELINES

> "*Learn from yesterday, live for today, hope for tomorrow. The important thing is not to stop questioning*" - Albert Einstein[40]

For me, the dashboard is the tipping point in the analytics process. Because with the dashboard comes the realisation that I have to do something with the results; then it becomes real. First we look forward "what do we want to do with the results?" and then we look backwards to see if the (original) questions are answered. More generally, the dashboard is a signalling and alerting system. For example, if we our speed is now 130 km/h and we look at the road sign (the baseline) that says 120 km/h, then in a blink of an eye we see that we are speeding and we need to slow down to prevent getting a fine. The dashboard can contain multiple metrics, figures, colours and if needed sounds to get our attention. So, the dashboard is like an alarm clock. We can ignore it if everything is operating within pre-set limits, but when all the signs go flash, beep, or blink, we have to start investigating what is wrong and needs to be done.

A simple but effective example with a baseline is the dashboard I see when I enter my local supermarket, see figure 2 below. This dashboard is put just before I pick up a shopping cart and enter the shop. The dashboard has only one message "Every day we have the lowest price, just see". The dashboard is SMART because it is specific (message about the price), measurable (price calculation), actionable (pick up a leaflet and tell me if the price is lower somewhere else), relevant (make this supermarket my price reference) and time bound (this week). This supermarket example also has a baseline. In the

picture 2, the last column on the right in red says "price difference". Here, the baseline is created by taking a sample of products and comparing the prices. If the price is higher, an alarm is triggered and the price is directly adjusted.

Figure 2 - Example supermarket dashboard

Baseline

The "price difference" baseline is a benchmark to show what the average product price is compared to the directly competing supermarket. The average product price is based on a sample group of "normal" products. At first, this seems like a good baseline. It suggests that this supermarket really is the cheapest. But if we take another look and think about these selected products then we realize: Do I normally buy these products? And is this the actual price at this moment or is there any latency? When using sampling we should be aware of the impact of different sampling techniques. This is what Darrel Huff calls "sampling with bias".[41] The problem with this supermarket sample is that the list really doesn't say anything, because no one is going to check if it's true, and actually buys this exact list of groceries in both supermarkets. We also don't know the date on which these prices are checked in the other supermarket. And even if this is a correct list, it could be possible to make a similar list like this, but with products that are cheaper in the competing supermarket. To have a more useful baseline the list of product should be fixed, consisting of "ordinary", daily used products (like bread, milk, eggs, some vegetables, cheese and meat). The prices should be compared every week on a fixed date, only then we can see if this sample is representative.

Creating a good baseline takes time and experience in selecting data, but also in understanding the impact of change over time (like seasons), bias, noise, cleaning data and filtering. In the supermarket example, we are looking for a list of products. Everybody has a list of about 200 products that rotates every 4 weeks and every week we buy 40-50 products from this list.[42] So we should collect the list of 200 products and collect the prices of the two supermarkets for 4 weeks. This is what Harty calls "the normal numbers for each user group".[43]

Looking forward

When we start creating our own dashboard with a baseline, it helps if we start looking forward. If it is clear what the goal is, we can work our way backwards. As a tester, the goal is to be able to test a mobile app better (more efficiently and effectively with analytics). What kind of dashboard would signal or alert me as a tester? A dashboard should give me feedback on how features interact with other apps, with the OS version, with the user, etc. To make it actionable the dashboard should be linked to my risk analysis (as far as possible). I use the I SLICED UP FUN mnemonic perspectives from Kohl for my product risk analysis.[44] The perspectives are a point of view on the application, a focus for our testing. We can also think about perspectives as a quality attribute of the application.

Examples to test with these perspectives are

- **I**nput: Which gestures, sensors and events are used to give input;
- **S**tore submission: App store reviews per version and persona;
- **L**ocation services: Usages of location (GPS, Wi-Fi, IP address, time zone) within features;
- **I**nterruptions/Interactions: With which other apps are we interacting? How is the app landscape on a device look like?
- **C**ommunication: How are users talking to us (calling, SMS, chat, email, forum)?
- **E**rgonomic: How many actions (tap, swipe, clicks) are needed per scenario, how much time does a scenario take to execute, physical complaints from using the app;
- **D**ata: How much data is used, which data is used, which action/screen uses how many data, which API uses which data on which screen;
- **U**sability: Type of defects per persona, used gestures, problems per concept
- **P**latform: App version, manufacturers, spread of devices, country
- **F**unction: Which feature is most/never/first/combined used, how long is every feature used, error/crashes per feature
- **U**ser scenarios: What is per persona the data on numbers of errors, pages visits, gestures, API calls, time, crashes, devices, app kills, app version, update history
- **N**etwork conditions: Wi-Fi, 3/4G, network speed, including performance metrics like CPU, battery, memory, latency, error messages, response time and API call per network type

If we have all these areas of an app covered on our dashboard, and updated regularly, the greatest risks will become clear in the analytics. Based on these risks I know what tests I should perform. In this way the analytics are a live feed into the risk areas of mobile app testing.

REPORTING VIA STORYTELLING

> "*Data are just summaries of thousands of stories – tell a few of those stories to help make the data meaningful*" - Chip & Dan Heath[45]

Collected data is not perfect, records can be missing, maybe not everything got recorded, there can be too much or too little or it's not possible to create a report. Analytics is not about collecting more data on a more frequent basis, instead it is about filling the gaps and connecting events to a complete story about what is happing with the application when it is used by the users. Reporting is the communication

part of analytics. Communication, in order to make a decision. The dashboard from the previous paragraph is the alarm clock to get the attention of the tester. The next step is delivering the message to our product owner, scrum master, other testers or to the whole development team.

The message in the report is much more effective if we use a story.[46] Storytelling is a technique to help remember the data that we used. The story of my problems with the IONO tour guide app is much easier to remember than a table with a list of defects and the relative ranking. I have to confess that my reports and presentations were and still are filled with tables, pie-charts and bar-graphs. It is hard to change a bad habit. I think my bad habit came from my bias to analyse data with Excel. When I wanted to communicate to a development team, I started with importing the data into Excel and created tables, graphs and timelines. These "visualisations" then went into a report in PowerPoint.

To start a new habit I followed a training from Cole Nussbaumer Knaflic and this opened my eyes.[47] The training made me realize how much brain-work I was giving the reader of my reports. Communication is not a success, when "I have done my talk", when I have displayed my message. It is a success when we have understood what it is I wanted to say. My audience had to process all the data in the tables and come to a conclusion. I had no idea my message was not coming across. The data I used were not supporting the story but were filled with clutter. According to Nussbaumer Knaflic storytelling exists of 5 steps:

1. Understand the context
2. Choose an appropriate visual
3. Identify and eliminate clutter
4. Focus audience's attention
5. Tell a story

These steps help us to use the data (dashboards, baselines, statistics) as supporting material in the story we want to tell. Below I'll give a short summary of these five steps.

Step 1 - Understand the context
Like how analytics doesn't start with processing the data, storytelling doesn't start with creating graphs and tables from the data. First we need to have a solid understanding of what we want to do. In this step the story you want to explain needs to become clear: To whom are we communicating? Who is your audience? Are we talking or is it only in a written form? Is there already a relationship with your audience or do we have to build up credibility? With a clear image to whom we are talking, we need to understand why they should care about what we say. Make your message relevant for your audience. What is the "something" we want our audience to know or do? Maybe you want them to 'accept your reasoning', or let them 'see' something.[48] In this context the collected data becomes supporting evidence of the story we build and tell. A good technique is to combine all the elements from the context into a visual outline like a storyboard. This storyboard then becomes the structure for the whole story. Understanding the context is a crucial step, if we would skip this step then the following steps becomes an misguided missile.

Step 2 - Choose an appropriate visual
Each type of visual, like a simple text, a table or a graph, can be effective, but in it's own area. A simple text is ideal to highlight a single number or word. A table is for example good to read in a report, but not in a live presentation because our audience start reading instead of listening to us. A graph with points (to show relationship), lines (continuous data), bars (compare groups or periods) or areas (multiple dimensions) can be processed faster. What's the right graph for my situation? What is the easiest for our audience to read. To validate this we can perform a test with a friend or colleague:
- Where do they focus?

- What do they see?
- What observations they make?
- What questions they have?

Step 3 - Identify and eliminate clutter

Eliminate elements of communication that's not adding informative value, or not adding enough informative value. Every element that is in the visual requires "processing power" or has cognitive load form the audience. Think about minimizing perceived cognitive load. A reasonable minimal number of elements and still allows us to get information across. Remove, for example, colours, lines, difference in size and numbers that are not needed for the story. If all elements send out a signal "I'm important" than nothing is important and everything is noise. Attention needs to be guided. And by removing cluttering we can focus the audience's attention in a way that supports your argument.
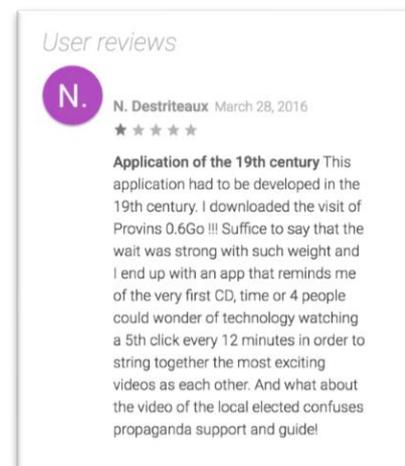
Step 4 - Focus audience's attention

Attention is learned. This is what is also called "we need to learn to read a graph". We need to learn to understand the visual cues. When we look at a visual we don't only see with our eyes but also with our brain, everything you have learned and the memory of all previous experiences. A simple test is to create multiple versions of a visual and show them to a friend or colleague: to which point is the attention drawn? Where are the eyes drawn to? Is the attention going to the point where the conclusion is in the visual? All the cues create a visual hierarchy of the elements that lead the audience through the information in the way we want them to process it.
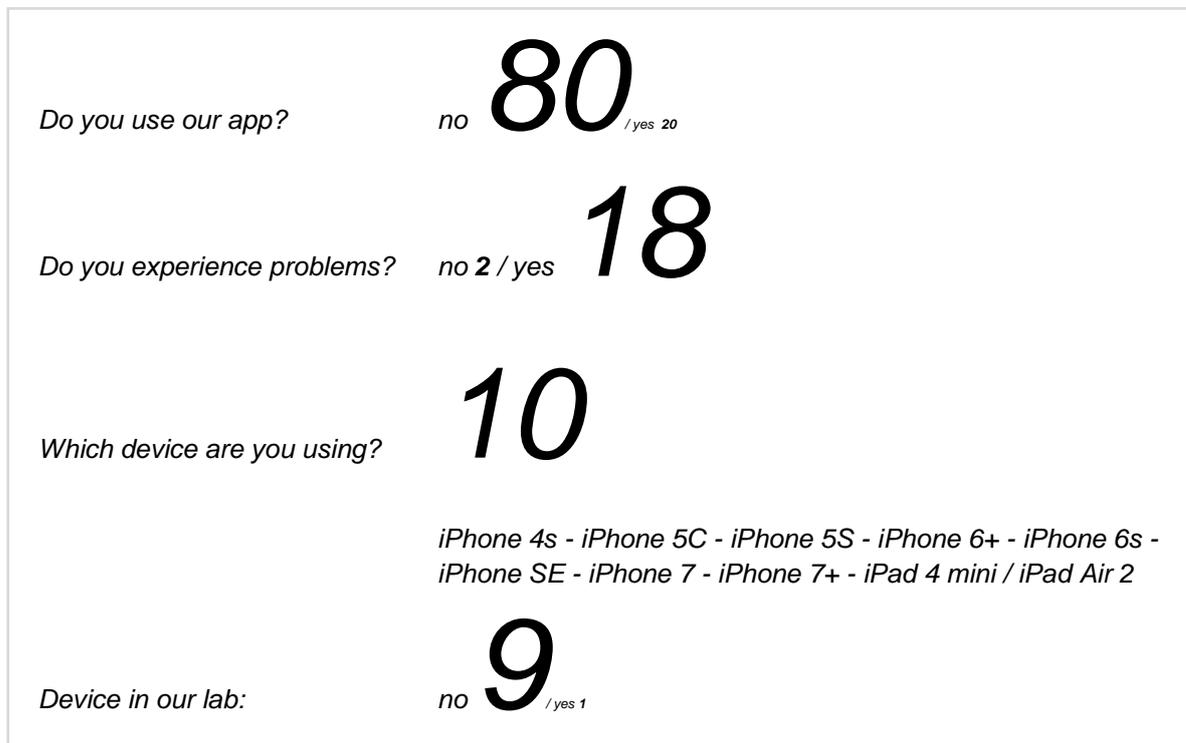
Step 5 - Tell a story

Let's go back to the IONO tour guide case from the beginning of my paper and assume that I have contacted the development team. How could we use storytelling to get the message across about the problems with the IONO app? Not just that the app is really performing badly, but also why we think it is, supported with data. Let's assume that we went through the whole data processing, we have created a dashboard with problems per used devices, the % crashes per platform, API error responses, one star review rating from the app store, number of service calls with the category and the install/de-install rate per week. Now we need to present the next step to our product owner and the development team. I've created two stories we could use to pitch; from the end user's perspective and from a tester's perspective.

The story for the end user point of view

"After reading review after review from the app store I wanted to validate the experience from real end users that visit a city and uses our app. Let's imagine: you're in front of an ancient church in the beautiful city Provins in France. We perform a simple survey and tally every response. I want to know: are you using the IONO app, are you experiencing problems and which device are you using? The results of the survey are:

User reviews

N. Destriteaux March 28, 2016
★ ★ ★ ★ ★

Application of the 19th century This application had to be developed in the 19th century. I downloaded the visit of Provins 0.6Go !!! Suffice to say that the wait was strong with such weight and I end up with an app that reminds me of the very first CD, time or 4 people could wonder of technology watching a 5th click every 12 minutes in order to string together the most exciting videos as each other. And what about the video of the local elected confuses propaganda support and guide!

Do you use our app?	no **80** / yes **20**
Do you experience problems?	no **2** / yes **18**
Which device are you using?	**10**
	iPhone 4s - iPhone 5C - iPhone 5S - iPhone 6+ - iPhone 6s - iPhone SE - iPhone 7 - iPhone 7+ - iPad 4 mini / iPad Air 2
Device in our lab:	no **9** / yes **1**

If we want to compete with our direct competitor, the TourPal Travel Guide app, we need to see what end users experience, we need to change. This means we need to invest in the current test lab of devices."

The story from the testers point of view

"If we want to compete with our direct competitor, the TourPal Travel Guide app, then we need to bring many new features to the market. But with every new release, we also introduce defects. If we want to have a chance to compete, then we need to limit the defects we introduce. We looked at the collected data to find out: why are we missing those defects?

To show the current problem, we compared for each user group (tourist, tour guide and student) the list of devices that visited the Provins website and clicked on the link to download the IONO app, with the list of devices that we currently have available in our test lab. This resulted in:

10%

Coverage of iOS devices used versus available in the test lab, for Android it is 5% coverage.

If we want to compete we need to change this coverage. We need a test lab with a mix of popular, new, low-end and old devices, newly released ones, *high/low screen resolution, fast/slow networking*, et cetera. *This means we need to invest in the current test lab of devices."*

These two stories tell the reasons why the organization needs to update the current test lab. By combining real data with a visual presentation it is easier to understand the message and what the call to action is. In this case the call to action is to invest in a test lab. See also other material about how to tell stories with data, for example check out the narrative visualization from Edward Segel and Jeffrey Heer[49] or the podcast about storytelling from Tom Cagley[50].

EPILOGUE - DAILY TESTING WITH ANALYTICS

"Just Do IT" - Nike[51]

The path to implement analytics into my system was not an easy task. This is something that grew over time. With every new project for every new customer I discovered new ways to execute and use it. Every time I learned how to answer the same question in a better way. Repetition is needed to formulate better questions and to find patterns faster. The analytics process is by far not a tester's only task or responsibility. To create a good running analytics process the whole organization needs to be involved. Within the organization there needs to be an awareness that testing can be improved by using analytics. An organization can become more mature and grow with the following steps:
- Analytics are set and used to collect data
- Collected analytics are refined to relevant information for testing
- Analytics need to be continuously managed and updated

Lessons learned
The path to write this paper started with "business sense". I experienced from the many different projects that organizations and testers are struggling, even fighting, to have some data from the end user environment. To have a running analytics process is a luxury that is not yet coming naturally to the testers. Writing this paper became a "creative experience". The world of analytics was way bigger then I imagined. Your journey into the world of analytics will probably also be a bumpy road. Keep your focus on the questions you want to answer and let me give you some of the lessons I learned during my journey:

- "Just do it", and start formulating your own questions. Look forward and then work backwards to record data into multiple collections;
- Start with what you already know. For example Excel: make a progress reports, imported and combine data, create and apply business rules to select the data, apply conditional formatting to signalling and alarm. From Excel you can go to Google Analytics and combining that into your dashboard;
- Get yourself familiar by using and experimenting with the process, it can take some iterations
- I don't see analytics as a preparation of the test execution phase but as a test activity. Asking questions and finding answers is like reviewing an user story or like pair testing together with a developer;
- If you see a report with too many graphs and tables, then think about what's the story they are trying to tell and what you would change about this data presentation? Could you make it more clear by removing clutter;
- If you see a dashboard, then think about the data that is used to create the dashboard and what you would change about this data presentation?
- Create your own dashboard and work with the collected data. If you work with it, you start to see what the quality is of the recorded data and what you can do with it.

To make this analytics skills your own skill there needs to be room and time to play around and learn from your mistakes. Look at the books and online references I gave in this paper. For online training-possibilities, see the International Institute for Analytics.[52]

So my closing advice to you is: "Just do It" and just start formulating your questions, search for answers, improve your testing and start experiencing the magic of the mobile numbers.

REFERENCES

1 *Mobile as 7th of the Mass Media Cell phone, Camera phone, iPhone, Smartphone*, Tomi T Ahonen, August 2008

2 https://xa.yimg.com/kq/groups/22508646/798333193/name/TomiAhonen-Mobile7thMassMedia-Excerpt.pdf,

3 https://techcrunch.com/2014/08/21/majority-of-digital-media-consumption-now-takes-place-in-mobile-apps/ and http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/

4 *Mobile First*, Luke Wroblewski, A Book Apart, 2011

5 eBuddy, Layar, Whatsapp, Shazam, Flipboard, and PicNic (Click to view)

6 https://en.wikipedia.org/wiki/Time_to_market

7 I also use the term "risk of degradation". Degradation of the beleve that mobile app quality will hold the hurricane of online critics and how this will reflected on the developing company

8 https://en.wikipedia.org/wiki/Time_to_market

9 I also use the term "risk of degradation". Degradation of the beleve that mobile app quality will hold the hurricane of online critics and how this will reflected on the developing company

10 https://www.goodreads.com/quotes/133768-one-s-destination-is-never-a-place-but-a-new-way

11 http://www.provins.net/en/application-mobile-de-provins-uk.html and https://itunes.apple.com/fr/app/iono/id440568185?mt=8, https://play.google.com/store/apps/details?id=com.expressionomade.android.iono&hl=fr

12 http://www.provins.net/en/

13 http://www.vaux-le-vicomte.com/en/

14 http://www.expressionomade.fr/

15 https://en.wikipedia.org/wiki/Persona#In_user_experience_design and *Inmates Are Running the Asylum*, Alan Cooper, February 24, 2004

16 http://historymatters.gmu.edu/d/5057

17 https://en.wikipedia.org/wiki/Analytics

18 https://www.linkedin.com/pulse/difference-jeffrey-strickland-ph-d-cmsp-asep

19 https://www.shopify.com/blog/17921828-dont-fear-analytics-ill-make-it-simple-for-you

20 *The Mobile Analytics Playbook, A practical guide to better testing*, Julian Harty and Antoine Aymer, January 2016

21 http://www.developsense.com/presentations/2012-11-EuroSTAR-CriticalThinkingForTesters.pdf

22 https://en.wikipedia.org/wiki/One_size_fits_all

23 http://www.techradar.com/news/phone-and-communications/mobile-phones/20-best-mobile-

phones-in-the-world-today-645440/3

24 http://www.gsmarena.com/smartphone_buyers_guide_2017_h1-review-1635p7.php

25 https://opensignal.com/reports/2015/08/android-fragmentation/

26 http://communities-dominate.blogs.com/

27 https://www.youtube.com/watch?v=7RYW46_BuUw

28 GOOGLE ANALYTICS, MIXPANEL, FLURRY, LOCALYTICS, WEBTRENDS, UPSIGHT, APSALAR, APPSEE, COUNTLY, ADOBE ANALYTICS, AMAZON MOBILE ANALYTICS, AMPLITUDE, CRAZY EGG, TUNE, ADMOB, NETBISCUITS, ATINTERNET, ASKING POINT, ADJUST, APPANALYTICS , SEGMENT.IO

29 *Ask, Measure, Learn: Using Social Media Analytics to Understand and Influence Customer Behavior*, Lutz Finger and Soumitra Dutta , Feb 9, 2014

30 A fix would be to add a account feature, add tagging to collect the login events, then combine the time spend per platform with login events. Even then there is no difference between a student and a tourist because both don't use an account.

31 Extra examples of incomplet questions: a. Which device is actually in the hands of the users? b. What are they viewing (website, mobile app) and how long does it take to pick up a feature per user group? c. Who are the real users (validate personas in experience, education, feature need)? d. What is the emotion and feedback from the app stores? e. Where is the money coming from (input, actions, output)? f. What is the memory, battery, storage footprint, network speed? g. What are the crashlytics telling us about stability of the app?

32 https://blog.kissmetrics.com/so-what-copy-test/

33 https://play.google.com/store/apps/details?id=com.expressionomade.android.iono

34 https://play.google.com/store/apps/details?id=com.opentech.tourpal&hl=en

35 https://en.wikipedia.org/wiki/Performance_indicator

36 https://www.citrix.com/

37 GOOGLE ANALYTICS, MIXPANEL, FLURRY, LOCALYTICS, WEBTRENDS, UPSIGHT, APSALAR, APPSEE, COUNTLY, ADOBE ANALYTICS, AMAZON MOBILE ANALYTICS, AMPLITUDE, CRAZY EGG, TUNE, ADMOB, NETBISCUITS, ATINTERNET, ASKING POINT, ADJUST, APPANALYTICS , SEGMENT.IO

38 https://en.wikipedia.org/wiki/Business_intelligence

39 https://en.wikipedia.org/wiki/Metadata

40 http://quotes.lifehack.org/quote/albert-einstein/Einstein-Learn/

41 *How to Lie with Statistics*, Darrel Huff, 1954, https://archive.org/details/HowToLieWithStatistics

42 http://www.grocerylists.org/ultimatest/ and (Dutch) https://www.dehuishoudcoach.nl/standaard-boodschappenlijst/

43 https://techbeacon.com/ramp-your-app-testing-mobile-analytics

44 I SLICED UP FUN mnemonic from Jonathan Kohl is: Input, Store submission, Location services, Interactions/interruptions, Communication, Ergonomic, Data, Usability, Platform, Function, User

scenarios, Network conditions, http://www.kohl.ca/articles/ISLICEDUPFUN.pdf

[45]   *Made to Stick: Why Some Ideas Survive and Others Die,* Chip & Dan Heath, 2007

[46]   https://en.wikipedia.org/wiki/Storytelling

[47]   *Storytelling with Data*, November 2, 2015, Cole Nussbaumer Knaflic,
       http://www.storytellingwithdata.com/book/

[48]   Or maybe you want them to agree, begin, believe, change, collaborate, commence, create, defend,
       desire, differentiate, do, empathize, empower, encourage, engage, establish, examine, facilitate,
       familiarize, form, implement, include, influence, invest, know, learn, like, persuade, plan, promote,
       pursue, recommend, receive, remember, report, respond, secure, support, simplify, start, try,
       understand, validate.

[49]   http://vis.stanford.edu/files/2010-Narrative-InfoVis.pdf

[50]   https://tcagley-wordpress-
       com.cdn.ampproject.org/c/s/tcagley.wordpress.com/2017/04/30/spamcast-440-two-storytelling-
       techniques-testing-conferences-mental-models/amp/

[51]   https://en.wikipedia.org/wiki/Just_Do_It

[52]   http://www.iianalytics.com/