



netidee

PROJEKTE

weBIGeo

Endbericht | Call 18 | Projekt ID 6745

Lizenz: CC BY

Inhalt

1	Einleitung	3
2	Projektbeschreibung.....	3
3	Verlauf der Arbeitspakete	6
	Arbeitspaket 1 - <i>Detailplanung und Formales am Projektstart</i>	6
	Arbeitspaket 2 – <i>WebGPU Engine</i>	6
	Arbeitspaket 3 – <i>Datenintegration und -aufbereitung</i>	7
	Arbeitspaket 4 – <i>Visualisierung</i>	8
	Arbeitspaket 5 – <i>Schnittstelle</i>	8
	Arbeitspaket 6 – <i>Dokumentation und Formales am Projektende</i>	9
4	Liste Projektergebnisse	9
5	Verwertung der Projektergebnisse in der Praxis.....	10
6	Öffentlichkeitsarbeit/ Vernetzung	10
7	Eigene Projektwebsite	11
8	Geplante Aktivitäten nach netidee-Projektende	11
9	Anregungen für Weiterentwicklungen durch Dritte	12

1 Einleitung

Das Ziel des Projektes weBIGeo war große geographische Daten in Echtzeit direkt im Browser auszuwerten und direkt darzustellen. Im Zuge des Projektes wurde die erste WebGPU-basierte Engine zur Darstellung von interaktiven 3D Geovisualisierungen im Netz erstellt (AP 2). Wir konnten anhand von zwei komplexen Simulationsanwendungen den Proof-of-Concept erbringen, dass es möglich ist, mit Hilfe von WebGPU Compute Shaders große geographische Daten in Echtzeit mit hoher Qualität auszuwerten und direkt auf einer 3D Karte darzustellen (AP 3 und AP 4). Mit Hilfe einer GUI können Nutzer_innen die Simulationsparameter in Echtzeit anpassen und so flexibel verschiedene Szenarios durchspielen (AP 5).

2 Projektbeschreibung

WeBIGeo ist ein Framework mit dessen Hilfe große geographische Daten in Echtzeit direkt im Browser analysiert und visualisiert werden können. Dies ermöglicht datenwissenschaftliche Analysen und interaktive Explorationen. Im Allgemeinen kann das Framework für sämtliche Daten mit geographischem Zusammenhang verwendet werden. Die Visualisierung als Overlays auf unseren hochauflösenden 3D-Karten ist durch das Augenmerk auf eine modulare Programmierung unserer Graphik-Pipeline unkompliziert möglich.

Für den Proof-of-Concept unseres Frameworks haben wir uns entschieden die Thematik der Schneevisualisierung und der Lawinen-Simulation aufzugreifen. Zweiteres kann geopolitische Entscheidungsprozesse wesentlich vereinfachen.

Das Projektergebnis bedient folgende Zielgruppen:

- **Schnee- und Lawinenforschung:** Unser neuartiger Ansatz Lawinenabgänge mithilfe physikalischer Grundgesetze in Echtzeit zu simulieren, ermöglicht schnelle Gefahrenbeurteilung in gefährdeten Gebieten. Verglichen mit State-of-the-Art Systemen wie [AvaFrame](#) und [Flow-R](#) ist, neben der Berechnung in Echtzeit, vor allem die direkte Koppelung von Simulation und Visualisierung in der gleichen Arbeitsumgebung eine klares Alleinstellungsmerkmal. Damit können Simulationsforscher_innen den Einfluss von Parameter von Lawinenmodellen erstmals in Echtzeit untersuchen (Abbildung 1 - Abbildung 4).
- **Datenwissenschaftler_innen:** Der modulare Ansatz unserer Compute Pipeline, wie in unserer [technischen Dokumentation](#) beschrieben, ermöglicht die Erweiterung von weBIGeo als Simulationstool für weitere Problemstellungen mit geographischem Kontext. Beispiele dafür wären:
 - Schnee- und Windsimulationen

- Simulation anderer Naturkatastrophen (Hochwasser, Murenabgänge, Überflutungen)
- Berechnung des Solarpotentials sowie Verschattung
- **GIS Community:** Die modulare Umsetzung von weBIGeo bietet grundsätzlich die Möglichkeit rasterisierte Overlays auf Basis verschiedener Daten anzuzeigen. Das ermöglicht die interaktive Exploration beliebiger Datensätze.
- **Schneesport-Community:** Benutzerdefinierte Routen können über das Web-Interface hochgeladen und angezeigt werden. Durch die Lawinensimulation können gefährliche Stellen entlang der Route aufgezeigt werden (siehe Abbildung 5). Wir sehen hier vor allem Potential in der Freeride- und Skitourenszenen.

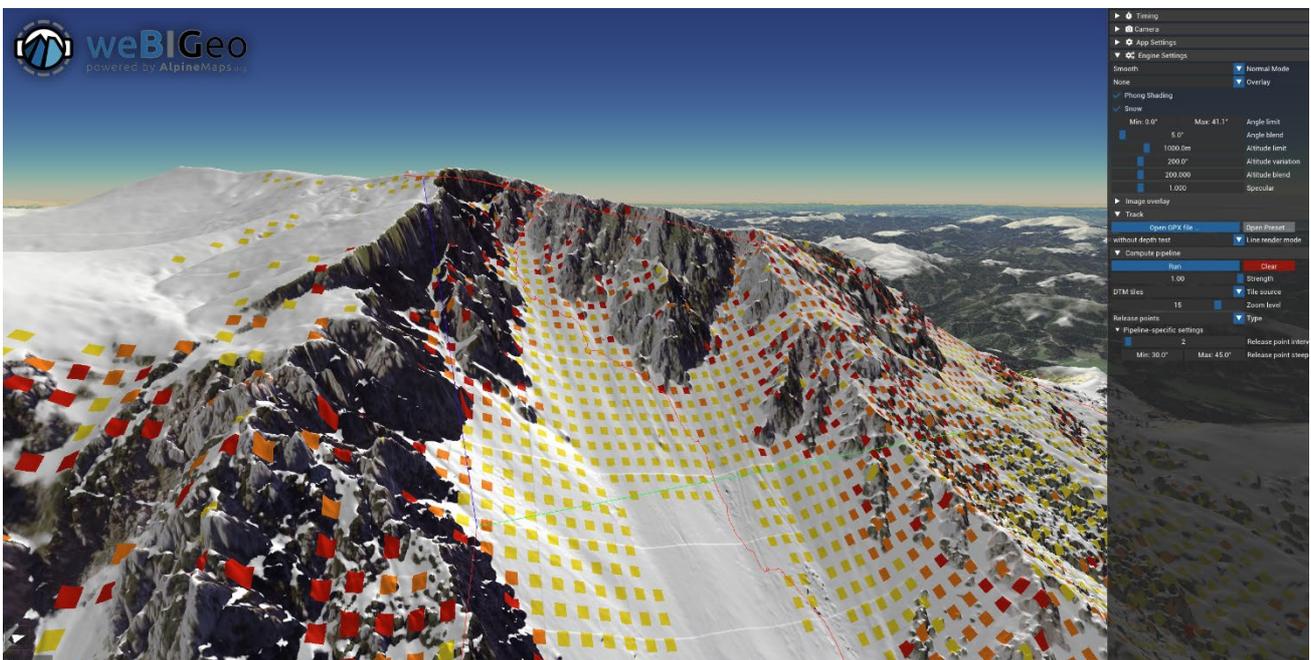


Abbildung 1: Mögliche Lawinenauslösepunkte in der Umgebung eines GPX-Tracks (rote Linie). Farbe codiert die Steilheit am Auslösepunkt. Mit weBIGeo können alle möglichen Lawinerverläufe in Echtzeit parallel ausgewertet und angezeigt werden [Breite Ries, Schneeberg, Niederösterreich].

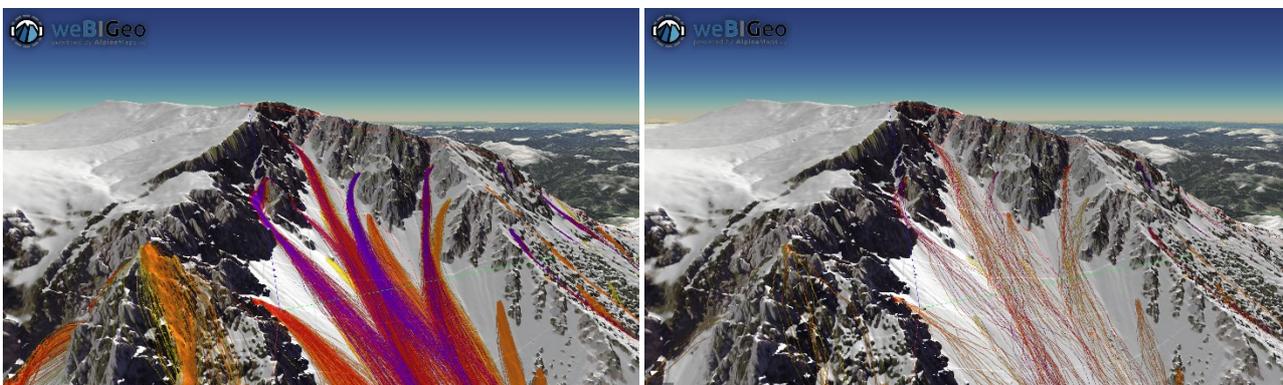


Abbildung 2: Verschiedene Parametrisierungen des Multi-Flow Modelles mit einer reduzierten Anzahl an Auslösepunkten: Links viele parallele Auslösungen und ein geringer Anteil zufälliger Richtungsänderungen und rechts wenige parallele Auslösepunkte aber mehr zufällige Richtungsänderungen.



Abbildung 3: Verschiedene Parametrisierungen eines Single-Flow Modelles ohne zufällige Richtungsänderungen: Links Berechnung auf einem groben Höhenmodellraster (ca. 5x5 Meter) und rechts auf einem genaueren Raster.

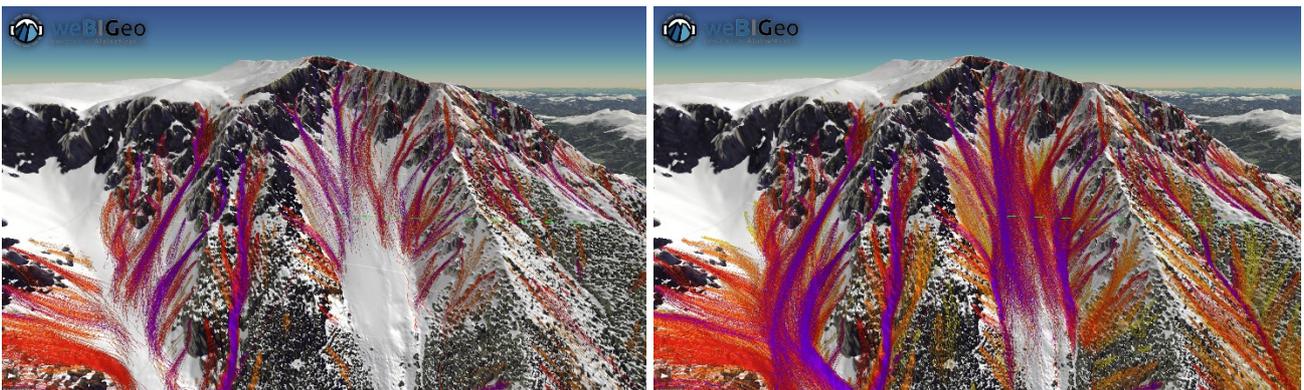


Abbildung 4: Verschiedene Parametrisierungen des einfachen Runout-Distance Modelles, das die Auslauflänge der Lawinen in Abhängigkeit des Geländewinkels kontrolliert (ein Faktor der Mächtigkeit der Lawinen): Links ein Winkel von 25° und rechts 21.5°.

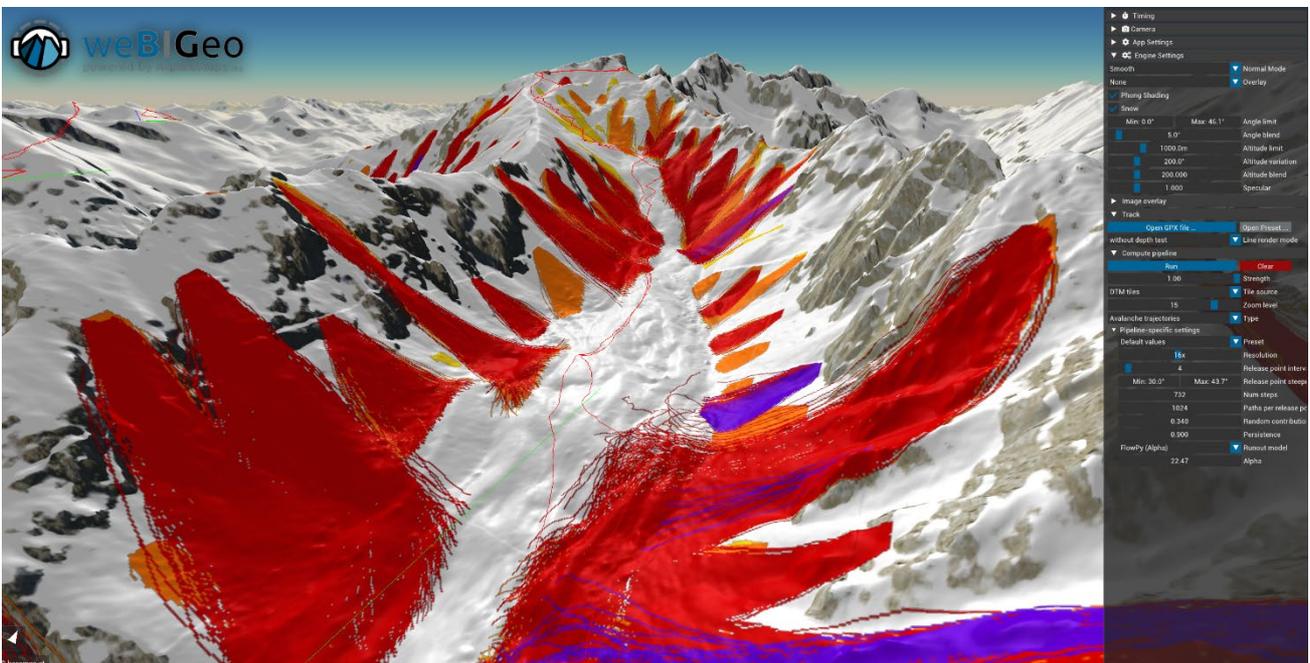


Abbildung 5: weBIGeo simuliert Lawinen in der Umgebung eines gegebenen GPX-Tracks. Für die Schneesport-Community hat diese Information das Potential, potentielle Gefahrenstellen entlang einer Route identifizieren zu können [Zechnerkar Spitze, Salzburg].

3 Verlauf der Arbeitspakete

Alle Arbeitspakete konnten planmäßig gestartet und abgeschlossen werden.

Arbeitspaket 1 - *Detailplanung und Formales am Projektstart*

Die **Haupttätigkeiten** zu Projektstart waren:

- Unterzeichnung des Vertrages
- Erstellung des Arbeitsplanes und Projektcontrolling
- Einreichung und Bewilligung der ersten Förderrate
- Aufsetzen und Einrichten von allen nötigen Ressourcen
- Einschulung der Mitarbeiter

Wir hatten einen erfolgreichen Projektstart und konnten, wie geplant, im Februar 2024 starten. Folgende Online Ressourcen wurden als **Ergebnisse** erstellt:

- <https://github.com/weBGeo> (Code in GitHub Repository)
- <https://www.cg.tuwien.ac.at/research/projects/weBGeo> (interne Projektwebseite)
- <https://www.netidee.at/webigeo/webigeo-kick> (Blogpost zum Projektstart)

Arbeitspaket 2 – *WebGPU Engine*

Folgende **Haupttätigkeiten** wurden in AP 2 durchgeführt:

- Ausführliche Plattform Tests zu Rendering und GUI mit verschiedenen Konfigurationen
- Implementierung einer GUI
- Implementierung einer 3D Basiskarte mit WebGPU Rendering
- Testen und Debugging

Folgende **Ergebnisse** wurden erzielt:

- Die Erkenntnisse und wesentlichen Ergebnisse unser Plattform Tests wurden in folgendem Blogpost zusammengefasst: <https://www.netidee.at/webigeo/der-tech-stack-fuer-webigeo>
- Einbindung aller nötigen Daten zur Erstellung der Basiskarte
- Darstellung der Karte mit GUI
- Unseres Wissens nach die erste WebGPU-basierte Rendering Engine für 3D Geländemodelle: <https://github.com/weBGeo/webigeo/tree/develop>
- Zusammenfassung der Stärken und Schwächen einer WebGPU-basierten Lösung im Vergleich zu WebGL in einem Blogpost: <https://www.netidee.at/webigeo/der-erste-webgpu-basierte-3d-terrain-renderer>
- Demoanwendung: <https://webigeo.alpinemaps.org/> (Achtung: läuft durch die noch eingeschränkte Verfügbarkeit von WebGPU nur auf einer aktuellen Version von Chrome und nicht auf mobilen Endgeräten!)

Wie bereits bei der Antragsstellung erwartet, ergaben sich **Probleme** durch die neue und dadurch noch unausgereifte WebGPU Grafikschnittstelle, insbesondere im Zusammenspiel mit Dawn (Google's WebGPU Implementierung) und Emscripten (der Compiler für WebAssembly, den wir für unseren Tech Stack verwenden). Technische Details zu den beiden Problemen (sowie deren Lösungen) haben wir in einem Blogpost dokumentiert: <https://www.netidee.at/webigeo/overcoming-limits-using-webgpu>

Als besondere **Erfolge** sind anzuführen, dass es sich sowohl bei Dawn als auch bei Emscripten um Open Source Projekte handelt und die kritischen Bugs so von uns gefunden und gemeldet werden konnten. Beide Probleme wurden von den entsprechenden Entwicklungsteams prompt aufgegriffen und gefixt. weBIGeo konnte daher bereits einen wichtigen Beitrag für die Open Source Community leisten.

Eine **Abweichung** vom ursprünglichen Plan war uns direkt auf 3D Karten zu fokussieren und keine 2D Version zu implementieren. Mit entsprechenden klassischen Overlays (anstelle der derzeit verwendeten Luftbilder), die auch als Open Data zur Verfügung stehen¹, können wir konzeptionell sehr einfach auch 2D Karten erzeugen.

Arbeitspaket 3 – *Datenintegration und -aufbereitung*

Die **Haupttätigkeiten** in AP 3 umfassten:

- Dokumentation von möglichen Szenarios für eine Proof-of-Concept Anwendung und Recherche über dafür nötige Daten
- Laden der benötigten Daten auf die GPU
- Compute Shader Implementierung von einfachen Szenarios mit steigender Komplexität
- Recherche über Lawinensimulationen und Austausch mit Simulationsexperten
- Umsetzung von Debugging Tools um die Ergebnisse und Performance der GPU-seitigen Datenanalyse zu inspizieren

Folgende **Ergebnisse** wurden erzielt:

- Eine Beschreibung von möglichen Anwendungsszenarios und dazu passenden Datenquellen, sowie eine genaue Beschreibung und Begründung für unser gewähltes Demo-Szenario, wurde in folgendem Blogpost zusammengefasst: <https://www.netidee.at/webigeo/warum-brauchen-wir-die-gpu-fuer-geographische-datenvisualisierung-teil-2>.
- Für eine benutzerdefinierte Region können die entsprechenden Daten von einem externen Server auf die lokale GPU geladen werden.
- Proof-of-Concept zur Echtzeit-Analyse basierend auf Höhendaten in Compute Shader anhand von folgenden Szenarios:
 - Berechnung von Normalvektoren
 - Simulation von Schneeoberflächen
 - Lawinensimulation: Gravitational Mass Flow Models wurden ausführlich recherchiert, in WebGPU nachimplementiert und für die parallele Berechnung mit Compute Shader optimiert

¹ <https://basemap.at/en/>

- Zum Inspizieren und Debugging der Berechnungsergebnisse werden die berechneten Daten im geographischen Kontext angezeigt. Es wurden zusätzlich GPU und CPU Performance-Timer implementiert und in die GUI integriert.
- Um Lawinensimulationen von Bäumen und Häusern unabhängig zu machen wurden innerhalb des Projektrahmens Höhenkacheln der Erdoberfläche für Österreich, abzüglich solcher Elemente berechnet und auf unserem Tile-Server zur Verfügung gestellt.

Es sind zwei besondere **Erfolge** hervorzuheben:

Erstens konnten wir zeigen, dass selbst Szenarios, die auch mit bereits bestehenden online Grafik APIs umgesetzt werden konnten, von einem Compute Shader Ansatz profitieren können. Unsere Ergebnisse zeigen bessere grafische Qualität durch höhere Details und weniger sogenannte Popping-Artefakte. Die technische Begründung und visuelle Beispiele finden sich in unserem Blogpost: <https://www.netidee.at/webigeo/let-it-snow>

Zweitens konnten wir erfreulicherweise – und **abweichend** vom ursprünglichen Projektplan - im Endeffekt ein Ziel umsetzen, das wir in unserem April-Blogpost noch als ferne Zukunftsvision formuliert hatten (<https://www.netidee.at/webigeo/warum-brauchen-wir-die-gpu-fuer-geographische-datenvisualisierung-teil-2>): Mit Unterstützung durch das Team von [avaFrame](#) für die theoretischen Aspekte konnten wir weBIGeo nutzen um großräumige Lawinensimulationen erstmals in Echtzeit online durchzuführen!

Arbeitspaket 4 – *Visualisierung*

Die **Haupttätigkeiten** in AP 4 umfassten:

- Umsetzung einer Overlayfunktion, um die analysierten Daten in ihrem geographischen Kontext anzuzeigen
- Erweiterung der GUI um Simulationsparameter zur Laufzeit anpassen zu können
- Testen und Debugging der Proof-of-Concept Szenarios mit Hilfe der erzeugten Visualisierungen

Folgende **Ergebnisse** wurden erzielt:

- Die Ergebnisse der Datenanalyse werden als Kartenoverlay gerendert. Die Auflösung der Simulation, entsprechend auch des generierten Overlays, können interaktiv angepasst werden. Selbst auf maximaler Auflösung benötigt die Simulation (abhängig vom Endgerät) nur wenige Millisekunden.
- Alle Simulationsparameter können in der GUI interaktiv verändert werden. Die Simulation wird bei jeder Änderung neu ausgeführt. Die Beschreibung der Simulationsparameter kann unserer [Dokumentation](#) entnommen werden.
- Blogpost über das Simulationsmodell, das die Grundlage unserer Lawinensimulation darstellt: <https://www.netidee.at/webigeo/lawinensimulation-echtzeit>

Arbeitspaket 5 – *Schnittstelle*

Die **Haupttätigkeiten** in AP 5 umfassten:

- Erweiterung der GUI um benötigte Daten auszuwählen und auf die GPU laden zu können
- Umsetzung des Ladens und Parsens von Dateien im GPX-Format

Folgende **Ergebnisse** wurden erzielt:

- Nutzer_innen können die Simulation auf eine durch GPX-Tracks definierte Region beschränken, welche über die GUI ausgewählt werden.
- Wir stellen eine [kleine Sammlung von GPX-Tracks](#) zur Verfügung, die zum Testen unserer Demoanwendung heruntergeladen werden können. (Hinweis: es können selbstverständlich auch eigene GPX-Tracks geladen werden!)
- In der [technischen Dokumentation](#) ist ersichtlich, wie unsere Implementierung funktioniert und erweitert werden kann.
- Die Verwendung der GUI und das unterstützte Datenformat ist in der [Nutzer innendokumentation](#) näher beschrieben.
- Blogpost über die interaktive Parametrisierung der Lawinensimulation und Ausblick: <https://www.netidee.at/webigeo/das-webigeo-finale>

Arbeitspaket 6 – Dokumentation und Formales am Projektende

Die **Haupttätigkeiten** in AP 6 umfassten:

- Aufbereitung und Bereitstellung der Dokumentation
- Erstellung des Endberichtes, der Zusammenfassung und der Endabrechnung
- Aktualisierung der NetIdee Webseite
- Finale Aktualisierung der Demo Webseite: <https://webigeo.alpinemaps.org/>

Das Projekt wurde erfolgreich im Februar 2025 abgeschlossen. Folgende **Ergebnisse** wurden im Zuge des finalen APs erzielt:

- Projektendbericht
- Zusammenfassung
- Endabrechnung und Projektcontrolling
- Aufstellung aller Projektergebnisse auf der NetIdee Webseite

4 Liste Projektergebnisse

1	Projektzwischenbericht	CC BY 4.0	https://www.netidee.at/webigeo
2	Projektendbericht	CC BY 4.0	https://www.netidee.at/webigeo
3	Entwickler_innen-DOKUMENTATION [EN] mit Informationen zu: <ul style="list-style-type: none"> • Development-Setup • Projekt-Struktur • Rendering- und Computepipeline 	CC BY 4.0	https://github.com/weBIGeo/webigeo/blob/release/netidee6745/docs/Setup.md https://github.com/weBIGeo/webigeo/blob/release/netidee6745/docs/Technical.md

4	Anwender_innen-DOKUMENTATION [EN] mit Informationen zu: <ul style="list-style-type: none"> • GPX-Track laden • Lawinensimulation • Schnee-Effekt 	CC BY 4.0	https://github.com/weBIGeo/webigeo/blob/release/netidee6745/docs/Usage.md
5	Zusammenfassung	CC BY 4.0	https://www.netidee.at/webigeo
6	Dokumentation Externkommunikation	CC BY 4.0	https://www.netidee.at/webigeo Endbericht, Kapitel 6
7	SW-Projektergebnis Teil 1: Source-Code	GPL 3.0	https://github.com/weBIGeo/webigeo/tree/release/netidee6745
8	SW-Projektergebnis Teil 2: Demo Webseite	GPL 3.0	https://webigeo.alpinemaps.org/

5 Verwertung der Projektergebnisse in der Praxis

Mit unserem Projekt konnten wir zeigen, dass selbst im Web komplexe Simulationen von Lawinenergebnissen in Echtzeit möglich sind. Ohne die Notwendigkeit, sich in komplexe Anwendungen einzuarbeiten oder ein aufwendiges Setup zu konfigurieren, können sämtliche Stakeholder potenzielle Geländerrisiken entlang einer Route erkennen und potentiell besser verstehen.

Wir betrachten die Lawinen- und Schneesimulation in weBIGeo als ein Demoprojekt – eine Grundlage für weitere Anwendungen und Forschungen. Es soll als Sprungbrett für zahlreiche interdisziplinäre Aufgaben dienen. In erster Instanz wird es in der Lawinenrisikobewertung eingesetzt, um die Kommunikation mit Stakeholdern zu verbessern. Die Echtzeitfähigkeit der Technologie ermöglicht zudem Sensitivitätsanalysen der zugrunde liegenden Modelle. In diesem Zusammenhang findet bereits ein fachlicher Austausch mit den Expertinnen und Experten des Schnee- und Lawinenteams am Bundesforschungszentrum für Wald (BFW) statt.

6 Öffentlichkeitsarbeit/ Vernetzung

In unseren [zehn Blogposts](#) erklären wir einerseits die technischen Grundlagen unseres Projektes und behandeln andererseits relevante fachliche Aspekte zu den Themen Visualisierung und Simulation. Besonders erfreut hat uns, dass dadurch mehrere Studierende – sowohl von der TU Wien als auch von externen Universitäten – auf unser Projekt aufmerksam wurden und uns ihr Interesse an einer Projektarbeit in diesem Themengebiet mitteilten.

Durch unsere vorhergehende Arbeit im Bereich der Lawinenrisikovisualisierung konnten wir bereits einige Kontakte zu Expert_innen aus dem Bereich der Lawinenforschung, Statistik und Kartographie herstellen. Mit dem Team, das hinter dem State-of-the-art Lawinensimulationstool [AvaFrame](#) steht, wurde gemeinsam ein Kurzartikel über unsere Echtzeitlawinensimulation eingereicht, der derzeit in Begutachtung ist.

Während der Projektlaufzeit hatten wir auch bereits die Gelegenheit Zwischenergebnisse unserer Arbeit anderen Forscher_innen zu demonstrieren – sowohl TU-intern als auch externen Wissenschaftler_innen, etwa vom deutschen Klimarechenzentrum und der Sorbonne Universität. Ein konkreter geplanter nächster Schritt ist, im Rahmen einer Forschungsarbeit zu untersuchen, wie Lawinenrisiko optimal visuell kommuniziert werden kann. Wir planen, die Ergebnisse bei einer wissenschaftlichen Konferenz im Bereich der Visualisierung einzureichen, wie beispielsweise der IEEE VIS.

7 Eigene Projektwebsite

Unsere weBIGeo Lawinensimulationsanwendung ist unter <https://webigeo.alpinemaps.org/> frei online verfügbar. Da WebGPU noch eine experimentelle API ist, wird sie noch nicht von allen Browsern unterstützt. Wir empfehlen Chrome auf einem Desktop PC. Es ist weiters davon auszugehen, dass sich die API auch in Zukunft noch ändern wird.

An unserem Institut erhält zusätzlich jedes drittmittelfinanzierte Projekt automatisch eine Webseite, auf der alle Publikationen und News geteilt werden. Hier finden sich im Moment alle für das Projekt relevanten Links: <https://www.cg.tuwien.ac.at/research/projects/weBIGeo>

8 Geplante Aktivitäten nach netidee-Projektende

Wir planen, das weBIGeo-Framework als Basis für weitere Projekte zu nutzen, wie beispielsweise Abschluss- und Projektarbeiten von Studierenden oder Forschungsarbeiten in den Bereichen der Visualisierung und Simulation. Folgende Folgeaktivitäten wurden bereits gestartet:

1. Eine systematische Evaluierung unserer Lawinensimulation im Vergleich zu State-of-the-Art Methoden, beispielsweise rechenintensiven physikalischen Modellen.
2. Eine Diplomarbeit zur detaillierteren Berechnung der Schneelage basierend auf frei verfügbaren Wettermodellen, sowie der realistischen Darstellung von Schneeoberflächen in Echtzeit.
3. Technische Unterstützung einer externen Diplomarbeit zur genaueren Simulation und Visualisierung von Lawinenauslösepunkten in Echtzeit auf Basis von weBIGeo.

4. Eine Diplomarbeit, die die optimale visuelle Darstellung von Lawinengefahr näher untersucht um die Gefahrenkommunikation mit verschiedenen Zielgruppen, etwa Bewohner_innen von gefährdeten Gebieten oder Wintersportler_innen, zu verbessern.

9 Anregungen für Weiterentwicklungen durch Dritte

weBIGeo bietet eine neuartige Plattform zum interaktiven Experimentieren mit Wettermodellen, sowie Modellierung von Naturkatastrophen. Alleine im Bereich der Lawinensimulation gibt es viele Weiterentwicklungsmöglichkeiten:

- Die Erweiterung unseres Lawinenmodelles, etwa durch die Integration alternativer Runout-Distance Modelle, bietet sich an. Durch die Echtzeitberechnung können verschiedene Modelle und Simulationsparameter effizient miteinander verglichen werden.
- Das interaktive Festlegen von Lawinen-Startpunkten („Klick-Lawinen“) in Kombination mit einem animierten Lawinenabgang ist ein weiteres Feature, welches interaktiv das Ausmaß von Lawinenkatastrophen vermitteln kann. Aufbauend darauf wäre auch das interaktive Platzieren von Lawinenschutzverbauungen denkbar. So könnten verschiedene Szenarien durchgespielt werden, um Bewusstsein für die Wirksamkeit und Wichtigkeit von Schutzmaßnahmen zu schaffen. Diese Funktionalität wären insbesondere für Bewohner_innen gefährdeter Gebiete sowie lokale Entscheidungsträger_innen relevant.
- Insbesondere für Tourengerher_innen würde die Einbindung des aktuellen, regionalen Lawinenwarnberichts in die Visualisierung einen Mehrwert bieten.
- Eine realistischere Darstellung der Schneeoberfläche würde helfen, unsere Simulationsergebnisse visuell ansprechender zu gestalten. Beispielsweise könnten mit oberflächlichen Effekten auch konkrete Lawinenprobleme visualisiert werden, wie beispielsweise Schneeverwehungen an Stellen, die anfällig für Tribschneelawinen sind.

weBIGeo ist aber auch abseits der Lawinensimulation eine interessante Basis für Weiterentwicklungen. Bereits in unserem [Blogpost zum Thema „GPU für geographische Datenvisualisierung“](#) bringen wir viele verschiedene Beispiele, die mit weBIGeo möglich wären.