## CAS Data Visualization

Autorenprojekt Flight Fare Visualization



## **Einleitung**

# CAS Data Visualization

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tristique nibh ac sem consectetur tristique. Nulla mauris odio, dignissim varius neque at, ultrices efficitur lacus. Maecenas cursus dignissim mauris vitae molestie. Ut luctus tortor in ultricies condimentum. Aenean auctor ante libero, vel facilisis felis gravida eget. Donec felis quam, consequat tempor orci vitae, viverra iaculis purus. Ut cursus eros sed erat dictum, et viverra elit dictum. Mauris egestas felis urna, nec ultricies purus tempus ut. Aenean tortor turpis, lobortis sit amet suscipit at, volutpat sit amet enim.

Die vorliegende Dokumentation beschreibt das Autorenprojekt "Flight Fare Visualization", welches im CAS Data Visualization an der Hochschule für Künste HKB erarbeitet wurde.

Das Autorenprojekt wurde begleitend zum CAS Data Visualization durchgeführt.



**Daten** 

## Dater

### **Datenerhebung**

Durchschnittlich gibt es täglich rund 93'000 Flüge, welche von ungefähr 9'000 Flughäfen weltweit starten. Zu jeder Zeit befinden sich rund um den Globus zwischen 8'000 und 13'000 Flugzeuge gleichzeitig in der Luft.

Jeder dieser Flüge besitzt einen Flugpreis, welcher sich über die Zeit verändert. Diese Veränderung der Flugpreise zu visualisieren ist das Ziel von Flight Fare Visualization (ffv).

Im Rahmen der Datenerhebung habe ich mich damit beschäftigt, die Flugpreise abzufragen und für die weitere Verwendung im Projekt zu persistieren.

Ein Flugpreis wird für einen Flug angeboten. Ein Flugpreis wird nach unterschiedlichen Kriterien festgelegt. Jeder Flug besitzt einen Abflugsort (Origin) und einen Zielort (Destination). Der Flug kann entweder einfach (one-way journey) oder retour (return journey) sein. Es gibt Direktflüge (single segment) oder Flüge mit Zwischenstopps (multi-segment). Die Klasse (Economy, Business oder First) ist ein weiteres Kriterium eines Fluges, welches sich auf den Preis auswirkt. Die Anzahl Personen ist ebenfalls ausschlaggebend für den Preis. Zudem ist die Fluggesellschaft (Carrier) ebenfalls eine Variable für den Flugpreis. Am Ende wird ein Flug mit den genannten Variablen von verschiedenen Anbietern (Agent) zu einem bestimmten Preis angeboten.

Um die riesige Menge von Daten überschaubarer zu machen und so die Komplexität der Visualisierung von Beginn weg etwas einzugrenzen, habe ich mich entschieden die Daten nach folgenden Kriterien zu erheben:

#### 1. Origin Zürich

Es werden nur Preise zu Flügen mit Abflug-Flughafen Zürich erhoben.

#### 2. **One-way journey**

Es werden nur Preise zu einfachen Flügen (Hinflug) gemessen.

#### 3. Single person

Der Flugpreis wird immer für eine Person angefragt.

#### 4. Economy class

Es werden nur Preise für die Economy-Klasse erhoben.

#### 5. Single-segment flights

Es werden nur Direktflüge berücksichtigt.

Unter diesen Einschränkungen ergeben sich folgende mögliche Variablen für die Visualisierung:

#### 1. **Destination**

Es werden zwanzig unterschiedliche Zielflughägen abgefragt.

#### 2. Zeitpunkt des Abfluges

Der Zeitpunkt des Abfluges lässt sich nach Bedarf in folgende Variablen weiter unterteilen

- a. Datum
- b. Tageszeit
- c. Wochentag
- d. Kalenderwoche

#### 3. Zeitpunkt der Preisanfrage

Auch der Zeitpunkt der Preisanfrage (request date) lässt sich nach Bedarf in dieselben Variablen wie beim Zeitpunkt des Abflugs unterteilen.

#### 4. Carrier

Alle verfügbaren Fluggesellschaften pro Destination werden erhoben.

#### 5. **Agent**

Alle verfügbaren Anbieter werden erhoben.

#### **Ermittlung der Destinationen**

Es sollen zwanzig Destinationen ermittelt werden - zehn Destinationen innerhalb Europa und weitere zehn Destinationen ausserhalb Europas (Overseas).

Die Destinationen werden nach folgenden Regeln ausgewählt:

- 1. Es wird die Liste die geschäftigsten Flughäfen von Europa bzw. Weltweit als Grundlage genommen
- 2. Pro Land darf nur ein Flughafen vorkommen
- 3. Das Land darf keine direkte Grenze zur Schweiz besitzen
- 4. Es muss mindestens ein Direktflug angeboten werden
- 5. Es werden die Top 5 anhand der Liste von 1 genommen
- 6. Es werden die Flop 5 anhand der Liste von 1 genommen
- 7. Die Regeln werden solange wiederholt, bis je 10 Destinationen ausgewählt sind

#### **Europa**

Rang	Flughafen, Stadt (Land)	ITAC Code	Bemerkung
1	London Heathrow, London (GB)	LHR	X
2	Charles de Gaule, Paris (FR)	CDG	Direkte Grenze
3	Istanbul Atatürk, Istanbul (TR)	IST	X
4	Frankfurt, Frankfurt (DE)		Direkte Grenze
5	Amsterdam Shipol, Amsterdam (NL)	AMS	x
6	Adolfo Suarez Madrid, Madrid (ES)	MAD	X

Rang	Flughafen, Stadt (Land)	ITAC Code	Bemerkung
7	Munich, München (DE)		Direkte Grenze
8	Leonardo da Vinci, Rome (IT)		Direkte Grenze
9	London Gatwick, London (GB)		Land vorhanden
10	Barcelona, Barcelona (ES)		Land vorhanden
11	Sherementyevo, Moskau (RU)	SVO	Х
80	Riga International, Riga (LTU)	RIX	Х
85	Keflavik, Reykjavik (IS)	KEF	Х
87	Belgrade Nikola Tesla, Belgrade (SRB)	BEG	х
88	Malta, Malta (MT)	MLA	Х
89	Rhodes, Griechenland (GR)	RHO	Х
96	Trondheim, Norwegen (N)	TRD	Kein Direktflug

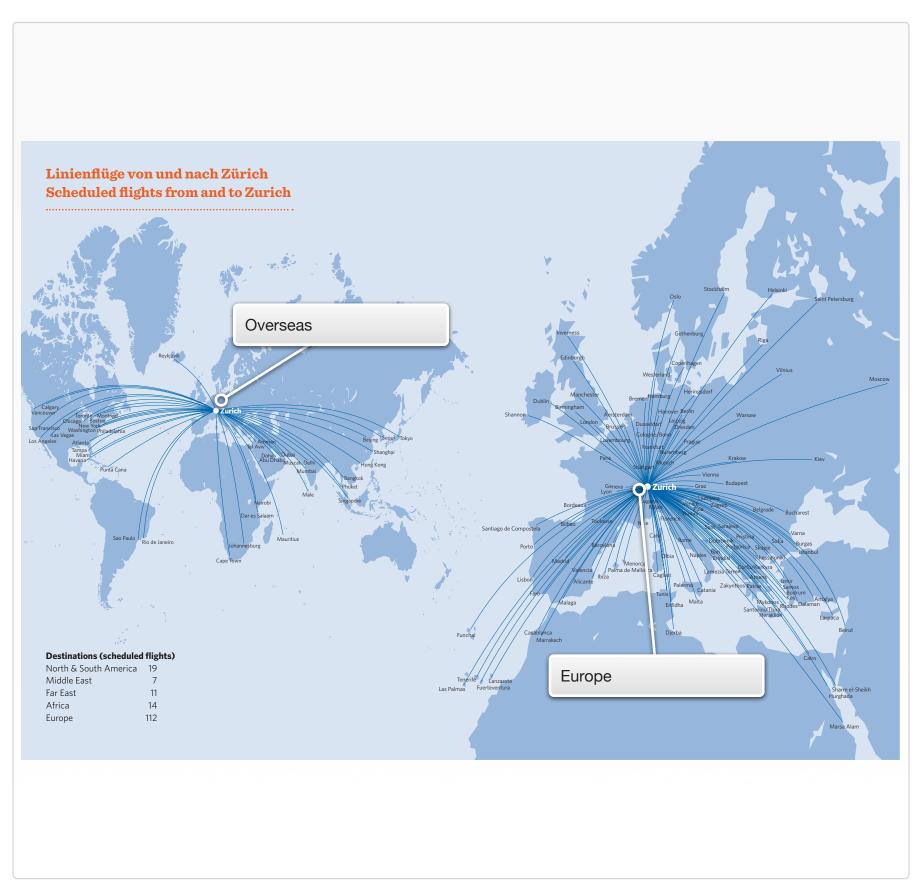
#### Weltweit 🕮

Rang	Flughafen, Stadt (Land)	ITAC Code	Bemerkung
1	Hartsfield-Jackson Atlanta, Atlanta (US) John F. Kennedy, New York (US)	ATL JFK	X
2	Beijing Capital, Peking (CN)	PEK	Х

Rang	Flughafen, Stadt (Land)	ITAC Code	Bemerkung
3	Dubai International, Dubai (AE)	DXB	X
4	Tokyo Haneda, Tokyo (JP)	NRT	X
5	Hong Kong, Hong Kong (CN)		Land vorhanden
16	Singapore Changi, Singapur (SG)	SIN	X
20	Suvarabhumi, Bangkok (TH)	BKK	X
22	Seoul Incheon, Seoul (KR)	ICN	X
33	Toronto Pearson, Toronto (CAN)	YYZ	X
35	Chhatrapati Shivaji, Mumbai (IN)	вом	X
40	Sydney Kingsford-Smith, Sydney (AU)	SYD	Kein Direktflug
41	Sao Paolo-Guarulhos, Sao Paolo (BR)	GRU	X
44	Taiwan Taoyuan, Tapei (TW)	TPE	Kein Direktflug
45	Benito Juarez, Mexico City (MX)	MEX	Kein Direktflug
49	Ninoy Aquino, Manila (PH)	MNL	Kein Direktflug

Bemerkung: Anstelle von ATL habe ich mich für New York (JFK) entschieden, da nach Atlanta nur sehr wenige Direktflüge vorhanden sind.

### Interactive 1.1 Swiss-Linienflüge von und nach Zürich 🐵



#### **Applikation zur Datenerhebung**

Für die Datenerhebung gemäss den vorgestellten Kriterien habe ich ein Java-Programm, ein sogenannter Crawler, entwickelt. Dieser fragt pro Tag viermal via API die Flugpreise für alle zwanzig Destinationen ab. Die Abfrage wird jeweils für den nächsten Tag und bis 90 Tage in die Zukunft durchgeführt.

Beispiel: Abfrage am Montag, 01. Juni 2016

Die Abfrage liefert die Flugpreise für alle Destinationen für den 2. Juni bis zum xx. August 2016.

Ursprünglich wollte ich die Abfrage via Google Flight API durchführen. Google erlaubt jedoch nur 50 kostenfreie Abfragen pro Tag. Bei 20 Destinationen à 90 Tage und vierfacher Abfrage pro Tag sind diese Gratis-Abfragen viel zu wenig. Aus diesem Grund musste ich mich nach einer Alternative umsehen. Die Lösung ist die Verwendung des Travel APIs von Skyscanner Business , welches eine unbegrenzte Anzahl Abfragen zulässt.

http://business.skyscanner.net/portal/en-GB/Documentation/ApiOverview?\_ga= 1.20813560.1378450330.1462389746

Die Antworten von Skyscanner werden transformiert und in einer neo4j Datenbank gespeichert.

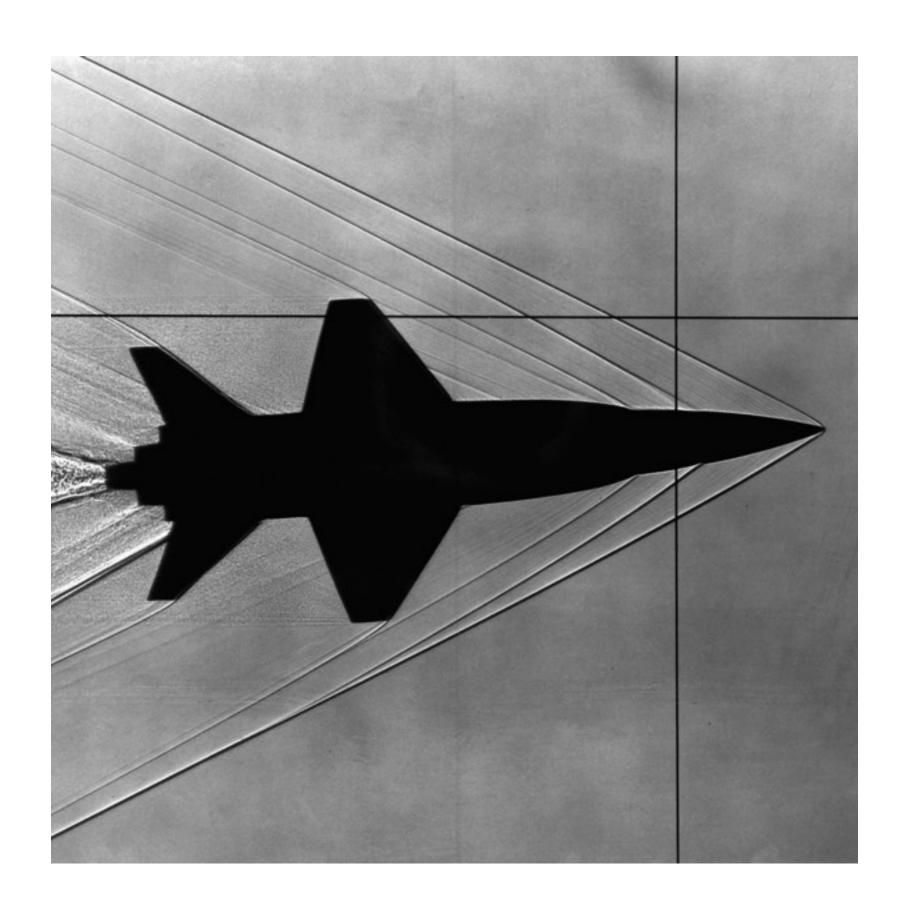
## Daten

## **Datenanalyse**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tristique nibh ac sem consectetur tristique. Nulla mauris odio, dignissim varius neque at, ultrices efficitur lacus. Maecenas cursus dignissim mauris vitae molestie. Ut luctus tortor in ultricies condimentum. Aenean auctor ante libero, vel facilisis felis gravida eget. Donec felis quam, consequat tempor orci vitae, viverra iaculis purus. Ut cursus eros sed erat dictum, et viverra elit dictum. Mauris egestas felis urna, nec ultricies purus tempus ut. Aenean tortor turpis, lobortis sit amet suscipit at, volutpat sit amet enim.

Morbi dictum ligula ante, sit amet ultricies est bibendum non. Nulla massa elit, efficitur in viverra ut, convallis et leo. Nunc malesuada vitae libero ut posuere. Donec quis libero non orci malesuada posuere. In cursus risus in ante elementum fringilla. Maecenas pharetra leo at sollicitudin interdum. Vivamus aliquam, metus non ultrices aliquet, lectus turpis pulvinar diam, vel hendrerit nisi nunc elementum arcu. Suspendisse quis massa hendrerit, elementum dui quis, tristique magna. Sed condimentum in mauris ac tempor. Aenean lacus diam, commodo non fringilla eu, feugiat ac odio. Curabitur eu ultrices orci.

Donec efficitur tellus lectus, varius commodo quam blandit nec. Aenean non euismod justo. Morbi vitae libero quis tellus accumsan convallis nec sit amet arcu. Nulla mollis turpis ornare fringilla lacinia. Integer pharetra odio lacus, ut tincidunt libero pellentesque eleifend. Etiam sit amet felis ac arcu facilisis semper et nec nisi. Aliquam purus magna, porta non justo et, rutrum placerat velit. Mauris a enim imperdiet, ultrices dui ut, mollis sem. Donec eu viverra urna, a vehicula ex. Sed ullamcorper, nibh sed tristique scelerisque, massa sem luctus mauris, vel sollicitudin arcu quam quis lorem. Etiam vulputate, nibh porta varius tempor,



<u>Ziel</u>



<u>Modelle</u>



<u>Methode</u>



<u>Medium</u>