

# Dokumentation – Results, Goals, Abnormalities

## Inhalt

Allgemein.....	2
Grundidee:.....	2
Ziel der Visualisierung: .....	2
Ideen zur Visualisierung: .....	3
FEEDBACK: .....	3
Features:.....	3
Must-Have: .....	3
Nice-to-have: .....	3
Not-planned:.....	3
Interaktionskomponenten:.....	3
Prototype implementieren:.....	3
FEEDBACK DATEN: .....	4
Daten digital: .....	4
Datenanfragen analog: .....	5
Pairing der Daten:.....	6
Vergleich von Daten: .....	7
Arbeitsaufteilung und Arbeitspakete: .....	8
Features umgesetzt: .....	11
Must-Have: .....	11
Nice-to-have: .....	11
Code Überlegungen.....	12
LineDiagram.....	12
CovidDiagram .....	15
Slider .....	16
Tooltips .....	17
Bubbles .....	19

## Allgemein

**Grundidee:** Visualisierung des Medienkonsums in Deutschland während der Corona Pandemie mit verschiedenen Unterkategorie

### Ziel der Visualisierung:

- Zusammenhänge ersichtlich machen  
zeitlich, in Bezug auf Coronazahlen und des Lockdowns
- Strukturelle Darstellung  
Interaktivität, Verbindung verschiedener Medien-Datensätze
- Gegenüberstellung  
digital (Games, Streaming, social Media) vs. analog (Puzzle, Fernsehen, Bücher)

Auswirkung von Corona auf die Nutzung von Medien in Deutschland

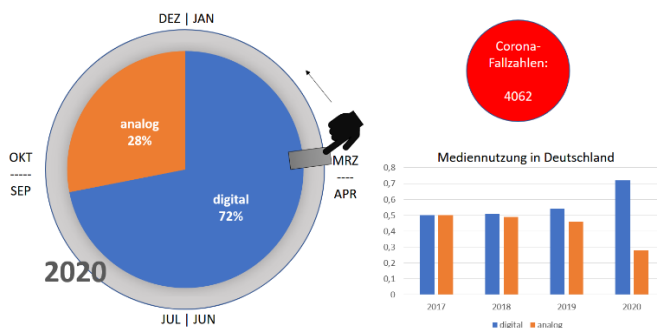


Abbildung 1: Grundidee der Visualisierung

Aber:

In Kuchendiagramm kann keine Aussage über Veränderung des Medienkonsums allgemein getroffen werden, sondern nur die anteilige Veränderung

⇒ Bubbles statt Kuchendiagramm!

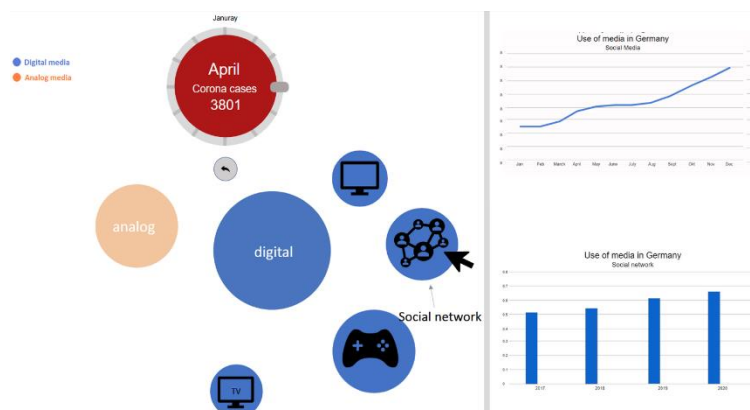


Abbildung 2: Überarbeitetes Konzept

### Ideen zur Visualisierung:

- Zoom der Bubbles über Mausrad
- Prozentanzeige durch Mausbewegung über Bubbles
- Bubblegröße gibt Auskunft über Nutzung im Verhältnis zu den Anderen

### FEEDBACK:

- Achtung, dass es nicht zu viel wird
- Fallzahlen nicht nach Corona-Ampel einfärben → missverständlich  
→ Einfärbung des Sliders während Lockdown + Fallzahlen sind ausreichend
- Legende für Symbole

Für analoge Daten: Nutzung über Verkaufszahlen darstellen

### Features:

#### Must-Have:

- Gegenüberstellung von Medienarten (digital/analog)
- Kreisförmiger Zeitregler:  
Corona Fall-Zahlen und Lockdown-Anzeige
- Bubble Diagramm:  
Vergleich und Auswahl der Medienkategorien
- Balkendiagramm:  
Mediennutzung der letzten Jahre
- Liniendiagramm:  
Mediennutzung über die Monate in 2020
- Legende

#### Nice-to-have:

- Genauere Zahlen bei Hovern über Bubbles (z.B. Stundenanzahl)
- Filterfunktion:  
Vergleich mehrerer Unter-Kategorien in einem Balkendiagramm
- 'coole' Animationen

#### Not-planned:

- Gleichzeitiges Anzeigen von digitalen und analogen Unterkategorien (zu viel Information auf einmal)
- zu viele Bubbles auf einem Screen (unübersichtlich)
- Corona Ampel

#### Interaktionskomponenten:

- Kreis-Slider
- Klicken auf Bubbles für Details (Diagramme)
- Hovern für genauere Infos (z.B. Name der Kategorie)
- "Zurück"-Button

### Prototype implementieren:

Nutzung des Programms UXPin um den Prototype zu erstellen

→ wie funktionieren State und wie kann man den Slider umsetzen?

→ Eher kompliziert das alles miteinander kommuniziert

## FEEDBACK DATEN:

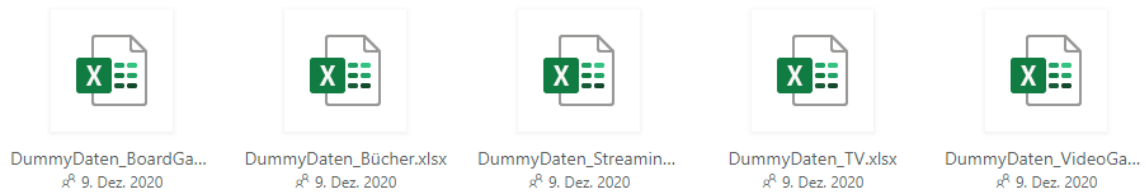
- Daten potenziell eingrenzen (nicht Streaming allgemein, sondern z.B. nur Netflix)
- Statista erhebt Daten nicht selbst, d.h. Quelle ist eine andere; evtl. Schauen, dass man die Quellen bekommt
- Daten anfragen

## Daten digital:

- manchmal monatsweise und manchmal quartalsweise
- manchmal weltweit und manchmal deutschlandweit
- Vorjahre manchmal enthalten, aber immer unterschiedlich
- in 2020 Daten bis maximal September

Online Nutzungsdaten allgemein	<a href="http://ausweisung.ivw-online.de/">http://ausweisung.ivw-online.de/</a>	Monat
Spotify (Anzahl Visits)	<a href="https://de.statista.com/statistik/daten/studie/111224/umfrage/anzahl-der-visits-pro-monat-von-spotifycom/">https://de.statista.com/statistik/daten/studie/111224/umfrage/anzahl-der-visits-pro-monat-von-spotifycom/</a>	Okt 2019 – Sept 2020
Spotify (Anzahl Downloads App)	<a href="https://de.statista.com/statistik/daten/studie/989698/umfrage/anzahl-der-downloads-von-spotify-ueber-den-google-play-store-weltweit/">https://de.statista.com/statistik/daten/studie/989698/umfrage/anzahl-der-downloads-von-spotify-ueber-den-google-play-store-weltweit/</a>	Apr 2016 – Sept 2020
Spotify (Umsatz)	<a href="https://de.statista.com/statistik/daten/studie/812269/umfrage/umsatz-von-spotify-nach-quartalen-weltweit/">https://de.statista.com/statistik/daten/studie/812269/umfrage/umsatz-von-spotify-nach-quartalen-weltweit/</a>	2016 – 2020 (nur Quartale)
Netflix (Anzahl Visits)	<a href="https://de.statista.com/statistik/daten/studie/1021414/umfrage/anzahl-der-visits-pro-monat-von-netflixcom/">https://de.statista.com/statistik/daten/studie/1021414/umfrage/anzahl-der-visits-pro-monat-von-netflixcom/</a>	März 2019 – Okt 2020
Netflix (Abonnenten)	<a href="https://de.statista.com/statistik/daten/studie/196642/umfrage/abonnenten-von-netflix-quartalszahlen/">https://de.statista.com/statistik/daten/studie/196642/umfrage/abonnenten-von-netflix-quartalszahlen/</a>	3. Quartal 2011 – 3. Quartal 2020
Netflix (Gewinn / Verlust)	<a href="https://de.statista.com/statistik/daten/studie/196637/umfrage/gewinn-von-netflix-quartalszahlen/">https://de.statista.com/statistik/daten/studie/196637/umfrage/gewinn-von-netflix-quartalszahlen/</a>	1. Quartal 2009 bis zum 3. Quartal 2020
Netflix (Umsatz)	<a href="https://de.statista.com/statistik/daten/studie/196629/umfrage/umsatz-von-netflix-quartalszahlen/">https://de.statista.com/statistik/daten/studie/196629/umfrage/umsatz-von-netflix-quartalszahlen/</a>	1. Quartal 2009 bis zum 3. Quartal 2020
Mediennutzung allgemein (tägliche Nutzungsdauer)	<a href="https://de.statista.com/statistik/daten/studie/165834/umfrage/taegliche-nutzungsdauer-von-medien-in-deutschland/">https://de.statista.com/statistik/daten/studie/165834/umfrage/taegliche-nutzungsdauer-von-medien-in-deutschland/</a>	2020
Spielkonsolen (Verkaufszahlen)	<a href="https://de.statista.com/statistik/daten/studie/311554/umfrage/weltweiter-absatz-der-playstation-4-und-xbox-one-pro-monat/">https://de.statista.com/statistik/daten/studie/311554/umfrage/weltweiter-absatz-der-playstation-4-und-xbox-one-pro-monat/</a>	Weltweit, Nov 2013 - Sept 2020
Anzahl Steam Users	<a href="https://www.statista.com/statistics/308330/number-stream-users/">https://www.statista.com/statistics/308330/number-stream-users/</a>	Jan 2013 - Sept 2020
Umfrage Mediennutzung allgemein	<a href="https://de.statista.com/statistik/daten/studie/1110488/umfrage/aktivitaeten-beim-zuhausebleiben-wegen-der-covid-19-corona-pandemie/">https://de.statista.com/statistik/daten/studie/1110488/umfrage/aktivitaeten-beim-zuhausebleiben-wegen-der-covid-19-corona-pandemie/</a>	(Stand: 31.05.2020)
(Mehr Quellen auf unserem Drive-Ordner)		

→ Um die Implementierung weiter voranzubringen: vorübergehende Nutzung von Dummy-Daten:



Datenanfragen analog:

### Zeitung

SZ	Keine Antwort	
taz	Keine Antwort	
tz/münchner merkur	<a href="https://www.ivw.eu/aw/print/qa/titel/2279">https://www.ivw.eu/aw/print/qa/titel/2279</a>	Quartal

### Buchverlag

Piper	Keine Antwort	
dumont		
⇒ Börsenverein des Deutschen Buchhandels	monatlich Daten zur Buchmarktentwicklung im Vergleich zum Vorjahresmonat: <a href="https://www.boersenverein.de/markt-daten/marktforschung/branchen-monitor-buch/">https://www.boersenverein.de/markt-daten/marktforschung/branchen-monitor-buch/</a>  Sonderanalyse zur Entwicklung des Leseverhaltens in der Corona-Krise: <a href="https://www.boersenverein.de/markt-daten/marktforschung/studien-umfragen/lesen-in-der-corona-krise-2020/">https://www.boersenverein.de/markt-daten/marktforschung/studien-umfragen/lesen-in-der-corona-krise-2020/</a>	Monat
Ullstein		
⇒ Gesellschaft für Konsumforschung		
⇒ AGF		
Rowohlt		
⇒ mediaControl	<a href="https://www.agf.de/daten/tvdaten/marktanteile/">https://www.agf.de/daten/tvdaten/marktanteile/</a>	
⇒ AGF	<a href="https://www.agf.de/daten/tvdaten/verweildauer/">https://www.agf.de/daten/tvdaten/verweildauer/</a> <a href="https://www.agf.de/daten/tvdaten/sehdauer/">https://www.agf.de/daten/tvdaten/sehdauer/</a>	

### Spieleverlag

Sphinx	Keine Antwort
amigo	Geben keine Daten raus
Franjos	Geben keine Daten raus
Ravensburger	Geben keine Daten raus
Fotopuzzle	Geben keine Daten raus

### Alles Medien

Deloitte Deutschland	Geben keine Daten raus, nur Pdf
Havas Media Germany	Keine Antwort

Konklusion:

Viele konnten keine brauchbaren Daten herausgeben (oft nur Jahres/quartalsweise oder nicht aktuell genug)

→ Umstrukturierung des Projekts aufgrund fehlender Daten

- *Möglichkeit 1:*

Zeitintervall nur quartalsweise, Reduzierung der Kategorien

→ Nicht mehr so gut vergleichbar ob in Lockdown höhere Nutzung

Es fehlen jeweils bei digital, deutschlandweite oder bei analog, weltweite Daten um diese anzugleichen.

- *Möglichkeit2:*

komplett neue Basis für Datensätze suchen

Neues Visualisierungskonzept zu zeitaufwendig, wir wollen unser Konzept beibehalten

→ Es geht um Visualisierung und nicht um Datensuchen

→ Visualisierung lieber polieren als gesamte Zeit in Datensuche zu stecken

## → Änderung des Konzepts:

Statt allgemeiner Vergleich von analog und digital, Vergleich von analogen und digitalen Zeitungen/Zeitschriften

⇒ „analoge/digitale Pressennutzung während Covid-19“

Hierbei jeweils verschiedene Zeitungen (evtl. aus verschiedenen Kategorien?) als Unterbubbles

⇒ Daten:

(digital) <http://ausweisung.ivw-online.de/>

(normal) <https://www.ivw.eu/aw/print/qa/>

→ die Unsicherheit zur Umstrukturierung unseres Konzepts führte zu einem enormen Zeitverlust (einige Wochen, lediglich Brainstorming, warten auf Antworten auf Anfragen der Quellen und suchen nach neuen Daten für anderes Konzept)

Pairing der Daten:

Analog	Digital
Gesundheit	
Apotheken-Umschau	Apothekenumschau.de
Nachrichten	
Süddeutsche Zeitung	Süddeutsche.de
Freizeit	
burda style	BurdaStyle.de
Sport	
Klettern	Bergsteiger.de

### Vergleich von Daten:

Wie könnte man die ausgewählten Daten am besten miteinander vergleichen, welche Diagrammtypen wollen wir?

#### 1. Option:

Allgemein: Liniendiagramm mit Coronazahlen

Durch Anklicken von Bubble werden die Daten zu Kategorie im Liniendiagramm hinzugefügt.  
Durch erneutes Anklicken wird Datensatz wieder herausgenommen.

→ zwei y-Achsen nötig: Coronazahlen | Zugriffe

#### 2. Option:

Oben: Coronadiagramm einzeln

Unten: Diagramm zum Vergleich

#### 3. Option

Oben: Liniendiagramm von Coronazahlen plus Zeitungen zum vergleichen

Unten: Vergleich von ausgewählten Bubbles über Jahre/Monate

#### 4. Option

Nutzer Auswahl anbieten, ob Vergleich gezogen werden soll oder nicht

**→ Auswahl der 2. Option, da die Datensätze evtl. unterschiedliche Y-Achse haben könnte es zu Verfälschungen kommen oder zu unübersichtlich werden**

## Arbeitsaufteilung und Arbeitspakete:

Die Aufteilung ist auch im Gantt-Diagramm sichtbar.

Insgesamt hat jeder auch mal woanders ausgeholfen, falls Probleme auftraten oder eine zweite Meinung gebraucht wurde.

### Feature Slider: Miriam

- In d3 existiert kein runder Slider weswegen dieser von Grund auf implementiert werden musste.
- als ersten umgesetzt, da viele andere Features vom Slider abhingen
- Kreise definieren und bei Verschiedenen Mouse-events jeweils die aktuelle Position den Sliders berechnen
- Für die Monate eine Skala benutzen und diese in einem Kreis anlegen und für jeden Tick den Wert im Bogenmaß im Kreis ausgerechnet
- Um den Slider und den Monat 'snappen' zu lassen wird in drag-end jeweils das aktuelle Bogenmaß des Slider-handels berechnet, verglichen was der nächste Monats-Tick ist und den Handle zu diesem gesetzt
- Monatsanzeige und Coronadaten mit in die Slider-Anzeige eingebunden

### Feature Bubbles: Anna

- Zur besseren Übersichtlichkeit, Bubbles Radius in bestimmtem Bereich (30,100) berechnet
- Bubbles erstellt mit dynamischer Anpassung der Bilder und Schrift, sowie der Bubbles zueinander
- Icons gezeichnet, damit keine Probleme mit Nutzung und passendes Design gegeben
- Verbindung der Bubbles mit Slider und Daten über aktuellen Monat
- Clickhandling mit Liniendiagramm (Hervorhebung der Bubbles bei Selektierung und Reset bei Deselektierung) und Slider (selektierte Bubbles gemerkt und Deselektierung möglich)
- Bubbles angepasst, für Fall, dass keine Daten vorhanden (ausgegraut und gleiche Radi)
- Implementierung des Mousehouvering, dann ersetzt durch Tooltips, siehe unten

### Feature Covid-Diagramm: Klaus

- Das Covid-Diagramm gibt eine Übersicht über den Verlauf Gesamtfallzahlen der Monate des Jahres 2020
- Die x-Achse zeigt die Monate
- Die y-Achse die Summe der Fallzahlen der vorherigen Monate + aktueller Monat (in 1000den)
- Durch Auswahl eines Monats via Slider wird der jeweilige Monat im Diagramm mit einem Balken markiert
- Initialisierung der Daten erfolgt aus einer Variable (dict)

### Feature Linien-Diagramm: Manuela, Klaus

- Je nachdem welche Bubble ausgewählt ist/sind, das entsprechende Datenfile einlesen und entsprechende Daten zwischenspeichern zum späteren Zeichnen
- Bei Initialisierung und wenn keine Unterbubble ausgewählt, wird Diagramm von analog und digital gesamt gezeichnet → alle Datenfiles lesen und Datenpunkte aufsummieren
- x-Achse aus den Monaten generieren
- y-Achse aus den Werten die anzuzeigen sind
- Titel des Diagramms aus Titel/Bezeichnung des Datensatzes



### **Front-End und allgemeine Verbesserungen: Lisa**

- Sichtbarkeit des Buttons für das Zeigen des Popups anpassen (Verstecken während das Popup geöffnet ist)
- Generelles Aussehen (runde Ecken für die verschiedenen Divs)
- Farbauswahl für die beiden Hintergrund-Farben (2 verschiedene Blautöne), für die Bubbles (hellblau für digital und orange für analog), für den Zeitregler und die Tooltips (weiß-Ton), Schriftfarben etc.
- Positionierung und Design der Tooltips (runde Ecken, Schriftfarbe, Hintergrundfarbe, Text-Alignment)
- Anpassungen von Margins etc. im allgemeinen Layout
- Textanpassungen und Rechtschreib-Korrekturen allgemein

### **Tooltips Bubbles: Miriam, Lisa**

- verschiedene Ansätze ausprobiert, viele haben nicht funktioniert und die Mausposition musste unabhängig von der relativen Mausposition errechnet werden
- Laden und Speichern von den jeweiligen relevanten Titeln und Zahlen für den ausgewählten Slider Monat

### **Tooltips Linediagramm: Manuela, Lisa, Miriam, Klaus**

- Für die Tooltips des Line-diagramms haben wir viele Versuche ausprobiert, aber dadurch das für jede Kategorie neue Daten eingelesen wurde führte es immer zu Bugs oder Nicht-Anzeigen der Tooltips
- Lösung: Tooltip nur für gehighlighteten Monat

### **Tooltips Coviddiagramm: Manuela**

- Tooltips bei gehighlightetem Monat wie bei Linediagramm

### **Overall Layout + Legende + Jahreswechsel: Miriam**

- Positionierung zunächst über Flexboxen
- auch verschieden Ansätze ausprobiert und viel Try-and-Error um das Ganze zu formatieren und auf eine Seite zu bekommen
- Wenn das Fenster zu klein wird, wird automatisch alles untereinander gezeichnet
- um das ganze reaktiver zu gestalten: Wechsel zu Grid-Layout mit verschiedenen Grid-areas
- Legende in Grid-Area platzieren und einfügen, Zeilenanbruch anpassen und reactive testen
- Intro-Slide erstellt mit einleitendem Text

### **Popup mit Informationen zu den Daten: Manuela**

- Buttons zum Öffnen und Schließen eines Popups, das Infos zu den Daten liefert
- Positionierung in grid-area: bubbles
- Weiterführende Links zu den jeweiligen Seiten

### **Scroll-Funktion: Lisa**

- Scroll-Funktion, wenn im Intro auf den Pfeil geklickt wird -> schnelleres Anzeigen der Visualisierung als beim manuellen Scrollen

### **Suche nach digitalen Daten und Erstellung von Dummy-Daten: Lisa**

- Sammeln von "digitalen" Datensätzen -> vorwiegend auf Statista, jedoch ohne Abo kein Zugang, große Unterschiede in den gefundenen Daten (bzgl. Zeitachse etc.)
- Dummy-Daten als vorübergehende Lösung, um mit Datensätzen weiterarbeiten zu können bis die "echten" Daten einsatzbereit waren (erstellt mit Excel nach dem Vorbild anderer Datensätze)

### **Daten sammeln und formatieren: Manuela**

- Viele Anfragen → kaum Daten
- Nach Umstrukturierung Auswahl von Datensätzen für digital und analog zu den Kategorien (Gesundheit, Nachrichten, Freizeit, Sport)
- Digitale Daten: Umwandeln der vorhandenen csv zu Tabelle, die ausgelesen werden kann
- Analoge Daten: Aufspalten der Quartalswerte in Monate; entweder Wert/3 oder auch zwischen den einzelnen Wert/3 zweier Quartale interpoliert → Nutzung der interpolierten Werte

### **Versuch: Resizing Window -> Resizing Diagramme: Miriam**

- Tooltips von Bubbles werden abgeschnitten
- Tooltips von Diagramm nicht mehr sichtbar

## Features umgesetzt:

### Must-Have:

- Kreisförmiger Zeitregler: Corona Fall-Zahlen und Lockdown-Anzeige
- Bubble Diagramm: Darstellung und Vergleich der Unterkategorien
- Liniendiagramm für Corona Fallzahlen
- Liniendiagramm zum Vergleich: Pressenutzung über die Monate in 2020
- Legende

### Nice-to-have:

- Tooltips: Genauere Information bei Hoover über Bubbles
- Filterfunktion: Auswahl von Unterkategorien und Anzeige in Liniendiagramm

Nicht umgesetzt: ,coole' Animationen:

→ Entscheidung gegen Animationen, wegen potenzieller Überladung der Visualisierung

Nach Scroll-Funktion der Intro-Slide wird der Nutzer nicht mit zu vielen überflutet

Außerdem Fokus auf Hover- und Highlight Funktionalitäten

### Feature table:

Subject	Feature	To-Do
Intro-Slide	Implementierung	done
Intro-Slide	Scrollfunktion	done
Popup (Infos über Daten)	Implementierung	done
Popup (Infos über Daten)	Button-Funktionalität (Öffnen, Schließen)	done
Kreis-Slider	Funktionalität des Zeitreglers	done
Kreis-Slider	Beschriftung des Zeit-Reglers	done
Kreis-Slider	Lockdown-Anzeige	done
Kreis-Slider	Toggle zwischen den Jahren	done
Bubble-Diagramm	Grundfunktionalität	done
Bubble-Diagramm	Verknüpfung von Bubble- und Linien-Diagramm	done
Bubble-Diagramm	Hovering für Tooltips	done
Bubble-Diagramm	Verknüpfung mit Slider	done
Linien-Diagramm (Corona-Fallzahlen)	Grund-Funktionalität	done
Linien-Diagramm (Corona-Fallzahlen)	Highlighting des ausgewählten Monats	done
Linien-Diagramm (Corona-Fallzahlen)	Hovering for Tooltips	done
Linien-Diagramm (Corona-Fallzahlen)	Toggle zwischen Gesamt-Fallzahlen und monatlichen Neuinfektionen	done
Linien-Diagramm (Medien-Nutzung)	Grund-Funktionalität	done
Linien-Diagramm (Medien-Nutzung)	Highlighting des ausgewählten Monats	done
Linien-Diagramm (Medien-Nutzung)	Hovering für Tooltips	done
Legende	Implementierung	done

## Code Überlegungen

Wiederkehrende Strings in Konstanten (constants.js)

D3.mouse obsolete → `var mouse = d3.pointer(event, chart.node());`

## LineDiagram

Laden einer csv Datei:

```
d3.csv("data.csv" )

    .then(function(data) {

        console.log("loaded successfully")

    })

    .catch(function(error) {

        console.log("loading error " + error)

    })
```

Laden mehrerer Dateien:

```
Promise.all([

    d3.csv("data1.csv"),

    d3.csv("data2.csv"),

]).then(function(files) {

    var data1 = files[0]

    var data2 = files[1]

    data1.forEach(function (d) {

        var data1quartal = d.Quartal

        console.log("quartal " + data1quartal)

    })

    data2.forEach(function (d) {

        var data2titel = d.Titel

        console.log("title " + data2titel)

    })

}).catch(function(err) {

    console.log("loading error" + err)

})
```

Daten umdrehen:

```
data.slice().reverse().forEach(function (d) {...})
```

Fehler beim direkten Öffnen: "URL scheme must be "http" or "https" for CORS request."

⇒ Nicht beim Starten aus der IDE oder über den Server

csv Dateien müssen durch: ", " getrennt sein nicht mit: " ; "

Wenn man Daten loggt, zeigt die Konsole im Browser zwar ein Ergebnis an, aber das wird später rein geladen, weil gemerkt wurde, dass das da geloggt wurde, aber zu dem Moment wo mans loggt sind die Daten eigentlich noch gar nicht da.

### bubbles

```
var analogSelected  
var digitalSelected
```

Bubble Click :

```
if analog  
  check analogSelected == current Clicked  
    → analogSelected = "" // → remove  
    → analogSelected = currentClicked.label // → add or change
```

```
if digital  
  check digital Selected == current Clicked  
    → digital Selected = "" // → remove  
    → digitalSelected = current Clicked.label // → add or change
```

```
parser (analogSelected, digitalSelected)
```

### lineDiagram

```
function parser (analog , digital)
```

// je nachdem was  
analog  
digital  
entsprechende .csv

```
var analogData = [  
  { ser1: data.Quartal , ser2: Wert },  
  ... für jedes  
]
```

```
visualize line Data (analogData , digital Data)
```

Abbildung 3: Überlegung, wie Kommunikation zwischen Bubbles und Diagramm funktioniert

### diagram

1. if (beide<sup>data</sup> == null) {  
 remove old diagramm  
}
2. if (entweder oder) {  
 remove old diagramm  
 draw new with one line  
}
3. if (both) {  
 remove old diagramm  
 draw new with one line  
 draw new with other lines  
}

Abbildung 4: Erste Überlegungen, was passieren muss, wenn Daten zu zeichnen sind

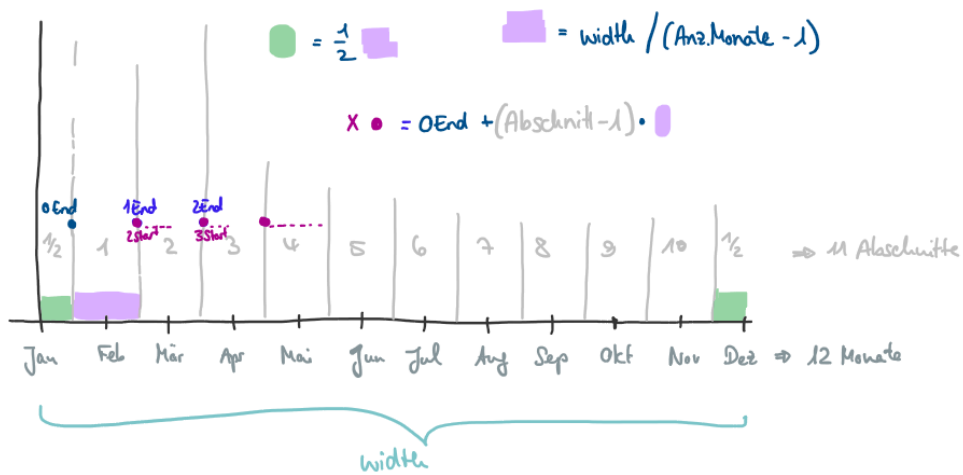


Abbildung 5: Aufteilung des Diagramms zum Highlighten des ausgewählten Monats

## CovidDiagram

### Senkrechte Hilfslinien bei ticks an x-Achse

```
let xAxis = d3.axisBottom()  
  .scale(x)  
  .tickSize(-height);
```

### Horizontale Hilfslinien bei ticks an y-Achse (ohne Linie oben)

```
const makeYLines = () => d3.axisLeft()  
  .scale(y)
```

```
chart.append('g')  
  .attr('class', 'grid')  
  .attr("opacity", 0.4)  
  .call(makeYLines()  
    .tickSize(-width, 0, 0)  
    .tickFormat('')  
  )
```

### Area Graph

```
u.enter()  
  .append("path")  
  .data([data], function(d){ return d.ser1 })  
  .attr("fill", "darkred")  
  .attr("stroke", "none")  
  .attr("fill-opacity", .3)  
  .attr("d", d3.area()  
    .x(function(d) { return x(d.ser1); })  
    .y0( height )  
    .y1(function(d) { return y(d.ser2); })))
```

### Line Graph

```
u.enter()  
  .append("path")  
  .data([data], function(d){ return d.ser1 })  
  .attr("fill", "none")  
  .attr("stroke", "red")  
  .attr("stroke-width", 1.5)  
  .attr("d", d3.line()  
    .x(function(d) { return x(d.ser1); })  
    .y(function(d) { return y(d.ser2); })))
```

## Slider

Wie die Position und der nächste Monats-Tick berechnet wird (slider.js ab Zeile 170):

//Find the Phi degree of the current sliderHandlePosition and compare it to Phi-values of the MarkTicks

```
var cosPhi = (d.x / circumference_r)

var sinPhi = - (d.x / circumference_r)

var phi = 0

if (d.y >= 0) {
    phi = Math.acos(cosPhi)
}

else {
    phi = 2 * Math.PI - Math.acos(cosPhi)
}

//Find the closest tick:
var closestPhi = tickphiright[0];
var diff = Math.abs(phi - closestPhi);
var postion = 0

// between hightst and 0 value it will always take highest and not 0
for (var val = 0; val < tickphiright.length; val++) {
    var newdiff = Math.abs(phi - tickphiright[val]);
    if (newdiff < diff) {
        diff = newdiff;
        closestPhi = tickphiright[val];
        postion = val;
    }
}

//Set/Snap handle to closest TickMark:
d3.select(this)
    .attr("cx", d.x = circumference_r * Math.cos(closestPhi))
    .attr("cy", d.y = circumference_r * Math.sin(closestPhi));
```



## Tooltips

Wie tooltips angezeigt werden:

```
.on("mouseover", function (d) {

    console.log("in mouseover")

    var matrix = this.getScreenCTM()
    // Get the position of the hovered bubbles

    .translate(+ this.getAttribute("cx"), + this.getAttribute("cy"));

    tooltip.transition().duration(200).style("opacity", .9);

    var id = this.id

    tooltip.html(tooltipDetails(id, currentMonth))

    .style("left", (window.pageXOffset + matrix.e + 30) + "px")

    .style("top", (window.pageYOffset + matrix.f - 59) + "px");

})

/* Remove the tooltip */

.on("mouseout", function (d) {

    console.log("in mouseout")

    tooltip.transition().duration(500).style("opacity", 0);

})
```

Wie die Tooltip-details erzeugt werden:

```
//Get the Text for the Tooltip

function tooltipDetails(currentid, currentMonth) {

    var details;

    if (currentid > 2 && currentid < 7) {

        details = ("<b>Kategorie:</b> " + idToLabel(currentid) + "<br><b>Quelle:</b> " +
        titleCollection[currentid] + "<br><b>Verkaufszahlen:</b> " +
        amountCollection[currentMonth][currentid]);

    }

    else if (currentid > 6 && currentid < 11) {
```

```
        details = ("<b>Kategorie:</b> " + idToLabel(currentid) + "<br><b>Quelle:</b>  
" + titleCollection[currentid] + "<br><b>Besuche:</b> " +  
amountCollection[currentMonth][currentid]);  
    } else {  
        details = ("Nothing here");  
    }  
    console.log(details);  
    return details;  
}
```

## Bubbles

### Dynamische Bildanpassung auf Bubbles:

```
//Icons in Bubbles
elemEnter.append("svg:image")

    .filter(function (d) { return d.id > 2 })
    .attr("x", function (d) { return computeImagePos(d.id) })
    .attr("y", function (d) { return computeImagePos(d.id) })
    .attr("width", function (d) { return computeImageSize(d.id) })
    .attr("height", function (d) { return computeImageSize(d.id) })
    .attr("id", function (d) { return d.id })
    .attr("xlink:href", function (d) { return "icons/" + d.img })
    .style("opacity", function (d) { if (currentMonth > 9) { return 0.5 } else { 1 } })
    .on("click", function (d) { return bubbleClick(d3.select(this)) })
```

### Bubble-Clickhandling:

```
function resetBubble(clicked_bubble) {
    elemEnter.selectAll("circle")//.append("circle")
        .filter(function (d) {return (d.id == clicked_bubble) })
        .attr("fill", idToColor(clicked_bubble))
        .style("opacity", function (d) {
            if (currentMonth > 9) {return 0.5}
            else {1}
        });
}

function markMainB() {
    markBubble(1); markBubble(2);
    //demark old_bubbles
    sameClick = 1;
    if (clicked_Analog != 0) { resetBubble(clicked_Digital); }
    if (clicked_Digital != 0) { resetBubble(clicked_Analog); }
    clicked_Analog = 0; clicked_Digital = 0;
    selectedAnalogBubble = ""; selectedDigitalBubble = "";
}
```

```

/* Event handler for mouse click on bubbles */
function bubbleClick(d) {
    bubbleName = idToLabel(idClick);

    //parameter for parser to show diagramm
    var currentLabel = idToLabel(d.attr("id"));
    var currentID = d.attr("id");

    //default case
    if (d.attr("id") == 2) {
        //do nothing, because mainBubbles
        selectedAnalogBubble = ""; selectedDigitalBubble = "";
    }

    //default case
    if (d.attr("id") == 1) {
        //do nothing, because mainBubbles
        selectedAnalogBubble = ""; selectedDigitalBubble = "";
    }

    //bubble-selection
    if (d.attr("id") >= 3 && d.attr("id") <= 6) {
        console.log("analog sub-bubble clicked: " + currentLabel)
        // was selected before, so remove selection
        if (selectedAnalogBubble === currentLabel) {
            selectedAnalogBubble = "";
            markedBA = 0; //saves Bubblesclick before Slidermove
        }
        // add new selection
        else {
            selectedAnalogBubble = currentLabel
            markedBA = currentID; //saves Bubblesclick before Slidermove
        }
    } else if (d.attr("id") >= 7 && d.attr("id") <= 10) {
        console.log("digital sub-bubble clicked: " + currentLabel)
        // was selected before, so remove selection
        if (selectedDigitalBubble === currentLabel) {
            selectedDigitalBubble = ""
            markedBD = 0; //saves Bubblesclick before Slidermove
        }
        // add new selection
    }
}

```

```

        else {
            selectedDigitalBubble = currentLabel
            markedBD = currentID; //saves Bubblesclick before Slidermove
        }
    }

    //Clickhandling, first deselect all Bubbles
    for (var i = 1; i < 12; i++) {
        resetBubble(i);
    }

    //bubble-selection
    if (selectedAnalogBubble == "" && selectedDigitalBubble == "") {
        markedBA = 0; markedBD = 0
        markMainB(); //default Case in Digramm
    } else {
        if (selectedAnalogBubble != "") {
            markedBA = AlabelToId(selectedAnalogBubble)
            markBubble(AlabelToId(selectedAnalogBubble));
        }
        if (selectedDigitalBubble != "") {
            markedBD = DlabelToId(selectedDigitalBubble)
            markBubble(DlabelToId(selectedDigitalBubble));
        }
    }

    parser(selectedAnalogBubble, selectedDigitalBubble) //show diagramm in
lineDiagram.js
}

//Helpmethods for Bubble-Clickhandling
function markBubble(idClick) {
    elemEnter.selectAll("circle")//.append("circle")
        .filter(function (d) {return (d.id == idClick)})
        .attr("fill", function (d) {
            if (idClick > 6) {return "#08456e"; //digital}
            if (idClick > 2 && idClick < 7) {return "#a84d0a"; //analog}
            if (idClick == 1) {return "#a84d0a"; //analog}
            if (idClick == 2) {return "#08456e"; //digital}})

```

```

        .style("opacity", function (d) {
            if (currentMonth > 9) {return 0.5}
            else {1}
        });
    }

```

## Radius berechnen:

```

bubbleRadius.forEach(function (a) {

    var id = a.id
    var month = a.month
    var amount = a.amount
    var title = a.title

    const radiusScale = d3.scaleSqrt()
        .domain([endmin, endmax])
        .range([30, 100]) //push radius in Range, so that bubblesize is ok
    var r = radiusScale(amount)

    radius[month][id] = r //save radius for bubbles in external Array

    //For the Tooltip, save the amount and title to display
    amountCollection[month][id] = amount
    titleCollection[id] = title
})

//compute Main Bubble Radius
function computeMainBubbles(bubbleRadi, month) {
    var summedig = 0;
    for (let bubb = 3; bubb < 7; bubb++) {
        summedig = summedig + bubbleRadi[month][bubb] + 10
    }
    bubbleRadi[month][1] = summedig / 4; //load digital sum in radius

    var summeana = 0;
    for (let bubb = 7; bubb < 11; bubb++) {
        summeana = summeana + bubbleRadi[month][bubb] + 10
    }
    bubbleRadi[month][2] = summeana / 4; //load analog sum in radius
}

```