Start-Up Technology – A Field Study



Trademob mit Christoph Straser, ehemaliger CTO (siehe www.trademob.com)

Geschäftsmodell: Es gibt inzwischen fast eine Million Apps alleine im iOS Appstore. Das bedeutet für die Produzenten der Apps, dass es zunehmend schwierig wird, Aufmerksamkeit des Kunden auf sich zu ziehen und Downloads zu generieren. Deswegen greifen die Herausgeber immer mehr zu verschiedenen Marketing-Strategien, wie Werbung auf Banners und in besonderen Promotion-Apps, wie z.B. App des Tages. Im Gegensatz zum Online Marketing über den PC, ist es insbesondere auf iOS – Geräten schwer, das Nutzerverhalten zu tracken auf Grund der Restriktionen von Apple. Das heißt, dass die Produzenten den Erfolg ihrer Werbekampanien nicht quantifizieren können. Trademob war der erste Anbieter einer innovativen technologischen Lösung. Sie schaffen es durch die sogenannte Fingerprint-Technologie den Klick des Nutzers auf einen bestimmten Banner im Browser oder einer anderen App mit dem nachfolgenden Download und weiteren Aktionen innerhalb der App zu verbinden. Hierzu nutzen sie einen SDK, welcher ein universeller Code ist, welchen der Produzent in seiner eigenen App implementiert und Trademob es ermöglicht, Ihre mobilen Werbekampagnen auf iOS und Android zu tracken. Eine wichtige Rolle spielt hierbei die node.js (Server Technologie), um im Frontbereich Daten einzusammeln.  
Das Trademob – Team hat derzeit ca. 60 Mitarbeiter. Unser Interview führten wir mit dem ehemaligen CTO, der die Zusammenarbeit von 4 IT-Teams orchestriert hat. Dabei macht die IT-Mannschaft ca. die Hälfte der Mitarbeiter Trademobs aus, ihre Struktur wird im Folgenden genauer beschrieben. Der andere Teil setzt sich hauptsächlich aus dem Vertrieb, Marketing und Public Relations zusammen. Sie sind alle in der Kundenakquise und Kundenpflege tätig. Die Kunden im engeren Sinne des Wortes sind die Apphersteller und die Verbindungen zu Promotion-Netzwerken, über welche die Promotion der Apps am Ende tatsächlich geschaltet wird den anderen dieses triangulären Ökosystems.

Frontend sind die äußeren Ränder des Systems, wo Daten von außen rein und rausgehen. Normalerweise fallen diese Orte bei den Softwareprodukten (im Browser) zusammen, doch bei Trademob gehen beim Frontend des SDKs die Daten rein und beim Frontend des Analytic Dashboards raus.

Beim Einsammeln der Daten am Frontend Server wurden die Daten von einem Analytics-team gepflegt, vor diesem Server sind jedoch die mobilen Apps, in denen der SDK integriert war. Dieselben Developer haben sowohl SDK geschrieben, als auch am Code von der API gearbeitet, die aus dem Frontend des Systems die Daten eingesammelt hat. Somit gibt es eine sehr gute Zusammenarbeit zwischen dem was man holt und dem wie man es abspeichert. Ein Server trackt (ohne SDK), wann der Nutzer auf den Banner geklickt hat und das SDK liest aus, wann etwas in der App geklickt wurde. Für den Input am Server wird Node.js benutzt, weil es viel mehr Requests pro Sekunde verarbeiten kann als PHP oder Python Server. Alle von Trademob verwendeten Tools bis auf Infobright sind Open Source Tools. Da wenig Bedarf bestand, wurden sie kaum angepasst.

**Team1 (Data Gathering)** Eingabeschicht – SDKs, Bannerclicks und sonstiges Erfassen auf der InputServerseite.

Hard tech skills: iOS, Android, Node.js, erweiterte HTTP

Soft tech skills: Auf Qualität achten, attention to detail, eher weniger der kreative Hacker, Gefühl zu Skalierung(in Koop. Mit Team4) (wie kann ich es parallel auf 20/30 Servern betreiben, sodass keine Konflikte entstehen)

**Team2 (Big Data Team)** arbeitet an allen bereits gesammelten sowie live-einfließenden Daten und verarbeitet diese. Man bringt riesige Datenmengen zusammen und erzeugt daraus Reports und KPIs, was das eigentliche Produkt von Trademob ausmacht, was sie dem B2B Kunden bereitstellen, damit sie die Erfolge ihrer Werbekampagnen messen können. Sie nutzen Technologien wie Cassandra, InfoBright, die miteinander kombiniert wurden.

Hardtech skills: up-to-date für die neuen Big Data Technologien(unter anderem Cassandra, InfoBright), performance-orientierte Programmiersprachen (C++, Java, Python, Go)

Softtech skills: gutes Verständnis für Daten und Relationen, in der Lage sein mehrdimensional zu denken

Infobright – Data Warehouse Technologie, Plugin für Mysql, kommerzielle Software. Damit haben sie ein Datawarehouse(Für OLAP - Online Analytical Processing – für Business Leute um auf die Daten bequem und dynamisch zuzugreifen – weiter weg von der Technologie und näher am Business) gebaut und konnten gleichzeitig auf Cassandra als Lock Storage zugreifen.

Cassandra als Lockstorage – ein „dummer“ aber sehr skalierbarer Speicher, um sehr viele Terrabytes an Rohdaten zu lagern. Dagegen schwerer dynamisch Diagramme aus den Daten zu erzeugen, dafür gibt es InfoBright.

**Team3 (Dash Board Team – Analytics Front End Team)** – Diese Daten werden vom Dash Board Team übernommen und nochmal präsentierfähiger gemacht, man schafft Diagramme und Tabellen aus diesen Daten. Es geschieht weniger per Hand, sondern per Software (Node.js), hat ein schönes Frontend(Datenausgabe), das mit Javascript, jquery,css gebaut wurde und eine Singlepage Javascript App ist. Frontend benutzt diese Technologien, damit die Oberfläche möglichst schön aussieht bei möglichst vielen Bildschirmauflösungen, skalierbar ist; sich gut warten lässt.

Hard tech skills:

Serverseite: node.js, Ruby on Rails (schlank und easy am Anfang beim Prototypen, aber je weiter man geht, desto schwieriger zu optimieren, sehr darauf ausgelegt schnell zu Ergebnissen zu kommen), PHP mit Model View Controller (MVC) – um APIs zu bauen. Rest/Websockets, Soap – wichtige Standards, die beherrscht werden sollten.

Frontend Seite: JavaScript, HTML, CSS etc. – für gute Visualisierungen

Soft Tech Skills: Wartbarkeit, gut abstrahieren, Usability, schöne Aufbereitung

**Team 4 (IT – Operations)** Als IT Administratoren werden die Server überwacht für reibungslose Prozesse, Backups, Sicherheit, automatisches Skalieren von Servern undzur Vermeidung von Redundanz. Arbeitet an allen Themen etwas mit, vor allem da, wo die Risiken am höchsten waren (Data Gathering, Big Data Verarbeitung)

Hard tech skills: sehr gute Server Kenntnisse, software deployment, Netzwerktechnologien, cloud: Linux, IP-Tables, Bash Programmierung, Ruby scripting (für die Automatisierung der Server-Systeme), Chef (Provisionierung der Server), Monitoring (Nagios)

**Skalierbarkeit**: Man muss die Skalierbarkeit hauptsächlich über eine richtige Software-Architektur lösen. Dabei ist auch die Shared Nothing Architektur wichtig. Manche Software hat Single Points of Failure, die sich schwer horizontal skalieren lassen (schwer einfach Server hinzuzufügen, da der Overhead-Aufwand zu hoch wird, sie würden nur damit beschäftigt sein, einander upzudaten, was gerade passiert). Daher muss man die Datenflüsse so organisieren, dass man weniger Zustände austauschen muss, und wenn, dann auf eine sehr effiziente Art.

Technologien und Plattformen: Mysql, Master-slave, Cassandra – linear skalierbar

Über Amazon DNS, NGINX Server (HTTP Server Proxy) konnte man die Requests sehr einfach auf viele Server verteilen.

Server: Cloudserver von Amazon.

Grund:

* Flexibler, man kann das ganze Amazon-Ökosystem nutzen.
* Man muss sich nicht physische hardware kümmern
* Viele Dienste verfügbar, die Zeit sparen: Amazon DNS, automatische Backups etc.

**Automatisierte Tests**. Dafür gibt es einen spezielle Testperson (Dedicated Quality Assurance), der die neue Software auf ihre Funktionalität testet, besonders für die Data Collection/Big Data.

Methoden: Behat, Unit testing, Acceptance testing

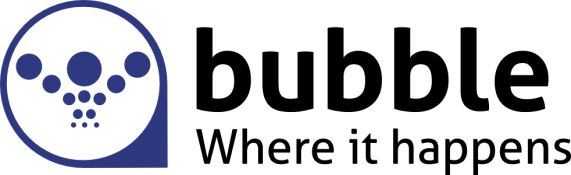
Im Nachhinein haben sie verstanden, dass man mehr Personen in dieser Funktion gebraucht hätte, jedoch besteht ein Kostenfaktor und deminishing returns pro weiteren Tester.

**Sicherheit**: „two factor authentication“ in manchen Systemen (Passwort eingeben und dazu eine SMS auf das Smartphone bekommen)

So wenig wie möglich sensible Daten speichern, dabei vor allem möglichst keine Personal Identifiable Information benutzen (PII)

Sichere Verschlüsselung für die Übertragungswege – SSH; SSL (Kommunikation über HTTPS)

Außerdem hat man sich ein Datenschutzsiegel akkreditieren lassen als Sicherheit für die Kunden.



Das start-up bubble hat eine App produziert, die auf einer Karte in Echtzeit fundiert, um sehen zu können welche Events im Moment statt finden. Beispielsweise möchte man abends ausgehen, hat aber keine Pläne oder Verabredungen getroffen. Ein kurzer Blick auf die App zeigt, ob deine Freunde zusammen in einer Bar sind oder ob es ein Konzert gibt, wo alle sind.

Definitiv ist Geschwindigkeit ein Schlüsselfaktor, denn wer “real-time” verspricht, muss auch Echtzeit liefern. Die App soll flüssig auf mobilen Geräten funktionieren ohne dass das Gerät überlastet wird oder zu viel Speicher benötigt wird. Im Folgenden werden die Server und Client Seite beleuchtet, um zu verstehen wie die Probleme Geschwindigkeit und Speicher gelöst wurden. Das bubble app Team besteht aus zwei IT´lern mit einem Technologiehintergrund im Studium, die auch schon auf Projekten für objective C (benutzt für Apple Produkte) gearbeitet haben und einem WHU´ler, der sich auf den Bereich Vertrieb und Business Plan fokussiert.

Auf der Server Seite werden Rohdaten benötigt. Wie lauten die Straßennamen, wo ist ein Park oder wie lang ist ein Fluss? Das open source Projekt OpenStreetMap wird zu diesem Zweck verwendet. Es ist eine XML Textdatei, die durch eine andere open source software PostgreSQL (ein Datenbankmanagementsystem). Die Vorteile im Vergleich zum „normalen“ SQL liegt in der speziellen Erweiterung für geographische Anfragen. Die software selektiert die Daten und stellt jedes Bild zusammen. Für diese Zusammenstellung wird das rendering tool Mapnik, ein opensource mapping toolkit zum Design der Karte genutzt. Das Team definierte in einem hohen Arbeitsaufwand, dass jede Straße schwarz angezeigt sind, jeder Park grün etc. Für den Prozess der data retrieval, style retrieval und letztlich das Bild auf dem Bildschirm, benutzt das Team CDash.

Auf der Client Seite liegt eine dreiteilige Struktur vor. Wir hörten von unseren Interviewpartnern, dass es auf dieser Seite kein sehr gutes open source Projekt gibt, für die Bildausgabe und benutzen daher ihren eigenen graphic engine. Dieser basiert auf Open Graphics Library for Embedded Systems (openGL ES). Statt für die Konstruktion ein toolkit zu verwenden, in dem man von verschiedenen Buttons oder anderen vorgegebenen Optionen wählen kann, haben sie jeden Pixel selbst gecoded. Dieser Ansatz sei sozusagen der direkteste Weg zur Grafik des mobilen Geräts, weil es direkt auf den Grafikchip zugreift, was die Geschwindigkeit und Gestaltungsmöglichkeit signifikant erhöht im Vergleich zu einem toolkit. Dafür ist allerdings ein größerer Arbeitsaufwand nötig.

Der graphic engine ist in der Programmiersprache C geschrieben, die wie der praphic engine normalerweise aus der gaming Industrie bekannt ist. Einer der IT´ler hat in diesem Bereich während des Studiums schon Erfahrung gesammelt. Weil der Bildschirm 60 Mal pro Sekunde neugeladen wird, ist der logische Aufbau und das coden in C wichtig für die Geschwindigkeit der App nach dem Prinzip “near to the hardware”. Der game engine wird in der App für das “shooten” von bubbles (=likes) genutzt, in ein Event, das man liked. Ein flüssiger Ablauf ist ohne C coding schwer möglich. Neben Geschwindigkeitsvorteilen ist außerdem die cross platform Funktionalität ein wichtiger Vorteil. Es kommt allerdings auch auf den logischen Aufbau der App an. Denn 60 bis 70% der Zeit verwendet der Programmierer für diesen logischen Aufbau.

Die dritte Strukturebene zielt auf das spezifische Endgerät ab. Denn GPS Daten werden auf iOS oder Android unterschiedlich ausgelesen. Auch Variationen in den touch Gesten müssen für den Ablauf der App berücksichtig werden.

Darüber hinaus benutzt das Team virtuelle Server, um die Anfragen zu bearbeiten. Durch die Miete ist eine Skalierbarkeit einfach zu erreichen. Im Moment läuft die App nur für einige Beta user, die in einer whatsapp Gruppe organisiert sind. Wenn ein Fehler auftritt, wird er in die Gruppe gepostet und so schnell wie möglich behoben. Das soll für die Zukunft durch eine Report Funktion automatisiert werden. Die Personalplanung sieht vor jeweils einen Spezialisten für iOS und Android zu beschäftigen, einen Mitarbeiter für die Server Wartung und einen für die Homepage. Es ist geplant hashtags und Videos in die App zu integrieren. Momentan wird das Team durch das Preisgeld eines Business Plan Wettbewerbs finanziert und die Büroräume werden von einem Gründungsstipendium gestellt.



Appleunity ist die ehemals größte Applecommunity Deutschlands (2010). Sie veröffentlicht die neusten Nachrichten rund um die Firma Apple, ihre Produkte und Technologien und wird von 4 Mitarbeitern betrieben. Florian spezialisiert sich auf Java und PHP, während Philipp Biel, Gründer und derzeitiger WHU Student auf das Frontend (HTML, CSS und Wordpress) spezialisiert ist.

* HTML ist die Sprache, um die Inhalte und ihre Bedeutung zu definieren
* CSS ist für das WO und WIE der Inhalte zuständig.
* Die Webseite ist mit Wordpress aufgebaut. Der Vorteil hierbei ist, dass das Backend somit bereits mitgeliefert ist. Es bietet Möglichkeiten, Plugins bequem zu installieren und SEO effektiv zu betreiben.

Als Programmierumfeld benutzt er Adobe Dreamweaver. Bei der Hinterlegung eines virtuellen Servers auf dem PC kann man alle Änderungen live verfolgen. Mit dem Mozilla Plugin Firebug nimmt man dann die Live-Änderungen an der Webseite vor, wo man diese erst nur alleine betrachten kann. Aus dem Dreamweaver kann man diese sofort online schalten.

BWL-IT technische Ziele der Webseite: Retention der Nutzer, um im Google Ranking aufzusteigen und somit weitere Nutzer anzuziehen. Dies erreicht man hauptsächlich durch Benutzerfreundlichkeit und Schnelligkeit der Ladezeit der Seite.

Um eine gute Schnelligkeit zu erreichen, ist ein sauberer HTML und CSS Code notwendig. Mit Services wie W3C und Pagespeed Service Google kann man die Seite optimieren. W3C analysiert die Webseite auf Fehler und PSG gibt Feedback zur Schnelligkeit der Seite auf kontinuierlicher Basis. Sie planen eine teilweise Umstellung von PHP auf Grails, was eine wesentlich einfachere Methode ist, für eine schnelle Seite, es ist stringent und nachvollziehbar für andere Programmierer.

Das Thema Sicherheit ist bei der Firma nicht stark im Fokus, da sie wenige benutzer-sensible Daten haben. Wordpress bietet leider viele Angriffsstellen für Hacker. Bei dem letzten Angriff war eine API innerhalb des Plugins von Wordpress so offen, dass der Hacker sich Administratorrechte erstellen konnte und immer tiefer ins System eigedrungen ist. Als Schutz werden kontinuierlich Backups angelegt. Die Nutzerdaten werden in einer MySQL Datenbank gespeichert.

Appleunity nutzt 2 eigenständige (dedicated) Server. Dies ist ihnen wichtig, da sie dadurch eine eigene IP-Adresse haben. Bei einem Web Space dagegen wird die IP-Adresse oft mit anderen Projekten, die über diesen Server laufen, geteilt. Dies kann negative Folgen für das Google Ranking haben, wenn auf der Adresse beispielweise Pornoseiten angemeldet sind.



Das Startup Bongocoin existiert erst seit ca. 3 Monaten und befindet sich gerade im Zustand eines Minimum Viable Products. Bongocoin ermöglicht das Shoppen in jedem beliebigen Online Shop, wenn Nutzer mit virtueller Währung bezahlen möchte. Der Nutzer kopiert den Link des gewünschten Items und den Preis auf die Bongocoin Seite. Ihm wird daraufhin der Preis in seiner gewünschter Währung angezeigt. Nach einer Überweisung der Coins vom Nutzer, wird der Artikel für ihn gekauft. Das Gründerteam besteht aus zwei Studenten von der EBS und einem Freelancer. Alexej ist zuständig für Wordpress Webseiten. Ein Template wurde bei Themeforest.net gekauft und auf Wordpress installiert. Die Themes sind von Grund aus optimiert für mobile, damit fast keine Anpassungen getroffen mehr getroffen werden müssen, um die Darstellung auf Handys und Tablets zu optimieren. Freelancer (Inder) beschäftigt sich mit Javascript (Calculator), PHP (Code von Wordpress anpassen) und SQL (Datenbank). Er ist zuständig für die API – Programmierung für den Währungsrechner. Er dient zur Berechnung der Bitcoins oder andere virtuelle Währungen, mit der man für ein Item bezahlen kann. Hierfür müssen bis zu drei Börsen angesprochen werden. Das läuft über API Schnittstellen: Yahoo Finance zur Umrechnung Dollar in Euro; Kraken.com zur Berechnung Euro in Bitcoin; Cryptsy.com für Umrechnung der Bitcoins in andere virtuelle Währungen, wie z.B. Litecoins.

Die tatsächliche Umrechnung passiert bisher manuell über eine Börse. Wichtig hierbei ist es, möglichst schnell die Transaktion durchzuführen, um keine Überraschung beim Wechselkurs zu bekommen. Die Überweisung der Cryptocoins landet auf einem Wallet. Dieser hat einen public key (Name des Wallets) und einen private key (Passwort).

Es ist ein virtueller Server im Einsatz, weil dieser bereits für ein anderes Projekt genutzt wird und man nutzt nun einen Teil dessen.

SSL Verschlüsselung wird für die Datenübertragungen genutzt. Sie sind darauf umgestiegen erst, nachdem einige Kunden angefangen haben, sich darüber zu beschweren, dass die Sicherheitsvorkehrungen zu niedrig sind.

Unsere Informationen haben wir nur aus Interviews gewonnen, es wurden keine weiteren Quellen verwendet.

# Unser Interviewleitfaden

Persons

* b) What sort of IT skills do they have among their employees or do they source out? (welche Sprachen spricht der jeweilige ITler)
* How big is the IT department, what has been implemented and what is planned?
* For data communication, how much do they store/exchange via solutions like google docs/dropbox (related to security issues)
* Managing admin/other access roles
* Inwiefern braucht man generalisten vs. spezialisten?
* Welche Profile Profile waren am schwierigsten zu finden?

Software

* a) What sort of software does the startup use?
* c) Do they develop startup-specific software?
* d) Do they use or develop open-source software?
* e) What sort of technologies do they use?
* Project management tools supporting communication across organizational boundaries.

Hardware

* What kind of hardware do they rely on?

Other

* f) Do they face specific IT-related challenges (such as scalability, mobility, security).
  + hacker security
  + Information security management
  + scalability
  + mobility
* Was sind die größten Aufgaben für eure IT-Abteilung und die größten Herausforderungen dabei?
* Priorisierung der auftretenden Probleme (durchlaufend/in bestimmten Intervallen/je nach Dringlichkeit)
* Größere Wechsel von Platformen/Systemen/Technologien

Investment in die IT

# Die Prezi

<http://prezi.com/jtv-gmcry4jh/?utm_campaign=share&utm_medium=copy>

