



Vom Desktop ins Web: Der neue Atlas der Schweiz

Alexander Müdespacher, ETH Zürich

FOSSGIS 2026 · Göttingen · 26. März 2026

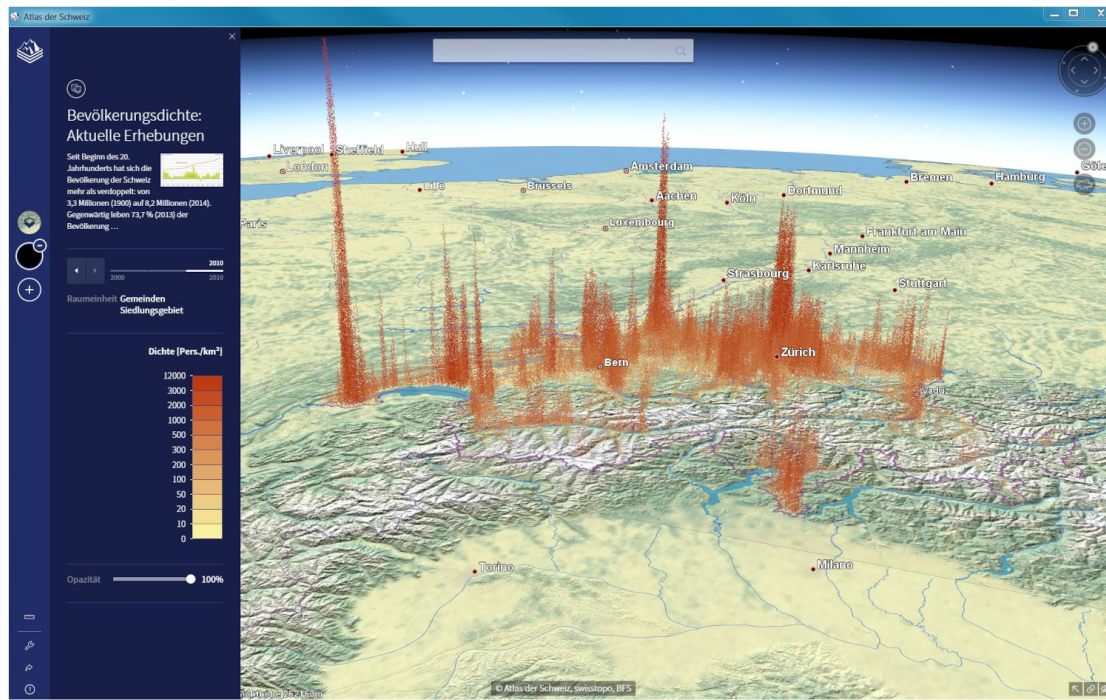
Einführung

Ein Atlas mit langer Geschichte macht den Schritt ins Web



Ausgangslage

Der Desktop-Atlas war technisch innovativ, passt aber nicht mehr zur heutigen Nutzung



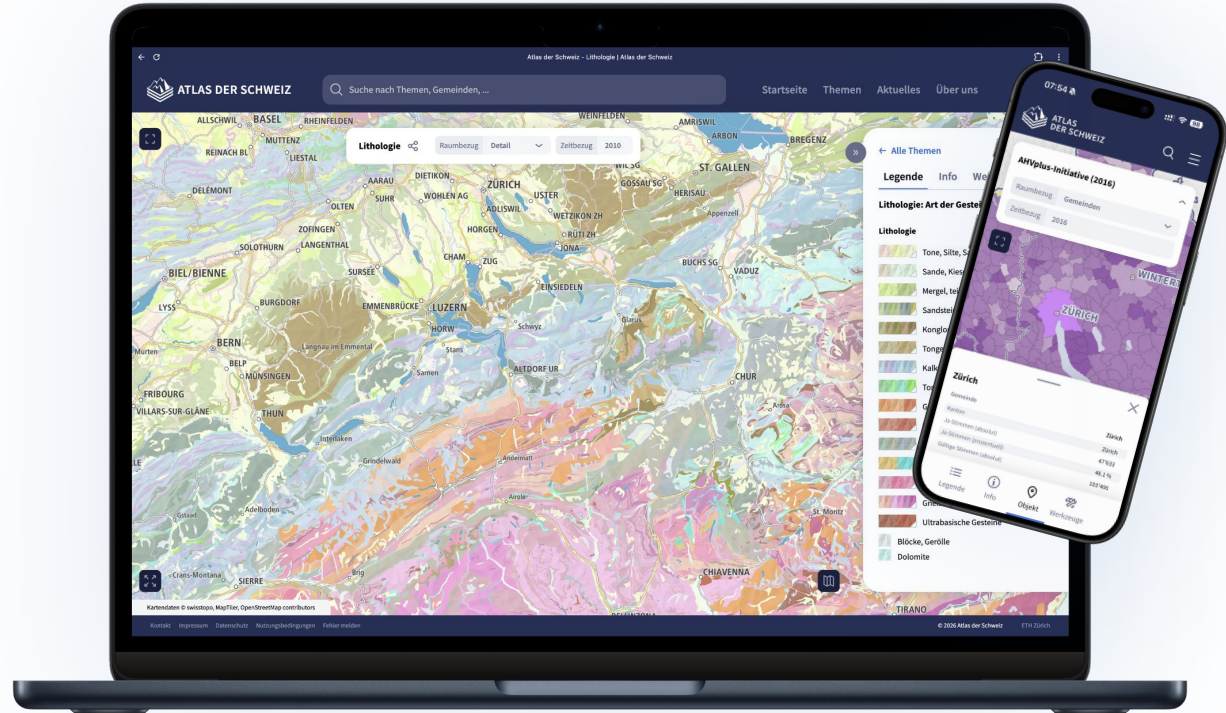
- Desktop-Applikation mit integriertem Chromium, Vue.js-Benutzeroberfläche und osgEarth-Kartenrenderere (C++)
- Virtueller 3D-Globus
- Geo- und Metadaten werden von einem Web-Server live bereitgestellt
- Installierbar auf Windows und macOS
- Inhaltliche Updates: ca. wöchentlich
- Releases: 2-3x jährlich

► Nutzer erwarten schnellen, direkten und webbasierten Zugriff;
ohne Hürden, ohne Installation, jederzeit verfügbar

Zielsetzung

Ein Web-Atlas, der auffindbarer, zugänglicher & zukunftsfähiger ist

- Ohne Installation direkt im **Browser** nutzbar
- Auf **Desktop und Mobilgeräten** verwendbar
- **Kernfunktionen** des Atlas schrittweise ins Web bringen
- **Auffindbarkeit** erhöhen
- Karten **direkt öffnen** und **teilen** können
- **Technische Basis** für weitere Innovation



Vorgehen

Von der Analyse zum “Minimum Viable Product”



► Klares Scoping und harte MVP-Priorisierung waren entscheidend

Analyse

Ein Nationalatlas bedient unterschiedliche Zielgruppen

Studierende



Alter	18 - 26	Zweck	Bildung
Erfahrung in Geo/Kartografie	Mittel	Nutzungsort	Universität zu Hause
Allgemeiner Anwendungsfall	Atlas nach...		

Lehrkräfte



Alter	25 - 65	Zweck	Bildung
Erfahrung in Geo/Kartografie	Hoch	Nutzungsort	Schule, Universität
Allgemeiner Anwendungsfall	Karten im Unterricht verwenden, um Konzepte in Geo- und Umweltstudien zu erklären		
Konkretes Beispiel	Ein Lehrer nutzt den Atlas, um Schülern die Veränderung der Landnutzung in der Schweiz in den letzten 50 Jahren und die städtische Expansion und ihre Auswirkungen zu erklären.		

Schülerinnen und Schüler



Alter	13 - 19	Zweck	Bildung
Erfahrung in Geo/Kartografie	Gering	Nutzungsort	Schule (z.B. iPad) zu Hause
Allgemeiner Anwendungsfall	Kategorien/Themen durchsuchen + Thematische Karten durchstöbern + Karteninformationen lesen + Quellen konsultieren		
Konkretes Beispiel	Die Schülerin erhält im Fach "Geografie" oder "Mensch und Umwelt" die Aufgabe, ein Thema im Atlas der Schweiz zu recherchieren und einen Vortrag zu verfassen. Dazu durchsucht sie den Atlas der Schweiz und taucht bei einem Thema, z.B. "Entwicklung der Nullgradgrenze" tiefer ein.		

Hobbyforscher und Rentner



Alter	18 - 75	Zweck	Freizeit, Selbstbildung
Erfahrung in Geo/Kartografie	Mittel	Nutzungsort	Zu Hause, Unterwegs
Allgemeiner Anwendungsfall	Themen durchstöbern + Daten abfragen und Legende lesen + Karteninformationen lesen + Funktionen ausprobieren		
Konkretes Beispiel	Ein älterer Hobbyforscher ist besonders für Eisenbahnen interessiert, sucht nach herauszufinden und versucht herauszufinden, wie es nicht mehr umge...		

Interessierte Bürger / Allgemeine Öffentlichkeit



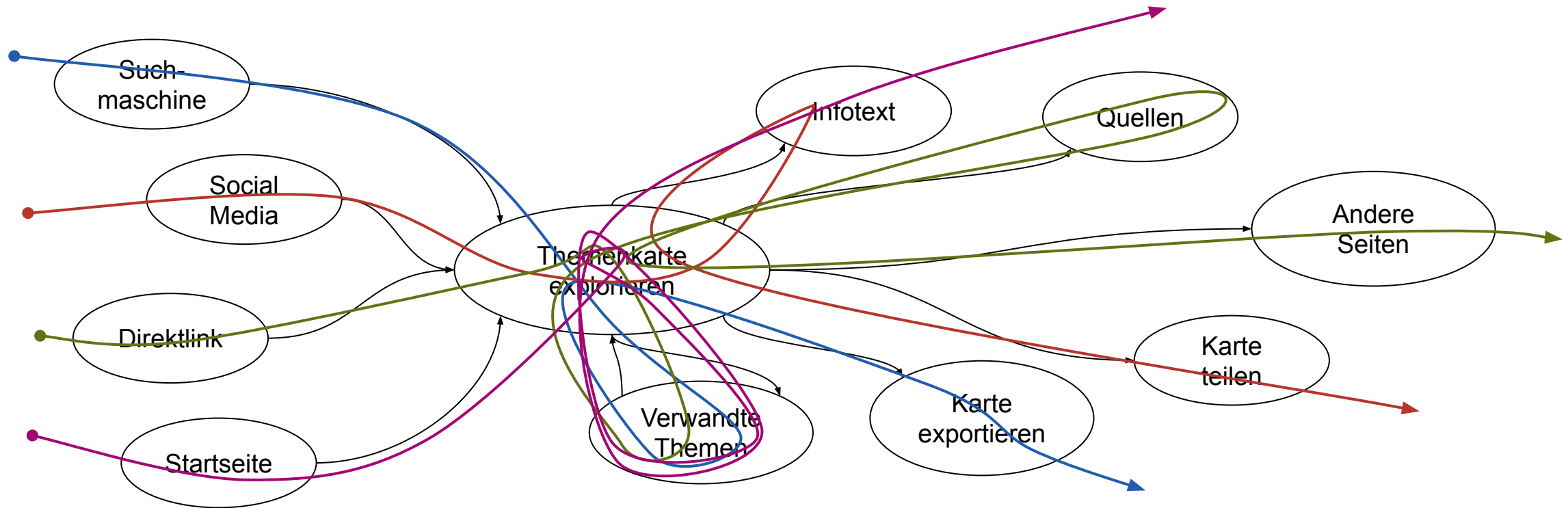
Alter	16 - 65	Zweck	Freizeit
Erfahrung in Geo/Kartografie	Gering	Nutzungsort	Zu Hause, unterwegs
Allgemeiner Anwendungsfall	Thematische Karten durchstöbern + Karteninformationen lesen + Zeitliche Animationen durchführen + Interessante Karten teilen		
Konkretes Beispiel	Eine Person liest auf LinkedIn über die neue Karte der Woche, klickt den Link, erkundet die Karte über "Zustand des Schweizer Waldes", schaut sich die lokale Umgebung an und teilt einen besonders interessanten Ausschnitt der Karte mit Freunden.		

► Definition von Personas hilft, sich in idealtypische Nutzer zu versetzen

Analyse

Nutzer kommen mit verschiedenen Nutzungsszenarien

EXEMPLARISCH

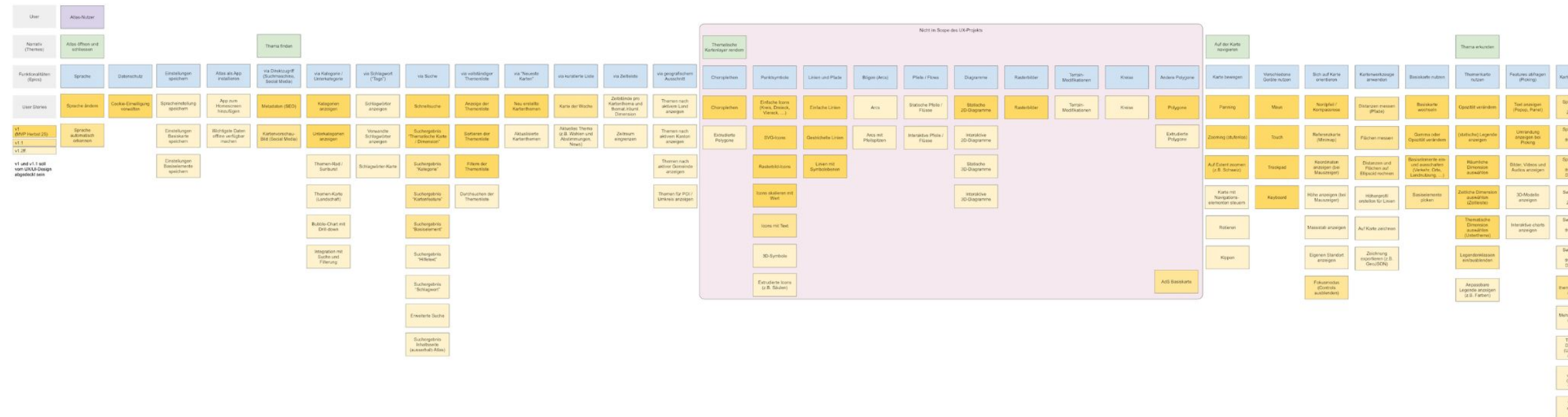
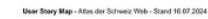


► Aus beispielhaften User Journeys können benötigte Funktionalitäten abgeleitet werden

Analyse

Mit einer User Story Map werden Funktionalitäten priorisiert

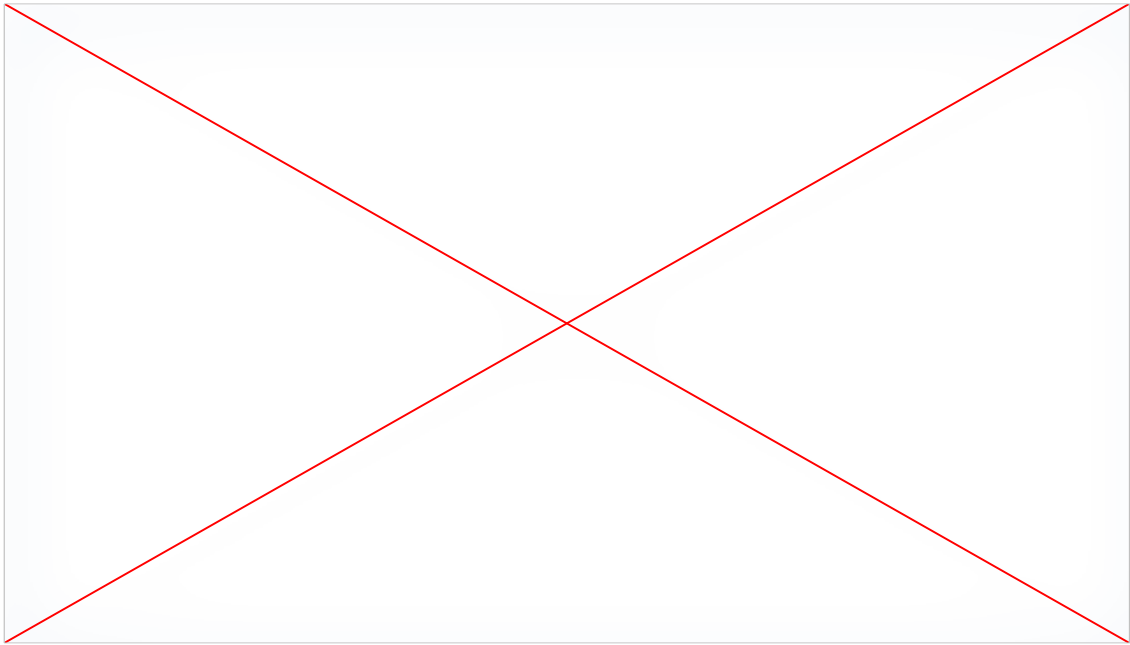
EXEMPLARISCH



- **Wichtigste Unterscheidung: Was muss ins “Minimum Viable Product” (MVP), was kann später kommen?**

Konzept

Mit klickbaren Prototypen kann die Navigation früh getestet werden



- Fokus auf **Informationsarchitektur** statt auf visuelles Design
- Wichtigste User Journeys als **klickbarer Ablauf**
- Nur **wenige Karten** als Mock
- Testen von “**Happy Path**” und “**problematischen Pfaden**”
- Auch: Grosser **Aufwand** für einen guten Prototypen

► Ziel: Nicht viel Zeit und Energie für Design-Fragen verschwenden; Fokus auf das Wesentliche

Konzept

Usability Testing mit idealtypischen Nutzern

Setup

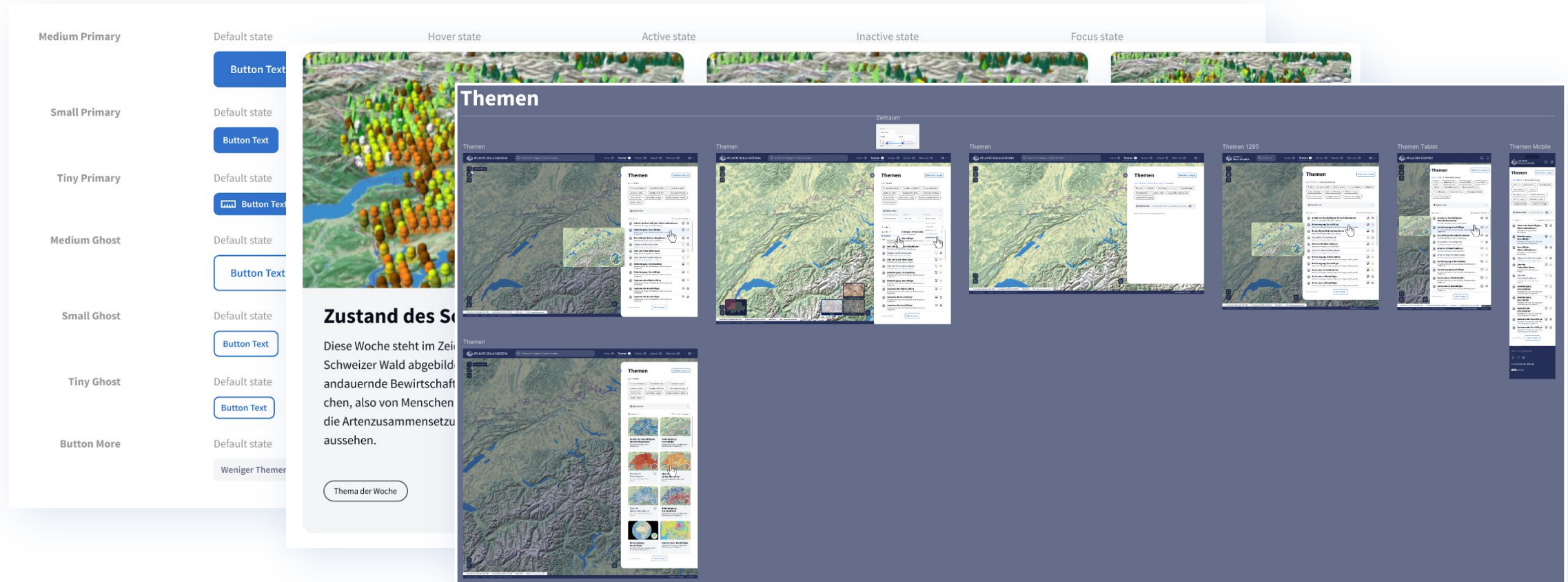
Vorinterview → Testperson löst Testaufgaben (virtuelle Beobachtung) → Nachinterview → Debriefing

Erkenntnisse

- **Vertraute Muster** aus **verbreiteten Kartenanwendungen** wurden am schnellsten verstanden
- **Themenhierarchie** und **Breadcrumbs** gaben gute **Orientierung**
- **Doppelte** Such- und Navigationsmuster **verwirrten** eher
- **Jüngere Nutzer wollten weniger Text** und stärker aufgabennah platzierte Funktionen

► Erkenntnisse aus den Usability-Tests flossen in die zweite Konzept-Iteration ein

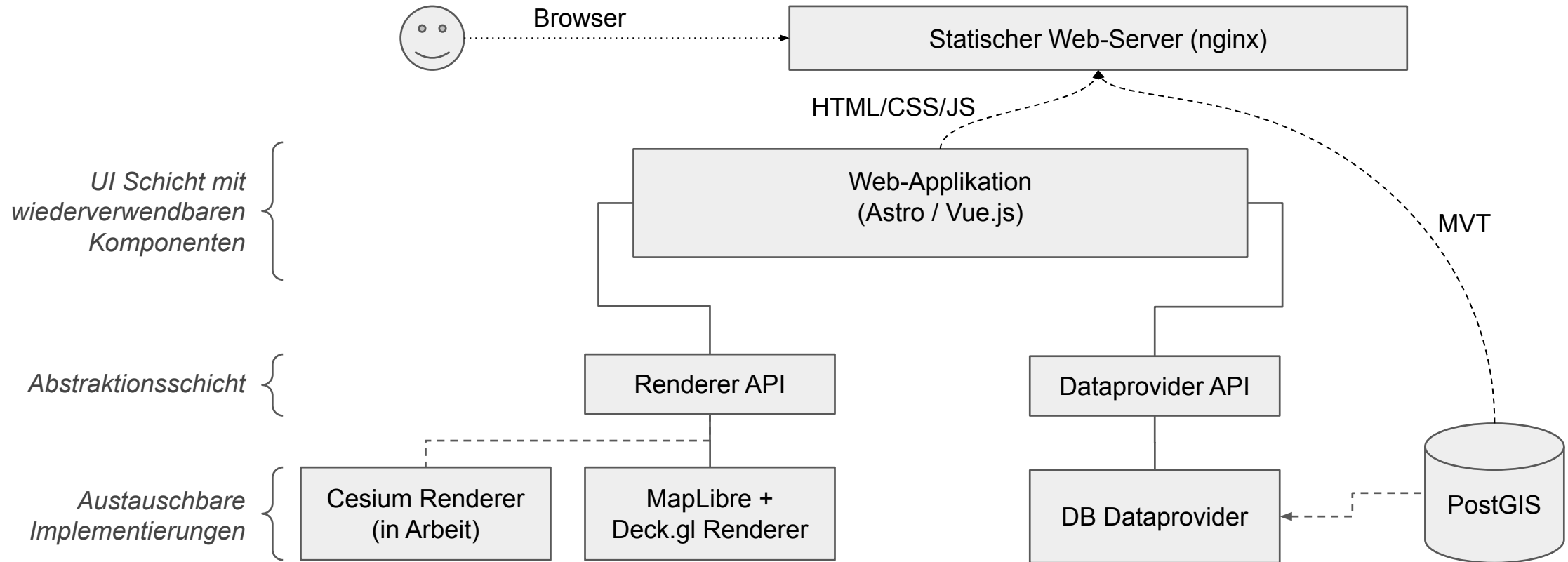
Konzept Finales Design & Designsystem



► Atomischer Aufbau des Designs erlaubt einen einfacheren Transfer in Code-Komponenten

Architektur

Eine möglichst schlanke, modulare Systemarchitektur



- Die Architektur trennt Benutzeroberfläche, Karten-Rendering und Datenzugriff klar voneinander und erlaubt eine schrittweise Weiterentwicklung ohne grundlegende Systemwechsel.

Statische Seiten + gezielte Interaktivität statt einer komplexen SPA



Idee

- Astro generiert statische HTML-Seiten
- Vue bringt Interaktivität dort, wo sie nötig ist (Karte, Suche, Filter, etc.)
- Einfache Auslieferung auf statischem Web-Server
- Gute Performance und schnelle Antwortzeiten
- Bessere Auffindbarkeit durch Suchmaschinen

Aber auch:

- Hydration-Albträume
- Lange Build-Prozesse
- Client-seitige Komplexität verschwindet nicht einfach

Kartenrendering

Für 2D und 3D brauchen wir unterschiedliche Karten-Renderer

MapLibre GL JS

- Fork von **Mapbox**
- Verbreiteter Renderer für **Basiskarten** (MVT / Mapbox/libre Style Spec)
- Unterstützt **Web Mercator** und seit einigen Monaten auch einen virtuellen **Globus**
- BSD 3-Clause-Lizenz

Deck.gl

- ehem. **Uber**, jetzt Teil der **OpenJS** Foundation
- WebGL-basierte **Visualisierungen** von **grossen Datensets**
- Gut **erweiterbar**
- Lässt sich mit MapLibre **“verschachteln”**
- MIT-Lizenz

Cesium JS

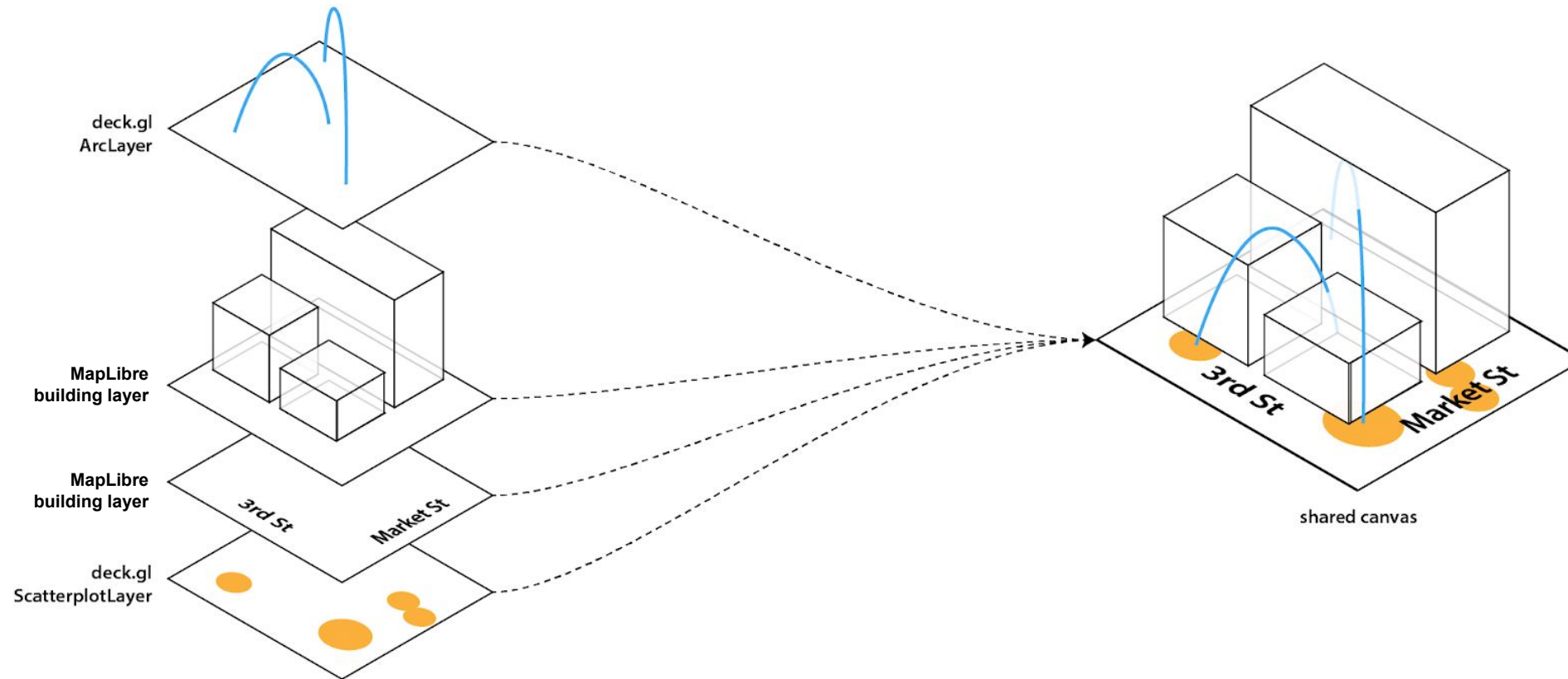
- Teil der Cesium Platform von **Bentley** Systems
- **Virtueller Globus** auf Basis von WebGL
- Stark für **3D-Rendering, Terrain** etc.
- Kein Support für 2D Vector Tiles; Fokus auf **3D Tiles** (OGC)
- Apache-2.0-Lizenz

- Kombination von MapLibre + Deck.gl für 2D / 2.5D;
Cesium (mit Anpassungen und Erweiterungen) für 3D und den Globus

Kartenrendering

MapLibre und Deck.gl liefern eine gemeinsame, integrierte Karte

Thematische Deck.gl-Layer lassen sich gezielt zwischen Layer der Vektor-Basiskarte platzieren. So entsteht ein gemeinsames Kartenbild statt zweier übereinander gelegter Systeme - in einem Canvas.



Quelle: <https://deck.gl/docs/get-started/using-with-map#interleaved>

Migration

Kartenstile aus dem Desktop-Atlas werden punktuell überarbeitet

EXEMPLARISCH

Styling function

```
1 var fill = attribute('1')
2   .unique_values('3 2 1 4 5 6 7 10 14 11')
3   .fill('#ffe5d7 #faffa5 #fcffd2 #e6e0c2
4     #edffeb #dcffd7 #e7eff0 #daff82 #b4d26b
5     #bfd79c')
6
7 function getStyle() {
8   return style(
9     fill, {
10       "stroke": "#414141",
11       "stroke-width": "0.015m",
12       "stroke-opacity": 0.4,
13       "altitude-clamping": "terrain",
14       "altitude-technique": "drape",
15       "render-order": 144
16     });
17 }
```

Save Copy Format Clear

JSON object

```
1 {
2   "attributes": {
3     "someAttribute": 10
4   }
5 }
```

Save Copy Format Clear

Evaluation result

- Eine eigene Styling-Library (D3.js-basiert) erlaubt das Beschreiben von Kartenstilen mittels JavaScript-Funktionen. Diese werden zur Laufzeit im Browser evaluiert.

Zwischenstand

170 der 400 Kartenthemen sind in der Beta-Version verfügbar

Demo

Nächste Schritte

Der Beta-Release ist der erste Schritt auf einer langen Reise

- Herbst 2026: Offizieller **Go-Live**
- Mehr Karten: **Migration** inkl. ausgewählter Rasterkarten
- Mehr **Interaktion**: Highlighting, erweiterte Feature-Abfragen, Messen und Zeichnen
- Mehr **Analyse**: Karten- und Objektvergleiche, interaktive Diagramme
- Bessere **Suche**: Feature-Suche und Conversational Search
- Mehr **Output**: Exportfunktionen, Einbetten
- Erweiterte **Darstellung**: 3D-Support / Globus für ausgewählte Karten

Abschluss

Learnings aus dem Projekt

- Klares **Scoping** und harte **MVP-Priorisierung** waren entscheidend, um eine erste Version **termingerecht** ausliefern zu können.
 - **Static Site Generation** (SSG) kann auch für interaktive Geo-Anwendungen ein tragfähiger Ansatz sein.
 - Die **inhaltliche Arbeit** darf nicht unterschätzt werden: 400 thematische Karten zu prüfen, anzupassen und ins Web zu überführen, ist ein grosser Teil des Projekts.
- **Vom Desktop ins Web heisst nicht kopieren; sondern neu denken, gezielt priorisieren, übersetzen und vereinfachen.**

Kontakt

Alexander Müdespacher

Fullstack Web Developer
amuedespacher@ethz.ch

Dr. Andreas Neumann

Projektleiter
aneumann@ethz.ch

ETH Zürich

Atlas der Schweiz
HIL G 12.1 / 13.1
Stefano-Franscini-Platz 5
8093 Zürich

www.atlasderschweiz.ch

