

---

# US FLIGHT

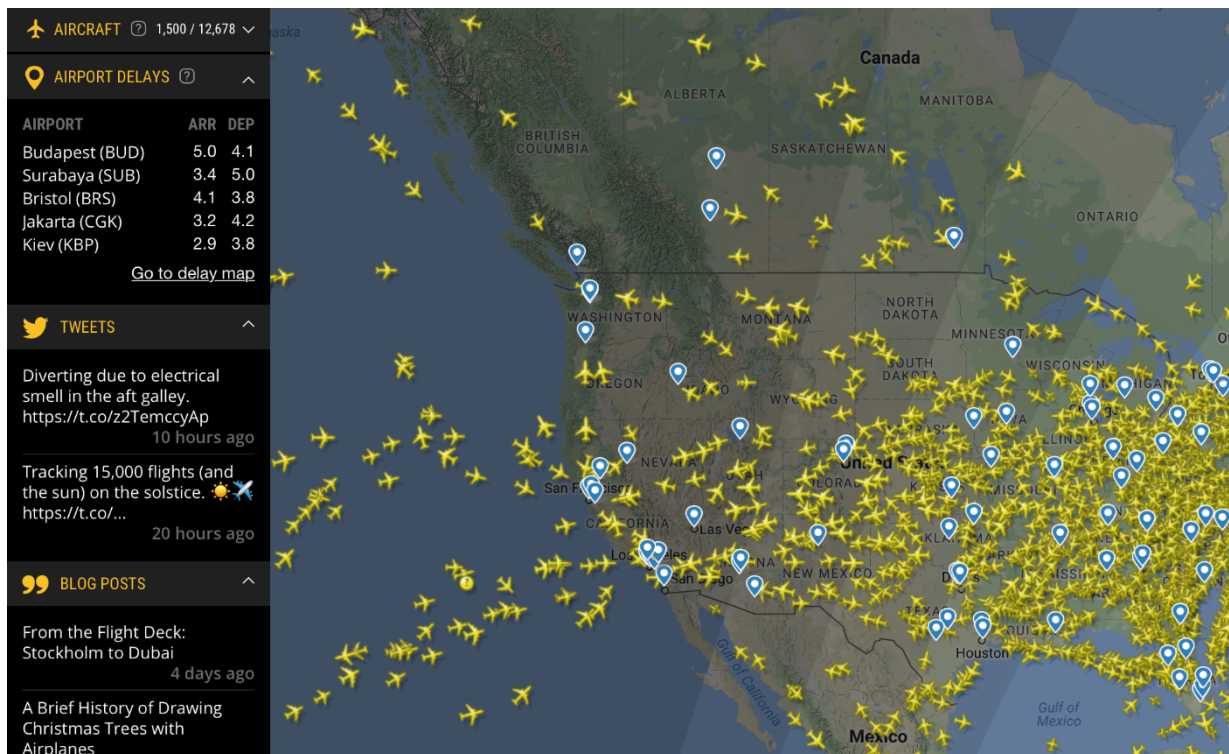
---

Data visualization – Process book – fall 2017

Kévin Brou Boni, Clément Pavué, Mohamed EL Mehdi Roussaky

# Overview, motivation and target audience

Try to google "Flight tracker" on the internet, you will see a bunch of website with several features. As an amateur, the visualization could be painful. On the one hand, we have beautiful visualization, but on the other hand there is practical visualization. An example below from the website <https://www.flightradar24.com/>.



Most of them are identical, information is not easily accessible and readable. An obstacle course begins to understand, filter then select needed information. We could not criticize these platforms because they are targeting aviation enthusiasts.

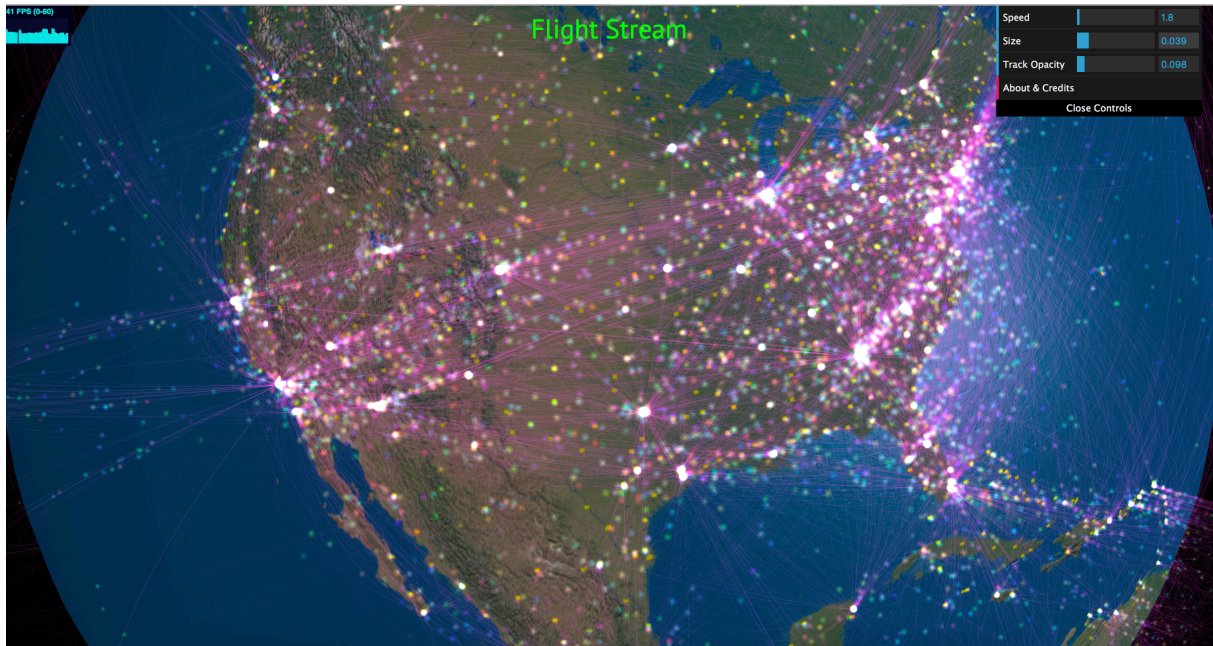
In this project, we are willing to offer an accessible visualization of flights and airports. As a group of amateur, the idea is to quickly explain some facts associated with data visualizations.

The target audience is not only aviation amateur, it will be a platform where we first tell a story with data visualization. Therefore, all the curious greedy to learn some facts about aviation in general are also welcome.

# Related work and inspiration

---

Our goal is to combine comfort and convenience. The first related work was seen in on the #help channel in slack:

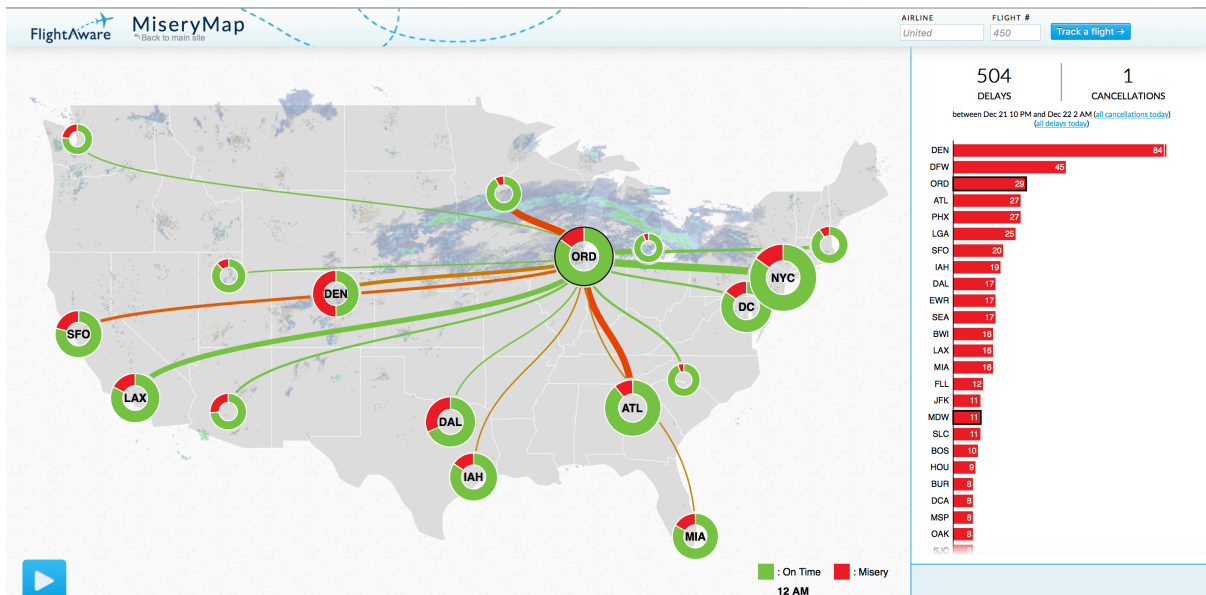


Source: [callumprentice](#)

What we liked about this visualization is that the user has an interface in order to control the flight traffic (speed, size). That is beautiful and enjoyable but not very useful. There is no legend or other information except several dots moving. We can easily understand that they represent an airplane nevertheless we don't any information on their routes.

This map is convenient to have an overview on the flight traffic but in order to find the weather or the current traffic airport it is pointless. We can face it by reducing the number of interactions. Lower the number of airports, combine different visualization tools to save space.

A second inspiration source shows the delays across major US airports:



Source: FlightAware

This visualization has less feature but guarantee a good understanding. Airports are represented by a dot with his code and a gauge showing the delays average. User can select an airport by clicking on it, then some continuous weighted lines representing an air road are displaying.

What we like here is the intuitive experience. No legends or numbers, this visualization explain itself. Moreover, a cloud shape is displaying, a manner to maybe justify delays.

# Questions: What am I trying to show this my viz.?

---

Allow any person to know according to a city where he can travel. A convenient tool displaying the minimum information like the forecast, airlines available and their frequencies.

On top of that, giving tips about airlines like their cruising speeds or the delays in airports.

Our goal is to provide a minimalist tool, it is not a 24 hours live tracker. We want through the visualizations giving the whys and the wherefores of United States history in aviation

# Dataset: where does it come from, what are you processing steps?

---

Our dataset comes from [Kaggle](#), upload by Department of Transportation. Summary information on the number of on-time, delayed, canceled, and diverted flights is published in DOT's monthly Air Travel Consumer Report and in this dataset of 2015 flight delays and cancellations.

The dataset contains 3 files, the first rows below.

- [airlines.csv](#):

IATA_CODE	AIRLINE
UA	United Air Lines Inc.
AA	American Airlines Inc.
US	US Airways Inc.
F9	Frontier Airlines Inc.
B6	JetBlue Airways

- airports.csv:

IATA_CODE	AIRPORT	CITY	STATE	COUNTRY	LATITUDE	LONGITUDE
ABE	Lehigh Valley International Airport	Allentown	PA	USA	40.65236	-75.44040
ABI	Abilene Regional Airport	Abilene	TX	USA	32.41132	-99.68190
ABQ	Albuquerque International Sunport	Albuquerque	NM	USA	35.04022	-106.60919
ABR	Aberdeen Regional Airport	Aberdeen	SD	USA	45.44906	-98.42183
ABY	Southwest Georgia Regional Airport	Albany	GA	USA	31.53552	-84.19447

- airlines.csv:

YEAR	MONTH	DAY	DAY_OF_WEEK	AIRLINE	FLIGHT_NUMBER	TAIL_NUMBER	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCHEDULE
2015	1	1	4	AS	98	N407AS	ANC	SEA	
2015	1	1	4	AA	2336	N3KUAA	LAX	PBI	
2015	1	1	4	US	840	N171US	SFO	CLT	
2015	1	1	4	AA	258	N3HYAA	LAX	MIA	
2015	1	1	4	AS	135	N527AS	SEA	ANC	

This dataset is clean, moreover there are much more features needed. Relatively new, we have still to reduce dramatically his size (more than 700MB). The preprocessing steps:

-**Explanatory analysis** (python) in order to estimate the possibility of reach our goal.

-**Reducing and splitting the data** (python) in smaller csv file. This step was mandatory in order to decrease the computing resource so time in the navigators.

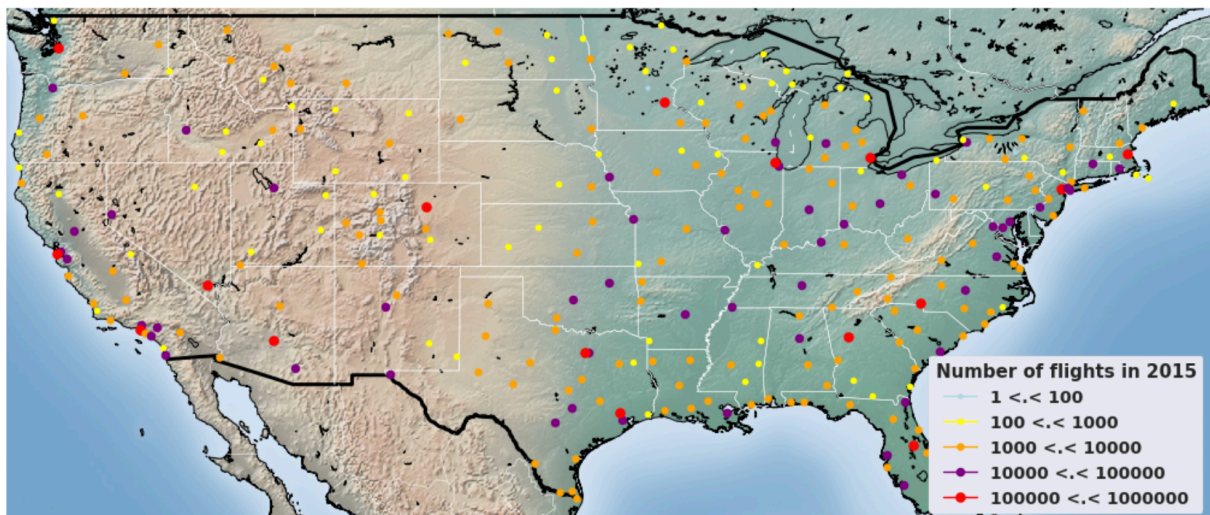
-**Add some features** (python) like the weather thanks to the [Dark Sky API](#).

# Explanatory Analysis: What visualization have you used to gain insights on the data?

---

We dramatically decrease the computing time by only taking in account 2 months of data. Previously we verified that the distribution is followed.

First of all, we had to draw a US map showing the airports and their number of flights in 2015:



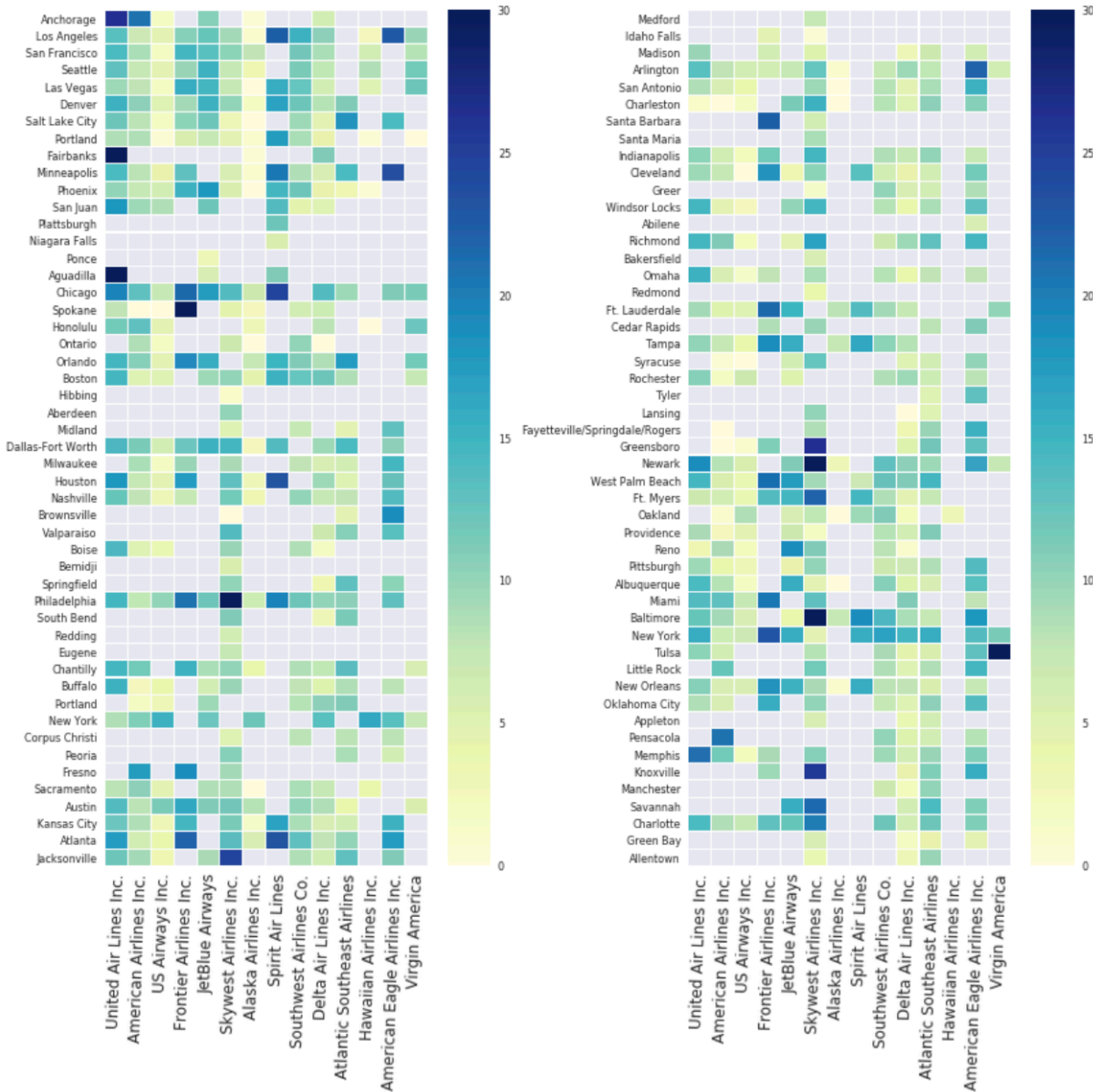
*Jupyter Notebook*

There is a huge number of flights in US. We decided to divide airport in 3 categories:

- L: more than 200 flights a day.
- M: between 200 and 50 flights a day.
- S: less than 50 flights a day.

In a second part, we would know how airlines are responsible of delays. There were too many airports for a basic plot, the best decision was to plot a heat map.

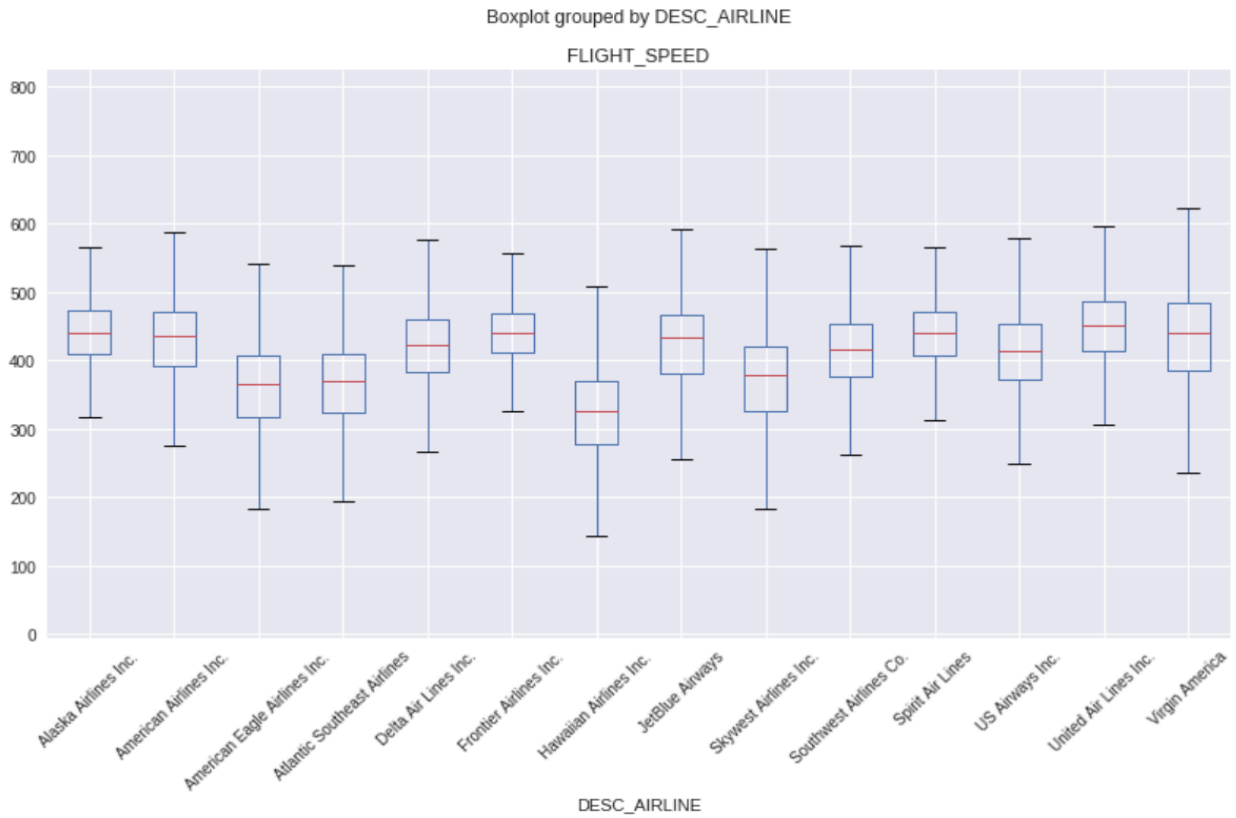
Scale of Delays from origin airport



*Jupyter Notebook*



We raised that even with a departure delay some flights always arrived on time. The average speed is higher from a normal flight to make up for lost time. The idea of going deeper in the cruising speed came. Below a box plot of speed (mph) by airline.



*Jupyter Notebook*

# Implementation:

---



**Fig. 1: Scroll navigation**

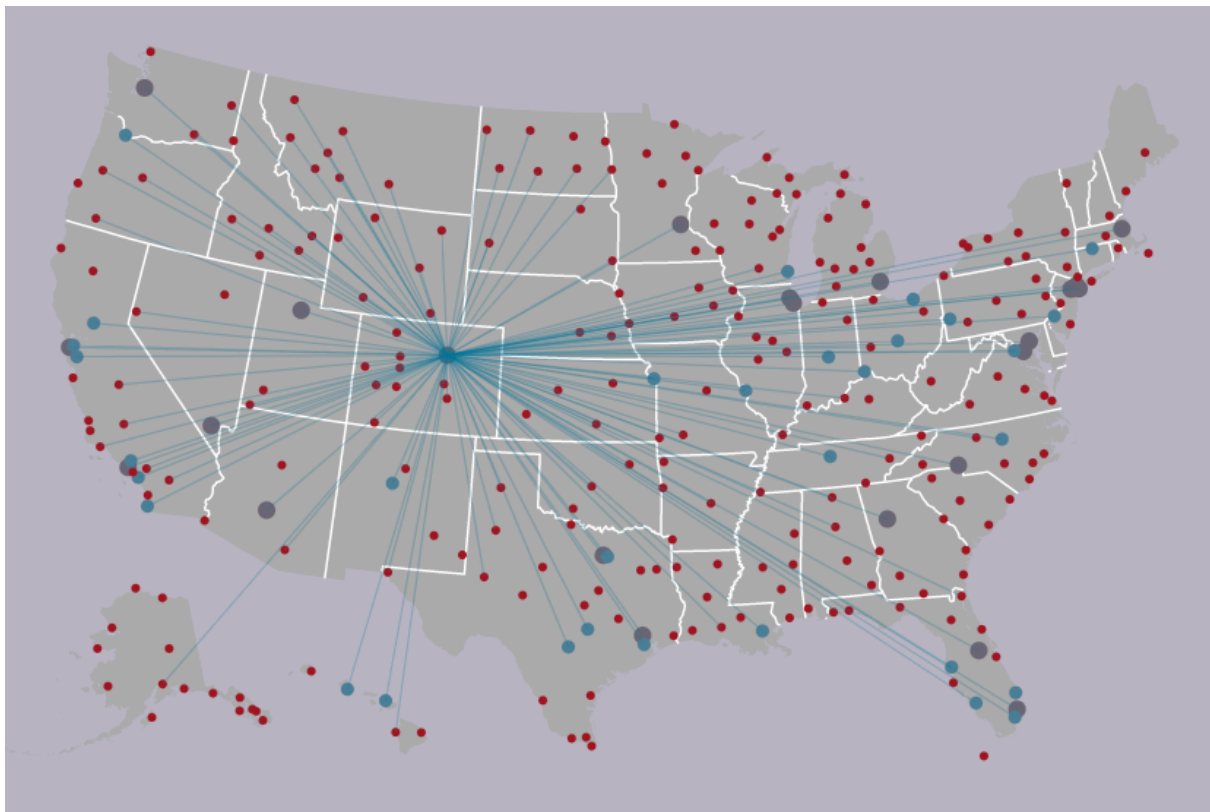
Our website is built like a book, there is chapter. You don't navigate by scrolling but by turning a page. It is a continuous flow, on each page you can read a small story before interacting with the visualization. You are guided through the process.

You can navigate either using the scroll bar on **(Fig. 1)** or by selecting in the menu on the top right corner **(Fig. 2)**.



**Fig. 2: Menu navigation**

**Fig. 3: Map**



Each dot represents an airport. There are three types as explained above, click on one to display his information on the right side (**Fig. 4**).

**ST LOUIS (MO)**

Temperature: 1°C, Partly Cloudy  
Humidity: 65%  
Time zone: UTC-6  
IATA code: STL

**AIRLINE(S) AVAILABLE**

- US Airways Inc.
- American Airlines Inc.
- Southwest Airlines Co.
- Delta Air Lines Inc.
- Skywest Airlines Inc.
- +5 other(s)<sup>?</sup>

**Fig. 4: Airport information**

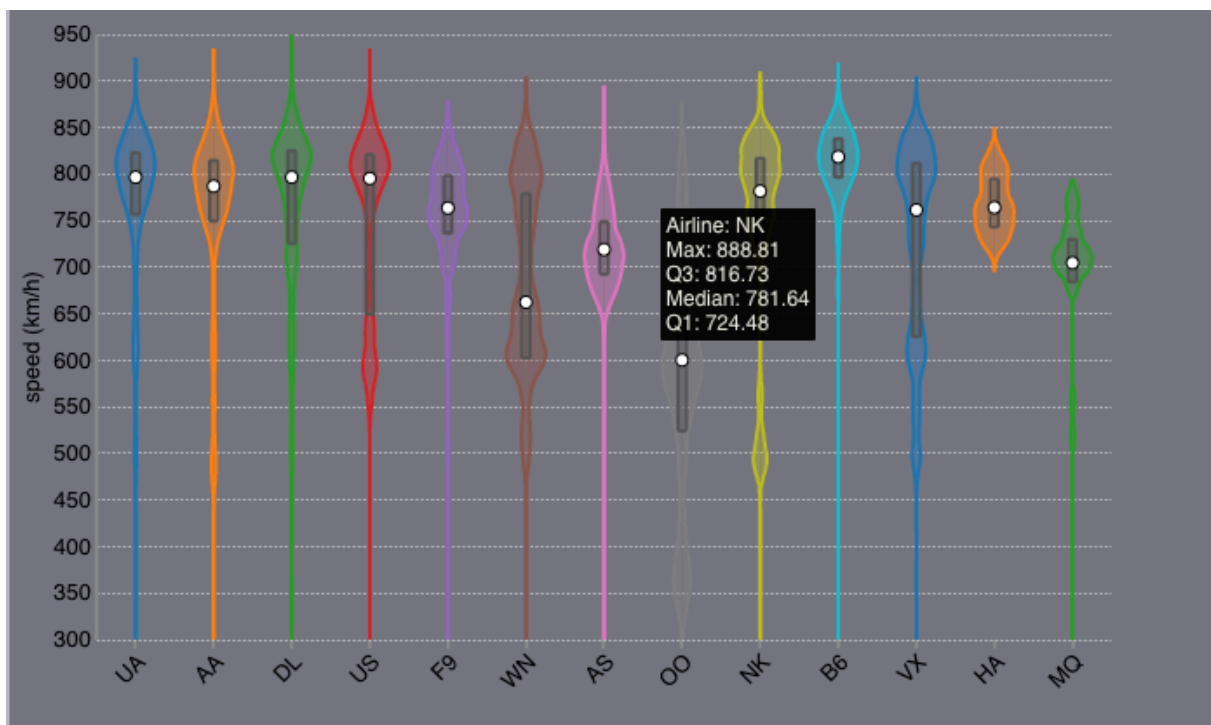
**Fig. 5: Legend**



Click on one circle to filter and select only this type of airport. Click a second time to come back to the initial state.

**Fig. 5: Violin plot**

Speed distribution by airline, a tooltip appears when the mouse come close to a plot.



## Evaluation & peer review:

---

First point, visualization is not that easy. You really have to change your point of view because it can be clear for you but not for others. That is why an outside perspective was needed.

Another important aspect was to make a concession between what we want to display. In our case, several feedbacks were to have a bigger map. It is fine but we have to change the position of the violin plot so stop the interaction between these two. Now we are aware of the difficulties, in our group we disagreed on several points but we manage to have the best user-friendly interface. Everyone work hard for this project but we don't fulfill our goals.

We have lost a lot of time due to the data preprocessing. And sometimes for not implementing. The positive side is that it allows us to find the hidden face of airline and airport. Basically, we are proud of our visualizations even if we could not manage to render a heat map for the delays with D3.

In a nutshell, our website is easy to use and allow people to learn a bit or just taking intel. We could improve it by displaying on live data and by adding a heat map and other stories.

# Understand



*goal:* gather, observe, and research available information to find the needs of the user

*artifacts:* design requirements

## 1) identify the challenge & users

generate

*think big! what is the **problem**? **who** is affected by it? what is known/unknown? orient yourself with all of the project's who, what, why, when, & how.*

Flight traffic website are too complex to use or the contrary too superficial. Useless for amateur.



## 2) find questions & tasks

*what can you **ask** about the challenge? what do users want to do with data? think high and low level. revisit this worksheet to break these down further.*

How to bring all informations without saturation ?

*!! box #3 may help you revisit this box later*



## 3) check with users or explore data

*users: what did you find out? what sparked curiosity? data: characterize aspects of the data. what is it like?*

- Possibility to add the weather
- Display the delays

*!! get the real data and talk to real users if possible!*



## 4) brainstorm design requirements

*what are recurring trends? what are key design **opportunities**? are there **constraints** worth listing?*

- Several airlines and airports
- Flights are the same
- Data available, easy to plot



## 5) compare and rank design requirements

evaluate

*choose a method for comparison: **pros/cons table**, **rank** based on your findings/user needs/tasks, **cross out** the list based on listed justifications, or **pick top 3** to keep and why. explain and review with a group or partner.*

- 1) Show a understanding map
- 2) Filter and interact with the map
- 3) Display other useful plot

*!! is this the right challenge to tackle? is there enough detail? or too much? too many or not enough requirements? complete this worksheet again to refocus the project.*



# Ideate



goal: generate good concepts and ideas for supporting some of the project's design requirements

artifacts: ideas & sketches

generate

## 1) select a design requirement

how might we address the challenge using the requirement? which questions would a user ask? revisit this worksheet for each important design requirement.

Clickable airport ? Filter ?  
HMW display the information ?

!! revisit this worksheet for all important design requirements for your project



## 2) sketch first idea

show how to address this requirement using an **informal sketch** - focus on the big idea not the details.



## 3) sketch another idea

try another **sketch**, think of a new perspective, be different, do not build off of your previous sketch.



## 4) sketch a final idea

think of a different abstraction. challenge constraints and assumptions to **draw** something new or surprising.

!! is three enough? not always. have other ideas? fill out another worksheet!



## 5) compare and relate your ideas

evaluate

for each sketch, break apart **what works well (+)** and **what doesn't (-)** in the **table** below. make connections. reflect on best parts. can you **combine ideas**? review the table with a partner or group.

	sketch #1	sketch #2	sketch #3

!! combining ideas and sketches is not easy. sometimes it may open up new possibilities and ideas - guess what, ideate again!



# Make



*goal:* concretize ideas into tangible prototypes which are approximations of a product in some aspects

*artifacts:* prototypes

## 1) set an achievable goal

generate

*what should the prototype **achieve**? what are the specific **criteria for success**? break a larger goal into parts with clearer feature sets.*

Allow to not lost the user while browsing.  
The user enjoys the navigation.

!! break a goal apart into multiple and create a worksheet for each sub-goal



## 2) plan encodings & layouts

*what are good visualization **encodings** or **layouts** for which data? use the ideas you just came up with, and remember to justify for users and their tasks.*

-Map: legend with 3 bins.  
Continuous lines for the nodes.  
Good color scale  
-Violin plot, Heatmap: not more than 8 features. Good color scale.



## 3) plan support for interactions

*what can the user do? what is required given the chosen encodings? **justify** your design decisions.*

He can display the category of airport he want.  
Visualize the possible routes.  
Can display more information if needed.



## 4) sketching additional views

*what other parts of the data must be seen? brainstorm how to show this data in the tool.*

Delays by airport or airlines  
Cause of delays, cancellation.

!! if you are thinking up new ideas to visualize, go back to the Ideate activity!



## 5) build the prototype and check-in

evaluate

*are your **goals met** by the prototype? test with users if possible. are design decisions properly justified? do any need to be revisited? were any new constraints or limitations discovered? write down your progress and additional justifications below. review this progress and the prototype with a partner or your group.*

-: not current data, more interactions, no immersion

!! did the prototype meet its goal/s? measure its success. make sure you have addressed the design requirement. does the prototype try to do too much?





# Deploy



goal: bring a prototype into effective action in order to support real world users' work & goals

artifacts: visualization system

generate

## 1) pinpoint a target audience

who are you deploying to? what are their **goals**? what will qualify this deployment as a success?

Curious people and aviation amateur.

!! does this audience match your users back on the Understand sheet? if not, revisit previous sheets!



## 2) fix usability concerns

can the tool be **easier to use**? what elements & interactions can be tweaked to avoid frustration?

The goal here is to have the easiest platform with a minimum of interactions

!! is this a new kind of interaction? should you ideate on the idea here instead?



## 3) improve points of integration

integrate data/tools. maximize algorithmic or storage efficiency. how does this fit in a user's workflow?

- Reduce the computation time to enhance the user experience.
- Integrate rare visualization tool in order to keep the user.



## 4) refine the aesthetics

is the use of color and typography consistent? what about the layout or use of whitespace? make it look pleasing!



## 5) consider a method to evaluate your system

evaluate

take a look at the provided supplement of possible methods. how would you test your system? what would be a successful test of this system? write an evaluation plan here. talk through this plan with a partner or your group. if you have time: test with one or more users, summarize your findings, insights, and recommendations below.

!! did any of the usability, integration, or aesthetic changes result in new ideas or requirements? revisit earlier worksheets as needed!

