

Fachhochschule München  
Fachbereich 08  
Geoinformationswesen

Wintersemester 2004/2005  
Studiengang: Kartographie  
und Geomedientechnik

# DIPLOMARBEIT

INTERAKTIVE WANDERKARTE DES YOSEMITE NATIONAL PARKS

von

Juliana Williams

Betreuer: Professor Dr. Reinhold Schumacher (FH München)  
Professor Dr. Lorenz Hurni (ETH Zürich)

Die Arbeit wurde angefertigt am Institut für Kartographie an der Eidgenössischen  
Technischen Hochschule in Zürich

## Zusammenfassung

Thema dieser Arbeit ist die Entwicklung einer interaktiven Online Wanderkarte des Yosemite National Parks. Die Applikation soll den Nutzer bei der Wahl eines Wanderweges unterstützen. Er kann die Wanderung an Hand von verschiedenen Kriterien (Name, Dauer, Schwierigkeit, Region) auswählen. Nachdem gewählt wurde, zeigt die Karte den Wanderweg an und gibt zusätzliche Informationen aus. Der Nutzer hat die Möglichkeit den Weg zu animieren und ein Höhenprofil berechnen zu lassen.

In einer Photogalerie kann der Nutzer aus verschiedenen Themen wählen. Die Aufnahmestandorte der Photos werden mit einem Symbol in der Karte dargestellt. Zu jedem Bild werden Metadaten mit angezeigt. Ebenfalls sind zwei Webcams der Yosemite Association integriert.

Der Webmapping Client ist vollständig in SVG und ECMAScript geschrieben und läuft sowohl in jedem Webbrowser, der vom Adobe SVG Viewer unterstützt wird, als auch im Apache Batik Viewer. Nachdem die Benutzeroberfläche geladen ist, wird eine Abfrage des Clients an die räumliche Datenbank und an einen OGC Webmapping Service gemacht, um die Kartendaten zu erhalten. Wegen der verschiedenen Detaillierungsgrade der Datensätze, kann in grösseren Datenmengen navigiert werden. Immer wenn der Nutzer navigiert, wird ihm ein massgeschneiderter Teil der Karte präsentiert. Thematische Daten für die Wanderwege und die Photogalerie werden im XML Format gespeichert und können direkt vom Client gelesen und dargestellt werden.

Daten für dieses Projekt stammen grösstenteils vom United States Geological Survey und vom National Park Service. Die Daten wurden innerhalb von ESRI ArcGis aufbereitet und vervollständigt, sowie in die endgültige Projektion umgerechnet (Datum: NAD83, Projektion: UTM, Ellipsoid: GRS80). Die Vektordaten wurden an eine räumliche Datenbank (PostGIS, opensource) übergeben, die u.a. räumliche Abfragen, das Ausschneiden und Vereinfachen von Daten und die Ausgabe in SVG unterstützt. Die Orthophotos werden direkt vom Terraserver Webmapping Service geladen. Zusätzliche Rasterdaten werden von einem internen WMS geladen. Die Trennung von Inhalt und Präsentation erlaubt eine rasche Adaption der Applikation an andere Regionen.

## Abstract

The topic of this project is an interactive online hiking map of Yosemite National Park. This application should help the user select a hike. He has the ability to choose a hike according to several criteria (name, duration, area). When a hike has been selected, the map shows the trail along with relevant information. The user can also animate the trail or calculate a profile.

The photo gallery allows the user to choose from several subjects. All photo positions are identified in the map with a symbol. Each photo has additional metadata. Two webcams from the Yosemite Association are integrated as well.

The webmapping client is entirely written in SVG and ECMA-Script and runs in every web browser and operating system supported by the Adobe SVG viewer as well as the stand-alone Apache Batik viewer. After loading the initial graphic user interface, the client makes requests to the spatial database and Webmapping Service (WMS) in order to receive the map data. The client allows navigation in larger datasets, as a result of a specified level of detail concept. Every time the user navigates, he is presented with a customized section of the map. Thematic data for hiking trails and georeferenced photos are delivered in XML-format which can be directly read and displayed by the webmapping client.

Data for this project was obtained from the USGS and from the National Park Service. The different data sources were compiled and completed within desktop GIS-Systems and reprojected to the final projection system (Datum:NAD83, Projection: UTM, Ellipsoid: GRS80). The vector data was transferred to an opensource spatial database (PostGIS) which supports spatial queries, intersections, simplifications and SVG output. Orthophotos are directly loaded from the Terraserver Webmapping Service. Additional raster data is also requested from an inhouse WMS. The separation of content and presentation allows a quick adaptation of the application to other regions.

## Dank

An dieser Stelle möchte ich mich bei all jenen Personen bedanken, die diese Arbeit ermöglicht und unterstützt haben.

Ein besonderer Dank geht an Prof. Dr. Reinhold Schumacher von der Fachhochschule München und Prof. Dr. Lorenz Hurni vom Institut für Kartographie an der ETH Zürich, die mir die Möglichkeit gaben, diese Diplomarbeit durchzuführen.

Dr. Hans-Ruedi Bär und Andreas Neumann stellten mir Programmcodes zur Verfügung, die ich in meinem Projekt integrieren durfte. Berhard Jenny war bei der Erklärung seines Programmes Shadow sehr hilfreich. Ausserdem danke ich Mark Grupé, Tom Patterson und Kent van Wagtendonk vom National Park Service und Tom Harrison für die freundliche Hilfe bei der Beschaffung von Daten.

Vielen Dank an meinem Freund Andreas Neumann, der mich als erster in die spannende Welt von SVG einführte, für seine Betreuung, seine unermüdliche Geduld in technischen Fragen und für seine tatkräftige Unterstützung.

Nicht zuletzt bedanke ich mich bei meiner Mutter Ulrike Williams, die mich während meines Studiums stets unterstützte.

## Inhaltsverzeichnis

Zusammenfassung.....	1
Abstract.....	2
Dank.....	3
Inhaltsverzeichnis.....	4
Abbildungsverzeichnis.....	6
1 Einleitung.....	8
1.1 Problemstellung.....	8
1.2 Aufbau.....	9
1.3 Innovation.....	10
2 Warum Webkarten?.....	11
2.1 Einflüsse des WWW auf Informationsbeschaffung von Kartennutzern.....	11
2.2 Vor- und Nachteile von Webkarten.....	11
2.3 Kartographische Visualisierung.....	12
2.4 Anforderungen an Bildschirmkarten.....	14
3 Möglichkeiten der Umsetzung.....	15
3.1 Client-seitige Techniken.....	15
3.2 Server-seitige Techniken.....	16
3.3 Interaktivität.....	17
4 Praktischer Teil: Umsetzung.....	18
4.1 Datengrundlage.....	18
4.1.1 Bearbeitungsgebiet.....	18
4.1.1.1 Allgemeines und Geschichte.....	18
4.1.1.2 Entstehung und Geologie.....	21
4.1.1.3 Vegetation.....	24
4.1.2 Datensätze.....	31
4.2 Gewählte Systemarchitektur, Technische Grundlagen.....	33
4.2.1 XML: Markupsprache zum Abspeichern heterogener, hierarchischer In- formationen.....	33
4.2.2 SVG: Textdateien zum Beschreiben von 2D Vektorgraphik.....	34
4.2.3 ECMAScript (JavaScript): Clientseitige Programmiersprache.....	37
4.2.4 PostgreSQL mit PostGIS spatial extension: Geographische Datenbank.....	38
4.2.4.1 Eigenschaften von PostgreSQL und PostGIS.....	38
4.2.4.2 Aufbau der Datenbank.....	39
4.2.4.3 Import geographischer Daten.....	40
4.2.5 PHP: Serverseitige Programmiersprache.....	41
4.2.6 Extrahieren/Generieren von SVG Daten.....	42
4.2.7 Higrd: ein Programm zur Extraktion von Geländemolldaten.....	44
4.2.8 Open GIS Consortium WMS (Web Map Service).....	45
4.3 Kartengestaltung.....	49
4.3.1 Masstabsabhängige Symbolisierung.....	49
4.3.2 Textplatzierung.....	51
4.3.3 Reliefschattierung.....	55
4.4 Graphische Benutzeroberfläche.....	57
4.5 Interaktivität.....	58
4.5.1 Räumliche Navigation.....	58
4.5.2 Räumlicher Bezug.....	60

---

4.5.3 Ebenenkontrolle.....	61
4.5.4 Attributdatenanzeige.....	63
4.5.5 Thematische Navigation.....	63
4.5.6 Wanderwege.....	63
4.5.6.1 Auswahl nach Benutzerdefinierten Kriterien.....	64
4.5.6.2 Anzeige und Animation des gewählten Wanderweges.....	64
4.5.6.3 Wanderweginformationen.....	67
4.5.6.4 Profilzeichnung.....	68
4.5.7 Photogalerie.....	71
4.5.8 Touristische Informationen.....	73
4.5.9 Externe Links.....	75
5 Schlussfolgerungen.....	76
5.1 Zusammenfassung und Beurteilung der Ergebnisse.....	76
5.2 Weiterführende Möglichkeiten der Arbeit.....	77
6 Bibliographie.....	79
7 Abkürzungsverzeichnis.....	82
8 Anlagen.....	83
9 Erklärung gemäss § 31 Abs. 5 RaPO.....	89

---

## Abbildungsverzeichnis

Illustration 1: "Map Use Cube" nach MacEachren und Taylor (1994).....	13
Illustration 2: Karte von Kalifornien mit Yosemite National Park nach USGS.....	18
Illustration 3: Klimadiagramm des Yosemite Valley.....	19
Illustration 4: Geologische Karte des Yosemite National Parks und Umgebung.....	22
Illustration 5: Klassifikation der plutonischen Gesteinsnamen.....	23
Illustration 6: Lodgepole Pine ( <i>Pinus contorta</i> ssp. <i>Murrayana</i> ).....	25
Illustration 7: Giant Sequoia ( <i>Sequoiadendron giganteum</i> ).....	25
Illustration 8: Ponderosa Pine ( <i>Pinus ponderosa</i> ).....	25
Illustration 9: Jeffrey Pine ( <i>Pinus jeffreyi</i> ).....	25
Illustration 10: Whitebark Pine ( <i>Pinus albicaulis</i> ).....	26
Illustration 11: Sugar Pine ( <i>Pinus lambertiana</i> ).....	26
Illustration 12: Red Fir ( <i>Abies magnifica</i> ).....	26
Illustration 13: Quaking Aspen ( <i>Populus tremuloides</i> ).....	26
Illustration 14: Live Oak ( <i>Quercus virginiana</i> ).....	27
Illustration 15: Black Oak ( <i>Quercus kelloggii</i> ).....	27
Illustration 16: Huckleberry Oak ( <i>Quercus vaccinifolia</i> ).....	27
Illustration 17: Greenleaf Manzanita ( <i>Arctostaphylos patula</i> ).....	27
Illustration 18: Gooseberries ( <i>Ribes uva-crispa</i> ).....	28
Illustration 19: Whiteleaf Manzanita ( <i>Arctostaphylos viscida</i> ).....	28
Illustration 20: Currents ( <i>Ribes rubrum</i> ).....	28
Illustration 21: Deer Brush ( <i>Ceanothus integrifolius</i> ).....	28
Illustration 22: Poison Oak ( <i>Toxicodendron diversilobum</i> ).....	29
Illustration 23: Mountain Misery ( <i>Kitkitdizze</i> ) ( <i>Chamaebatia foliolosa</i> ).....	29
Illustration 24: California Poppy ( <i>Eschscholzia californica</i> ).....	29
Illustration 25: Bigelow's Sneezeweed ( <i>Helenium bigelovii</i> ).....	29
Illustration 26: Jeffrey Shooting Star ( <i>Dodecatheon jeffreyi</i> ).....	30
Illustration 27: Indian Paintbrush ( <i>Castilleja minutata</i> ).....	30
Illustration 28: Lupine ( <i>Lupinus</i> ).....	30
Illustration 29: White Mountain Heather ( <i>Cassiope mertensiana</i> ).....	30
Illustration 30: Sierra Primrose ( <i>Primula suffrutescens</i> ).....	31
Illustration 31: Sky Pilot ( <i>Polemonium viscosum</i> ).....	31
Illustration 32: Anfrage und Extraktion von SVG-Daten.....	42
Illustration 33: Beispiel eines Terraserver Orthophotos.....	47
Illustration 34: Kartographische Textplatzierungsreihenfolge.....	53
Illustration 35: Nicht eindeutige Textplatzierung.....	55
Illustration 36: Digital erstelltes Relief des USGS vor der Bearbeitung.....	56
Illustration 37: Nach der Bearbeitung mit Shadow und Photoshop.....	56
Illustration 38: Schema des Layouts der Applikation.....	57
Illustration 39: Räumliche Navigation.....	59
Illustration 40: Massstabsleiste mit Kilometer- und Meilenangabe.....	60
Illustration 41: Massstabsleiste mit Meter- und Yardsangabe.....	60
Illustration 42: Ebenenkontrolle.....	61
Illustration 43: Beispiel einer Legende (Verkehr).....	61

---

Illustration 44: Legende für die generalisierte Geologiekarte.....	62
Illustration 45: Legende für die detaillierte Geologiekarte.....	62
Illustration 46: Beispiel eines Wanderweges mit Informationen.....	64
Illustration 47: Darstellung des Offsets für die Animation.....	66
Illustration 48: Informationen eines Wanderweges.....	67
Illustration 49: Beispiel eines Wanderweges mit Profil.....	68
Illustration 50: Detail Profil.....	69
Illustration 51: Schema der vertikalen Profilrechtecke und der Abschlusslinie.....	70
Illustration 52: Photogalerie.....	72
Illustration 53: Georeferenziertes Photo mit Angaben.....	73
Illustration 54: Legende der Hydrologie.....	87
Illustration 55: Legende der Vegetation.....	87
Illustration 56: Legende der Höhenkoten.....	87
Illustration 57: Legende des Verkehrssystems.....	87
Illustration 58: Legende der Geologie (generalisiert).....	87
Illustration 59: Legende der Geologie (detailliert).....	87
Illustration 60: Legende der Eigenschaften eines Wanderweges.....	88



# 1 Einleitung

## 1.1 Problemstellung

Häufig sucht man im Internet nach Informationen zu einer Wanderung oder möchte sich ein Bild der gewünschten Umgebung und Landschaft machen. Doch meist ist die Qualität der Karten und der Bilder im Internet nicht zufriedenstellend. Viele Karten sind schwierig zu bedienen, graphisch wenig attraktiv und beachten häufig die kartographischen Grundregeln nicht.

Ziel dieser Diplomarbeit ist es, eine qualitativ hochwertige Kartenapplikation für Wanderer zu entwickeln, die einen hohen Interaktivitätsgrad aufweist und einfach zu bedienen ist. Neben der detaillierten topographischen Karte und den Informationen zu den einzelnen Wanderrouten, sollen auch Photos zur Verfügung gestellt werden, die nach Möglichkeit auch georeferenziert sind. Notwendige Informationen zur Anfahrt, Übernachtungs- und Verpflegungsmöglichkeiten in der Gegend komplettieren das Informationsangebot. Als Extras sind weiterführende Links aufgelistet.

Yosemite National Park hat mich schon immer fasziniert und meine Aufenthalte dort waren meist zu kurz, um ausreichend in die Schönheit der Natur eintauchen zu können. Obwohl es sehr viele Informationen zum Park gibt, sowohl analog als auch digital, habe ich keine Seite im Internet gefunden, die sich der ausführlichen Darstellung von Wanderwegen widmet. Dies erstaunt um so mehr, da der Yosemite National Park der meist besuchte National Park in den Vereinigten Staaten ist, nur Disney Land hat mehr Besucher pro Jahr!

Viele Besucher kennen nur die kurzen und einfachen Wege, die im Tal zu finden sind. Viele der anstrengenderen Wege sind nur Insidern bekannt. Die in dieser Diplomarbeit erstellte interaktive Wanderkartenapplikation soll den Besuchern die Möglichkeit geben schon im Voraus auch die weniger bekannten Routen zu planen.

Bei meinen Recherchen im Internet konnte ich nur eine einzige umfassende Website mit detaillierten Informationen, inkl. Kartendarstellungen und Wanderwegprofilen für den Wanderer in US Nationalparks finden: eine neu gestaltete Seite vom Bryce Canyon National Park (<http://nps.gov/brca/home.htm>). Hier hat man die Möglichkeit eine Photogalerie mit 30 Bildern anzuschauen. Sie erscheinen in einer niedrigen Auflösung und können durch Anklicken vergrößert werden. Allerdings haben sie weder einen Titel noch eine Georeferenzierung. Dadurch ist es schwierig sich vorzustellen, wo die Bilder tatsächlich aufgenommen wurden. Der "Trail Advisor" ist besser gestaltet. Man hat die Möglichkeit einen Wanderweg nach der Schwierigkeit und/oder nach der Länge auszuwählen. Anschliessend werden die Wanderwege aufgelistet. Es folgen drei Graphiken: ein Ausschnitt aus der 25.000 topographischen Karte, ein Höhenprofil und eine 3-D Ansicht des Geländes. Es ist als positiv zu bewerten, dass es eine solche Seite gibt, die Wanderwege anschaulich darstellt und alle Angaben in Feet/Meter bzw. Meilen/Ki-

lometer angibt. Allerdings ist die Seite graphisch wenig ansprechend gestaltet. Die topographische Karte ist zu klein und der eingezeichnete Weg zu ungenau, um sich ein genaueres Bild darüber machen zu können, wie die Wanderroute verläuft. Das Höhenprofil ist sehr stark überhöht und es ist nicht ersichtlich wo das Profil in der Karte anfängt oder endet. Die 3-D Ansicht ist ebenso "lieblos" gestaltet und trägt nicht zum Informationsgewinn bei. Zusätzliche Informationen zu den Wanderwegen wie Start- und Endpunkt, Beschreibungen, Tipps oder zeitliche Länge des Wegs fehlen.

Die in meiner Applikation aufgeführten Wanderwege sind aus Büchern oder aus dem Internet recherchiert und sind keineswegs als vollständig zu betrachten. Dies ist nur ein Ausschnitt von vielen möglichen Routen. Jedoch ist das Wanderroutennetz vollständig vorhanden, und soweit wie möglich mit den entsprechenden Routennamen ergänzt worden. Detaillierte Angaben dazu sind allerdings nicht vorhanden.

## **1.2 Aufbau**

Diese schriftliche Ausarbeitung zur Applikation ist in sieben Teile gegliedert. Im ersten Teil wird auf Bildschirmkarten und deren Eigenschaften eingegangen. Hier werden die Einflüsse des Internet auf die Informationsbeschaffung von Kartennutzern und die Vor- und Nachteile der Bildschirmkarten erörtert. Ausserdem werden die kartographische Visualisierung und die Anforderungen an Bildschirmkarten abgehandelt.

Danach wird über Möglichkeiten der Umsetzung diskutiert. Hier werden die Client- und Server-seitigen Techniken erklärt. Wiederum auch mit ihren Vor- und Nachteilen. Ausserdem geht dieser Teil detailliert auf die Interaktivität ein.

Als nächstes kommt die Umsetzung der Arbeit. Zuerst wird das Bearbeitungsgebiet ausführlich vorgestellt. Hier wird die Geschichte, Geologie und Vegetation erläutert. Anschliessend wird auf die Datensätze Bezug genommen.

Danach werden die technischen Grundlagen in Form von Definitionen und Beispielen erklärt. Im nachfolgenden Text wird immer wieder auf diese Erklärungen zurückgegriffen.

Der Aufbau der verwendeten geographischen Datenbank PostgreSQL mit der PostGIS Spatial Extension wird nachfolgend erläutert. Ausserdem wird der Import von geographischen Daten und das Extrahieren bzw. Generieren von SVG-Daten erklärt.

Schliesslich wird die Kartengestaltung, die graphische Benutzeroberfläche und die Interaktivität der Karte ausführlich beschrieben.

Zum Schluss kommt es zu einer Zusammenfassung der Arbeit mit einer Beurteilung der Ergebnisse. Gedanken zu weiterführenden Möglichkeiten der Applikation runden die Arbeit ab.

Da die Applikation als Client-Server-Applikation konzipiert wurde, ist die beiliegende DVD keine sofort ab spielbare Multimedia-CD, sondern eine Kopie der Webserver- und Client Module, sowie der Daten. Die Geodaten wurden sowohl in den Rohformaten (ESRI Geodatabase, Geotiff Rasterdaten, Geländemodelle), sowie in PostgreSQL "Dumpfiles" (Wiederherstellung der Datenbank bei Systemabsturz) abgespeichert. Sämtliche zur Installation der Serverkomponenten notwendigen Scripte und Hinweise befinden sich ebenfalls auf der DVD.

Die Applikation ist jedoch online öffentlich unter der Adresse <http://www.carto.net/williams/yosemite/> verfügbar. Zum Betrachten der Website muss das Adobe SVG Plugin, erhältlich unter <http://www.adobe.com/svg/viewer/install/>, installiert werden.

### **1.3 Innovation**

Was macht diese Arbeit anders als bestehende Informationsseiten im Web? Die Karte, der zentrale Bestandteil der Applikation, ist von hoher Qualität. Die Wanderwegsektion, die viele Informationen zum Weg enthält, erlaubt eine Auswahl der Wege nach bestimmten Kriterien. Die gefundenen Resultate können auf Wunsch animiert dargestellt werden oder ein Höhenprofil des Weges erstellt werden. Die Karte selbst wurde mit zahlreichen Interaktionsmöglichkeiten ausgestattet. Der Nutzer kann in der Karte navigieren, Kartenebenen ein- und ausblenden, Informationen zu den geographischen Objekten anzeigen, sowie Bildmaterial abrufen. Das Layout ist zudem benutzerfreundlich und ansprechend gestaltet.

Die Daten wurden professionell in einem Geoinformationssystem (GIS) aufbereitet und sind georeferenziert abrufbar. Adaptive Massstabsleisten sowie eine verknüpfte Referenzkarte geben Auskunft über den gewählten Kartenausschnitt und dessen Massstab. Die Website gibt dem Nutzer die Möglichkeit mehrere verwandte Themen gleichzeitig anzusehen, z.B. Anfahrt zum Park, Übernachtungs- und Verpflegungsmöglichkeiten. Eine Photogalerie mit georeferenzierten Bildern gibt Einblicke in den Park und die gewählten Wanderrouten.

## 2 Warum Webkarten?

### 2.1 Einflüsse des WWW auf Informationsbeschaffung von Kartennutzern

In der heutigen mobilen und mediengerichteten Welt wird häufig lieber das Internet als Informationsquelle benutzt, als herkömmliche Medien wie Bücher oder Zeitschriften. Gründe dafür mögen die Bequemlichkeit der Nutzer, das schnelle Auffinden von Informationen, der günstige Preis des Mediums, sowie die potentiell hohe Aktualität der Informationen sein. Viele Kartennutzer wollen zuerst im Internet nach Karten und Informationen suchen und womöglich schon Entscheidungen treffen, bevor sie in den Buchladen gehen und eine Karte kaufen.

Eine Karte im Internet soll einen Überblick über das gesuchte Gebiet geben und schon bei der Planung helfen. Dies hat zur Folge, dass im Internet viele Karteninformationen bereitstehen, die jedoch leider nicht immer von bester Qualität sind. In der Praxis kommt es wohl häufig vor, dass sich Wanderer zunächst im Internet informieren und dann jedoch gezielt auf gedruckte Karten und Bücher zurückgreifen, da diese meist von höherer Qualität sind.

### 2.2 Vor- und Nachteile von Webkarten

Papierkarten haben sowohl Vor- als auch Nachteile. Sie können überall mitgenommen werden, sind also für unterwegs praktisch und man hat alle Informationen die man braucht, bei der Hand. Bildschirmkarten brauchen immer einen Rechner oder PDA, sowie die entsprechende Stromversorgung. Manche gedruckte Karten weisen eine zu hohe Informationsdichte auf und können leichter veralten, da sie aus Kostengründen nicht so häufig aktualisiert werden.

Bei Bildschirmkarten kann der Nutzer die für ihn individuell zusammengestellten Informationen anschauen. Die Karten können mit Hilfe eines Computers überall angeschaut werden und sind zudem sofort verfügbar, sofern ein Rechner zur Verfügung steht. Durch das Internet gibt es eine grosse Verteilung und Verbreitung von Bildschirmkarten. Ausserdem ist eine Darstellung von Text- und Graphikinformationen, Ton, Interaktivität und Animation möglich. Der Kartennutzer braucht zudem keine eigenen Daten oder eigene kartographische Software zu installieren. Doch es gibt auch Nachteile. Die Qualität von Bildschirmkarten ist meistens nicht so gut wie die von Papierkarten. Diese Karten sind nur auf dem Computer vorhanden, falls sie nicht ausdruckbar sind. Und auch dann handelt es sich meist um mindere Qualität. Während auf einer Papierkarte die Farben von Karte zu Karte konstant bleiben, sind die Farben von Bildschirm zu Bildschirm anders. Ausserdem werden häufig Plug-ins z.B. für SVG oder Flash, zusätzlich zur Webbrowser Software, zum Betrachten der Applikationen benötigt.

## 2.3 Kartographische Visualisierung

*“... a method of computing. It transforms the symbolic into the geometric, enabling researchers to observe their simulations and computations. Visualization offers a method for seeing the unseen... Visualization embraces both image understanding and image synthesis. That is, visualization is a tool both for interpreting image data fed into a computer, and for generating images from complex multidimensional data sets. It studies those mechanisms in humans and computers which allow them in concert to perceive, use and communicate visual information.” (McCormick et al, 1987)*

*“... a peculiar geographic skill demanding a synoptic view of the distribution (s) portrayed on the map(s)” (Board, 1978)*

*“... the use of concrete visual representations whether on paper or through computer displays or other media- to make spatial contexts and problems visible, so as to engage the most powerful human information-processing abilities, those associated with vision.” (MacEachren et al., 1992)*

Was ist kartographische Visualisierung? In der traditionellen Kartographie, beschäftigte man sich nur mit der Herstellung und Präsentation von statischen Karten und geographischen Daten. Heute können und werden die Daten mit Hilfe von Animation, Interaktivität und anderen multimedialen Möglichkeiten, wie z.B. Video, Animation, Bilder und Ton, präsentiert. Die zwei Komponenten der kartographischen Visualisierung sind Präsentation und Exploration.

Wenn die Visualisierung mit Geodaten in Verbindung gebracht wird, wird sie als eine Erweiterung der räumlichen Analyse betrachtet. Diese Analyse ist für die Erforschung der räumlichen Daten, die die Unterstützung der Visualisierung braucht, nützlich. Die Visualisierung dient aber auch zur Präsentation von Informationen und der Ergebnisse räumlicher Analysen.

Bertin<sup>1</sup> hat die erste systematische, detaillierte und umfassende Analyse der graphischen Elemente erstellt. Er definierte sieben visuelle Variablen: Position, Form, Orientierung, Farbe, Beschaffenheit, Wert (Helligkeit eines Zeichens) und Grösse. 1994 erweiterte MacEachren diese graphischen Variablen, indem er die Farbe in Helligkeit, Buntton und Sättigung unterteilte. Im Zuge der Einführung von Animationen kamen noch weitere Variablen hinzu, etwa Zeitpunkt, Dauer, Häufigkeit, Ordnung, Veränderungsrate und Synchronisation. Diese wurden u.a. von Dibiase und Dransch erweitert.

Die kartographische Visualisierung ist nicht synonym mit der kartographischen Präsentation. Hierbei beachte man die Unterschiede zwischen der animierten und der statischen Karte, die Interaktivität und Exploration gegenüber der einfachen Dar-

1 Jacques Bertin schrieb 1967 ein bahnbrechendes Buch zur Theorie der Visualisierung (“Sémiologie Graphique”)

stellung der Ergebnisse und die mehrfache Sichtweise versus einer einzigen, optimalen Karte.

Bei der Interaktivität gibt es Vorgänge, die während der Exploration (“Entdeckung”) das visuelle Denken stimulieren: Blinken, Hervorheben (mit anderen Farben), Ein- und Auszoomen, Panning, Dragging (Bewegung von Elementen) und Klicken. [JIANG 1996]

Laut [KRAAK et al. 2003] wird die Visualisierung in einer GDI (*Geospatial Data Infrastructure*) an vier Stellen eingesetzt.

- die Visualisierung kann verwendet werden, um zu entdecken (*explore*), d.h. mit unbekannten Daten umgehen
- sie kann bei der Analyse eingesetzt werden (*analyse*), um bekannte Daten zu manipulieren
- sie kann verwendet werden, um Geodaten zu präsentieren (*present*)
- sie wird ausserdem eingesetzt, um einen leichteren Zugang zu den Daten zu ermöglichen

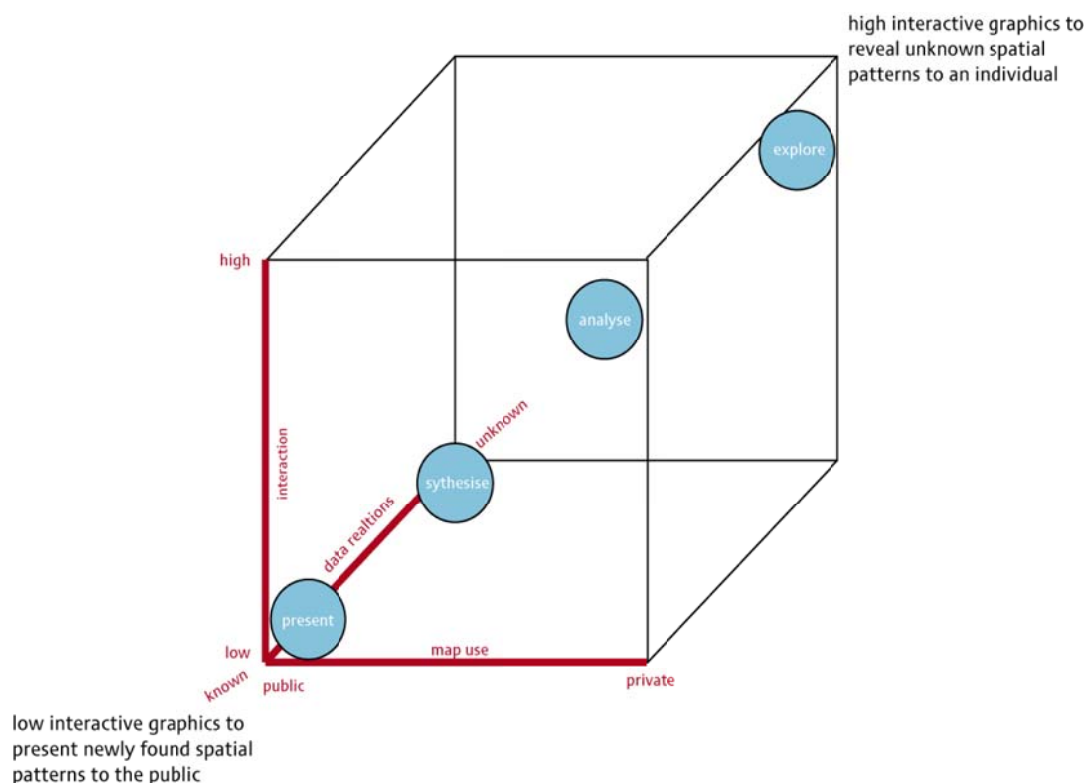


Illustration 1: "Map Use Cube" nach MacEachren und Taylor (1994)

MacEachrens Würfel verdeutlicht, dass es sowohl “private visual thinking” und “public

visual communication” gibt. D.h. es gibt einen Unterschied in der Art wie Daten dargestellt werden, je nachdem ob die Karten für einen breiten Benutzerkreis oder für den Einzelnen gemacht werden. Genauso hängt die Art der Visualisierung von drei Faktoren ab: der Kartenverwendung, der Interaktivität und dem Bekanntheitsgrad der Daten. Neue Daten/Erkenntnisse, die der Allgemeinheit zum ersten mal vorgestellt werden, werden üblicherweise in Karten mit wenig Interaktivität dargestellt. Auf der anderen Seite werden Experten, die mit der Datenmaterie bereits vertraut sind eher mit Karten mit einem hohen Grad an Interaktivität arbeiten. Interaktivität (das Experimentieren mit Visualisierungsparametern) hilft dabei bei der Entdeckung neuer Strukturen und Sichtweisen.

## **2.4 Anforderungen an Bildschirmkarten**

Um eine ansprechende und qualitativ hochwertige Karte im Web erstellen zu können, muss der Kartenautor sowohl kartographische und Programmierkenntnisse haben, als auch Kenntnisse über Layout und Design besitzen.

Um dem Leser ein schnelles und eindeutiges Erfassen einer Karte bieten zu können, muss die Karte gut gestaltet sein. Die ersten Sekunden sind am wichtigsten. In dieser Zeit entscheidet der Nutzer, ob die Karte für ihn ansprechend ist oder nicht. Inhalt, Graphik und Multimedia beeinflussen die Attraktivität einer Bildschirmkarte.

Für die Graphik ist es von Bedeutung, dass das Kartenlayout ansprechend ist und die graphische Bilddichte gut gewählt wurde. Ausserdem müssen die Signaturen klar differenziert und die Signaturformen gut gestaltet werden. Dabei ist zu beachten, dass die Symbole und die Schriften leicht lesbar sind. Die Farben sollen selbsterklärend sein und mit einander harmonieren. Der Inhalt soll aktuell und konsistent sein und eine umfassende Synthese der Thematik geben. Für Multimedia ist die Interaktivität wichtig. Navigieren, eine Abfrage von Objekten nach Attributen, eine Suchfunktion, die Wahl des thematischen Inhalts, die Änderung der Klassierung und Symbolisierung sind nur einige von den möglichen Interaktionspunkten. Zusätzlich zur Multimedia ist die Animation interessant. Ausserdem sollen leicht erkennbare Links für eine thematisch korrekte Verknüpfung sorgen. Es ist ausserdem wichtig, dass ein schneller Download und Bildaufbau erfolgt. [RÄBER, JENNY 2000]

### 3 Möglichkeiten der Umsetzung

Verteilte Webkartographie-Anwendungen folgen immer dem Client-Server Prinzip. Der Client ist der “User Agent”, in vielen Fällen ein Webbrowser. Er kann aber auch eine eigenständige Applikation mit Netzwerkanbindung sein. Schliesslich könnte der Client auch ein mobiles Gerät (PDA oder Mobiltelefon) sein.

Der Server stellt die Daten und die serverseitige Logik (etwa Datenbankschnittstelle, dynamische Generierung von Dateien oder Dateifragmenten, Benutzermanagement, etc.) zur Verfügung. Der Server liefert die Daten und clientseitigen Logikkomponenten an den Client, wenn dieser eine Anfrage stellt.

#### 3.1 Client-seitige Techniken

Um diese Art von Applikation zu implementieren, bieten sich verschiedene Möglichkeiten der clientseitigen Techniken an. Diese Techniken hängen davon ab, was erreicht werden soll und wie komplex das Thema ist.

Java ist eine plattformunabhängige, objektorientierte Sprache. Java Applets können in den meisten Webbrowsern ausgeführt werden. Neuere Java Sprachversionen benötigen jedoch häufig ein Plugin. Java2D ist eine Erweiterung des Java AWT (Abstract Window Toolkit)<sup>2</sup> um Vektorgraphik, Schriftanzeige, Rastermanipulationen, Transformationen, Farbmodelle, graphische Effekte, etc. Die Erweiterung erfolgt durch Hinzufügen neuer Klassen und Methoden. [HENNING 2001] Java2D ist kein Graphikformat, sondern eine hochwertige Graphikbibliothek. Alle Datenformate können gelesen und erstellt werden, wenn es Einleseroutinen dafür gibt. Interaktionen müssen zwar relativ aufwendig programmiert werden, sind daher aber auch sehr flexibel in Bezug auf ihre Möglichkeiten. Die Ergebnisse haben eine sehr hohe graphische Qualität, inklusive Antialiasing<sup>3</sup>, Transparenz und komplexe Textdarstellung. [NEUMANN et al. 2001]

Macromedia Flash ist eine integrierte Entwicklungsumgebung zur Erzeugung von “Flash-Filmen”, und ein proprietäres auf Vektorgraphiken basierendes Graphik- und Animationsformat. Inhalte und Animationen werden in Frames (Zeitleiste) abgelegt. Bei Animationen werden vom Autor Schlüsseleinstellungen festgelegt, das Autorensystem interpoliert dazwischen die gewünschte Anzahl von “in-betweens”. Die so generierten Frames werden dann vom Flash Viewer abgespielt. Flash findet sich häufig auf Internetseiten wieder. Die Quelldateien (\*.fla) werden zur Distribution in \*.swf-Dateien umgewandelt und werden dabei für das Abspielen optimiert. Die \*.fla-Dateien werden nur

---

2 AWT stellt eine Schnittstelle zur Applikationsprogrammierung zur Erzeugung und Darstellung einer plattformunabhängigen graphischen Benutzerschnittstelle für Java-Programme dar.

3 Antialiasing glättet Treppeneffekte, die bei der Rasterisierung von Vektorgraphik, insbesondere bei schrägen Linien auftreten durch zusätzliches Einfügen von Zwischenpunkten welche farblich zwischen der Linie und ihrem Hintergrund interpoliert werden



von den Autoren des Endproduktes während der Entwicklung verwendet und sind veränderbar. Die kompilierten \*.swf-Dateien dagegen sind nur für die Anzeige im Browser gedacht und können nicht mehr verändert werden.

Flash ist weit verbreitet und eine Einbindung von Interaktionen, Animationen, Filme oder Ton ist möglich. Allerdings, lässt Flash sich schlecht durch Suchmaschinen indizieren und braucht ein Plugin. Flashautoren setzen leider oft überflüssige Animationen ein oder verkomplizieren die Bedienabläufe durch graphische Effekte. Daher ist es empfohlen, Flash als Zusatzfunktion (für interaktive Animationen), aber nicht für eine komplette Seite zu benutzen. [NETLEXIKON] Es kommt noch hinzu, dass Flash eine kostenpflichtige Software ist und in ihr nur die Flash-eigene Scriptsprache Actionscript implementiert ist, welche jedoch ECMAScript sehr ähnlich ist. Die serverseitige Generierung von Flashdateien ist wesentlich komplizierter als bei textbasierten Formaten wie SVG, jedoch gibt es bereits eine Anzahl von Flash-Bibliotheken für verschiedene Programmiersprachen. [FIBINGER 2002]

SVG (Scalable Vector Graphics) ist eine XML- Anwendung zur Beschreibung von zweidimensionaler Graphik: Es werden Vektorgraphik, Rastergraphik und Text integriert. Diese sind durch die Verwaltung in Objektbäumen (Document Object Model)<sup>4</sup>, etwa mit ECMAScript, auch manipulierbar und animierbar. Multimedia- Integration und Animation wird durch Integration von SMIL<sup>5</sup> ermöglicht. Näheres zu SVG in Kapitel SVG: Textdateien zum Beschreiben von 2D Vektorgraphik ab Seite 34.

### 3.2 Server-seitige Techniken

In dieser Applikation werden auch serverseitige Techniken benötigt. Folgende Komponenten werden auf dem Server benötigt: Der Webserver zum Ausliefern der Dateien, die geographische Datenbank zum ausliefern von Vektordaten, ein Mapserver zum Ausliefern von Rasterdaten und eine Applikation zum Ausliefern von Geländemodelldaten. Am Server kommt zudem die Programmiersprache PHP<sup>6</sup> zum Einsatz. Sie nimmt Anfragen vom Client entgegen, leitet diese an die Datenbank weiter und generiert die Geometriefragmente die der Client benötigt.

Der MapServer stellt in diesem Fall die Reliefdaten zur Verfügung. Orthophotos werden direkt vom Microsoft Terraserver über eine vom OpenGIS Konsortium definierte Schnittstelle bezogen. Alle Punkte, Linien und Polygondaten werden von Postgis, einer räumlichen Erweiterung von Postgis serviert. "Higrid", eine am Institut für Kartographie der ETH Zürich geschriebene Applikation, extrahiert Geländemodelldaten in verschiedenen Auflösungen und schickt diese an die Online-Applikation. Alle

4 DOM (Objektbaum): beschreibt wie programmierspracheunabhängig auf die Methoden und Eigenschaften von HTML oder XML-Dokumente zugegriffen werden kann.

5 SMIL ist eine Abkürzung für Synchronized Multimedia Integration Language und ermöglicht das Verknüpfen und Synchronisieren von Multimediaelementen, sowie das Beschreiben und Kontrollieren von Animation

6 PHP ist eine "Hypertext Preprocessor", eine in HTML oder andere Textformate eingebettete Programmiersprache, welche aber auch ohne Hostsprache funktioniert. PHP kann mit Datenbanken kommunizieren und bietet Schnittstellen zu vielen anderen Programmierbibliotheken und Programmen.

Komponenten werden in den nachfolgenden Kapiteln ausführlicher erläutert.

### 3.3 Interaktivität

Was ist Interaktivität? Interaktiv stammt aus dem Lateinischen (*inter agere*) und bedeutet “wechselseitig” und “aufeinander bezogen”. In der Kartographie ist die Interaktivität nichts neues, denn eine gewisse Interaktion erfolgt auch während der Auseinandersetzung mit einer analogen Karten.

Es ist die Aufgabe gut gestalteter Benutzeroberflächen, den Nutzer indirekt in verschiedene Richtungen zu weisen, ohne dass etwas vorbestimmt wird. Die Interaktivität multimedialer Anwendungen soll “wissen” welches Ziel der Nutzer hat und ihm dabei helfen es zu erreichen. Für den Nutzer ist es wichtig, dass zur jeder Zeit die Möglichkeiten sichtbar sind, die ihm zur Verfügung stehen, die Informationen über seine ausgeführten Taten ihm zur Verfügung stehen und er über die Auswirkung seiner Taten benachrichtigt wird. [KIRSCH 1997]

[CRAMPTON 2002] definiert die Interaktivität als “ein System, das seine visuellen Daten nach einer Benutzerabfrage darlegt”. Der Nutzer hat die Möglichkeit verschiedene Grade der Interaktivität zu verwenden. Zu einem Grad der niedrigeren Interaktivität gehören die Beleuchtung, der Standpunkt (hauptsächlich für 3D-Anwendungen), ein- und auszuzoomen, Massstabsänderung und Änderung der Symbole. Ein höherer Grad beinhaltet, die Navigation, die Ebenen ein und ausschalten, Sortierung und Überflüge<sup>7</sup>. Der höchste Grad an Interaktivität lässt folgendes zu: Abfragen der Datenbank, Filterung (Ausschluss) und Hervorheben (Aufnehmen), Kombinieren von Datenebenen, Links, mehrere Darstellungen und das Nebeneinanderstellen von Fenstern. Man vergleiche den “Map Use Cube” nach MacEachren und Taylor (1994) im Kapitel “Kartographische Visualisierung” auf Seite 13.

In dieser Applikation steht die Interaktivität an einer zentralen Stelle. Der Nutzer hat die Möglichkeit sämtliche “Befehle” selbst auszuführen und zu kontrollieren. Da die vielen Informationen nicht gleichzeitig gezeigt werden können, sind sie an geeigneten Stellen untergebracht und der Nutzer entscheidet, was er wann machen und sehen will. Durch diese Eingriffe, hat der Nutzer das Gefühl, er würde mit dem System kommunizieren. Während der Benutzung, lernt der Nutzer die Applikation zu bedienen und macht immer komplexere interaktive Abfragen.

---

7 Hier sind virtuelle Flüge über ein Gelände gemeint

## 4 Praktischer Teil: Umsetzung

### 4.1 Datengrundlage

#### 4.1.1 Bearbeitungsgebiet

##### 4.1.1.1 Allgemeines und Geschichte

Das Bearbeitungsgebiet dieser Arbeit ist der Yosemite National Park im US-Bundesstaat Kalifornien. Der 3108 qkm grosse Park liegt in der zentralen Sierra Nevada Gebirgskette und ist ca. 240 km östlich von San Francisco gelegen.



Illustration 2: Karte von Kalifornien mit Yosemite National Park nach USGS

1890 wurde Yosemite zum dritten amerikanischen National Park ernannt. Der wahrscheinlich international bekannteste Park, wurde 1984 auf Grund der schützenswerten Wasserfälle, Granitplatten, klaren Seen und Flüssen, den grossen Sequoia Wäldern und der biologischen Vielfalt, als Weltkulturerbe ausgeschrieben. Innerhalb des Parks befinden sich Tausende von Seen, 1290 km an Wanderwegen, und leider auch 563 km Strassen. Jährlich besuchen mehr wie 3.5 Millionen den National Park, wobei die meisten Besucher sich im 18 qkm grossen Tal aufhalten. [NPS]

Wann man am besten Yosemite besucht, bleibt einem selbst überlassen. Es gibt wundervolle Wintertage, an denen Sonne scheint und der Schnee glänzt, die farbenprächtigen Wiesen und vollen Wasserfälle im Frühjahr, warme und angenehme Tage im

Sommer und fantastisch farbige Laubbäume im Herbst. Im nachfolgenden wird ein Klimadiagramm des Yosemite Valley gezeigt, zur besseren Vorstellung der Klimaverhältnisse im Tal.

Monat	Temperatur	Niederschlag
Januar	3.5 °C	158mm
Februar	5.5 °C	155mm
März	7 °C	132mm
April	10 °C	72mm
Mai	14.5 °C	33mm
Juni	18.5 °C	18mm
Juli	22°C	10mm
August	22 °C	7mm
September	19.5 °C	23mm
Oktober	13.5 °C	53mm
November	6.5 °C	140mm
Dezember	3 °C	142mm
Durchschnitt	18.9 °C	78.6mm

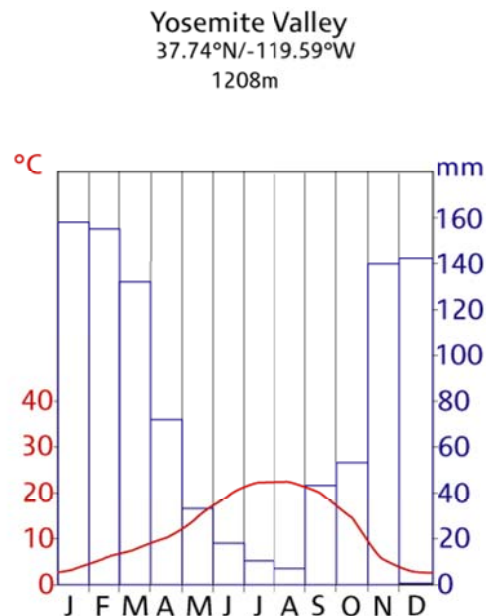


Illustration 3: Klimadiagramm des Yosemite Valley

### [YOSEMITE CA]

Die Ureinwohner der heutigen Yosemite Region waren die Sierra Miwok, wobei die meisten heute in Reservaten (rancherias) wohnen. Ihre Mythologie erzählt, dass ein Substamm, die *Ahwahneechee* (übersetzt als "Volk, das im Tal des klaffenden Mundes wohnt") aus dem Tal stammt und später aufgrund einer Epidemie gestorben ist. Die Überlebenden dieser Epidemie schlossen sich anderen Stämmen an. Um 1850 rief Teneiya, ein Ahwahneechee Häuptling, der bei den Mono Lake Paiutes geboren und aufgewachsen ist, andere Ahwahneechee zusammen und stellte die Heimat seines Volkes im Tal wieder her. Im März 1851 ist das Mariposa Battalion in Yosemite eingeritten und überredete Teneiya und sein Ahwahneechee Volk in einem Reservat zu leben. Im nächsten Winter durften die Ahwahneechee wieder nach Yosemite ziehen, aber die Auseinandersetzungen hielten die nächsten Jahre an und es ist immer noch unsicher ob Teneiya in 1853 durch das Mariposa Battalion oder durch verärgerte Mono Lake Paiute getötet wurde. Namen wie Hetch Hetchy, Kolana, Pohono, Pywiack, Tueeulala, Wampa und Wawona sind das Einzige, was von der Miwoksprache im Yosemite übrig bleibt. Der Name Yosemite stammt wahrscheinlich von *uzumate*, dem Miwok-Wort für Grizzlybär.

1855 kamen die ersten Touristen nach Yosemite, um zu sehen, ob das Tal wirklich so

schön war, wie es beschrieben wurde und sie wurden nicht enttäuscht. Natürlich folgte sofort eine Übernachtungsmöglichkeit und bald wurde das Tal für Bauern aufgeteilt, die einheimische Bäume für Brenn- und Bauholz fällten und Gärten und Obstbäume anpflanzten. Sie brachten Pferde, Kühe, Hühner und Schafe mit, die die Blumen und Gräser in den höheren Lagen zertrampelten.

Frederick Law Olmsted, ein Landschaftsarchitekt und der Gestalter vom Central Park in New York, war der Meinung, dass sich die Regierung zum Schutz natürlicher Räume verpflichten sollte. Zusammen mit Israel Ward Raymond, der um das Schicksal der Bäume im Yosemite besorgt war, schrieb er einen Brief an Senator Conness, mit einem Vorschlag für ein Gesetz, dass das Yosemite Tal und der Mariposa Grove zum Staat gehören würde. Am 30. Juni 1864 wurde Yosemite zum ersten *State Park* der Welt.

1889 überlegten John Muir und Robert Underwood Johnson, wie die grösseren Einzugsgebiete der Tuolumne und Merced Flüsse vor den Schafen und kommerziellen Interessen geschützt werden könnten. Am 1. Oktober 1890 wurde Yosemite zum National Park, welcher die Einzugsgebiete der Flüsse beinhaltet. Allerdings blieben das Tal und Mariposa Grove Staatseigentum. 1906, nach intensiver Unterstützung von John Muir und dem Sierra Club<sup>8</sup>, übergab der Staat Kalifornien das Yosemite Tal und Mariposa Grove an den National Park.

Der Tourismus nahm zu und Ende des 19. Jahrhunderts gab es sogar 9 Hotels im Park. Das erste Auto kam im Sommer 1900. Zwei Jahre später, als zwei weitere Autos kamen, wurden Autos im Park verboten, denn es wurde beschlossen, dass sie eindeutig fehl am Platz waren. 1907 fuhr die neue Yosemite Valley Railroad von Merced bis zur Parkgrenze in El Portal, wo Postkutschen die Besucher abholten und ins Tal brachten. Die Bahn hätte vielleicht ihre grosse Rolle als Beförderungsmittel beibehalten können, wenn nicht eine Überschwemmung 1937 einen Teil der Schienen weggespült hätte. Zwar fuhr die Bahn im nächsten Jahr wieder, aber bis dahin hatte der Autoverkehr zugenommen und war das beliebteste Verkehrsmittel. Die Zugverbindung wurde 1945 stillgelegt, wobei die Wiederherstellung der Eisenbahnlinie eine sehr beliebte aber politisch schwierige Idee bleibt.

Nachdem das Autoverbot 1913 aufgehoben wurde, kamen mehr als 12,000 jährliche Besucher nach Yosemite. Als eine unbefestigte Strasse über den Tioga Pass 1916 eröffnet wurde, kamen mehr als 30,000 Besucher. 1922 waren es mehr als 100,000 und als die Auto 1929 ein Problem wurden, wurden Strassengräben ausgehoben, um die Autos von den Wiesen fernzuhalten. 1926 wurde der Highway 140 (der "All-Year-Highway") eröffnet. Um die immer grösser werdenden Besucherzahlen zu bewältigen, wurde Anfang 1970 ein System von Einbahnstrassen, eine kostenloser Pendelbus und ein Quotensystem für entlegene Gegenden (Backcountry) eingeführt.

Als 1980 die Besucherzahlen auf 2,500,000 kletterten, konnte man mehr als 1000 Gebäude im Tal zählen. Die ca. 48 Strassenkilometer, wurden von mehr als einer Million

---

<sup>8</sup> Sierra Club: eine von Muir gegründete Umwelt Organisation und heute die grösste und einflussreichste in Amerika (<http://www.sierraclub.org>)

Fahrzeugen im Jahr benutzt. Da es vorauszusehen war, dass die Besucherzahlen weiterhin steigen würden, beschloss der National Park Service seinen ersten “General Management Plan”. Darin wurden Ziele genannt wie die Konservierung der Wildnis. Es wurde vorgeschlagen Restriktionen auf Privatfahrzeuge zu legen und den öffentlichen Verkehr zu verdichten. Das ultimative Ziel, alle Privatfahrzeuge im Tal zu verbieten, wurde zu einem Streitpunkt. Weiterhin sollten kommerzielle Dienstleistungen ausserhalb des Parks angesiedelt werden. Überlegungen zur Flussebene, führten zu Änderungen bei den Zeltplätzen. Als der National Park Service seinen Plan umsetzen wollte, mussten wegen negativen Reaktionen der Öffentlichkeit, Änderungen gemacht werden. 1992 wurden die Vorschläge zur Art und Anzahl von Übernachtungsmöglichkeiten modifiziert. Nach der Überschwemmung im Januar 1997 wurde erneut über die Entwicklung der Flussebene diskutiert. Die Campingplätze, die am Fluss lagen und während der Überschwemmung vernichtet wurden, wurden nicht wieder aufgebaut. Der neue Plan (Yosemite Valley Plan vom 29. Dezember 2000) sieht eine Restaurierung und den Schutz der natürlichen Ressourcen sowie eine Verringerung der schädlichen Auswirkungen auf die Umwelt (Autos miteinbezogen) vor. Die Umsetzung des Plans verzögert sich. Ausserdem wird noch über die Wichtigkeit der relativen Werte dieser Ressourcen und über ideale Verbrauchermuster diskutiert. Im Februar 2004 wurde ein “User Capacity Management Program for the Merced Wild and Scenic River Corridor” aufgesetzt. Dieses Programm soll eine “Richtung und Beratung anbieten wie Besucher-nutzung, Erschliessung von Land und Gebäudeanlagen und Schutz der Ressourcen geführt werden sollen”. Weitere Informationen dazu unter: <http://data2.itc.nps.gov/parks/yose/ppdocuments/ucmpdoc.pdf>. Anfang 2005 wurden die alten Dieselsebusse endlich durch Elektro-Diesel Busse ausgetauscht.

Obwohl die höchste Besucherzahl von 1996 (4.2 Millionen) seitdem glücklicherweise nicht wieder erreicht wurde, strömen immer noch jährlich Massen in den Nationalpark. 2004 wurden 3.468.954 Besucher registriert. In den zukünftigen Plänen müssen sich die Bedürfnisse der Besucher mit der Konservierung der natürlichen Schönheit des Parkes die Waage halten. [NPS] [LONELYPLANET 2002]

#### 4.1.1.2 Entstehung und Geologie

Vor 225 Millionen Jahren war die Sierra Nevada ein flacher See. Langsam füllte das Material von Inselvulkanen und die erodierende Landmasse den See mit Sedimenten. Die nordamerikanische Platte schob sich im Westen über die pazifische Platte. Die pazifische Platte schmolz und formte den Sierra Nevada Batholith. Die Sierra Nevada entstand aus mehreren “magmatischen Schmelzungen”, die jeweils eine Granitart formte. Als der Granitbatholith sich nach oben schob, wurde das alte Gestein mit nach oben bewegt. Reste von dieser Verschiebung gibt es bei Mt. Dana und im Ritter Range. [LONELYPLANET 2002]



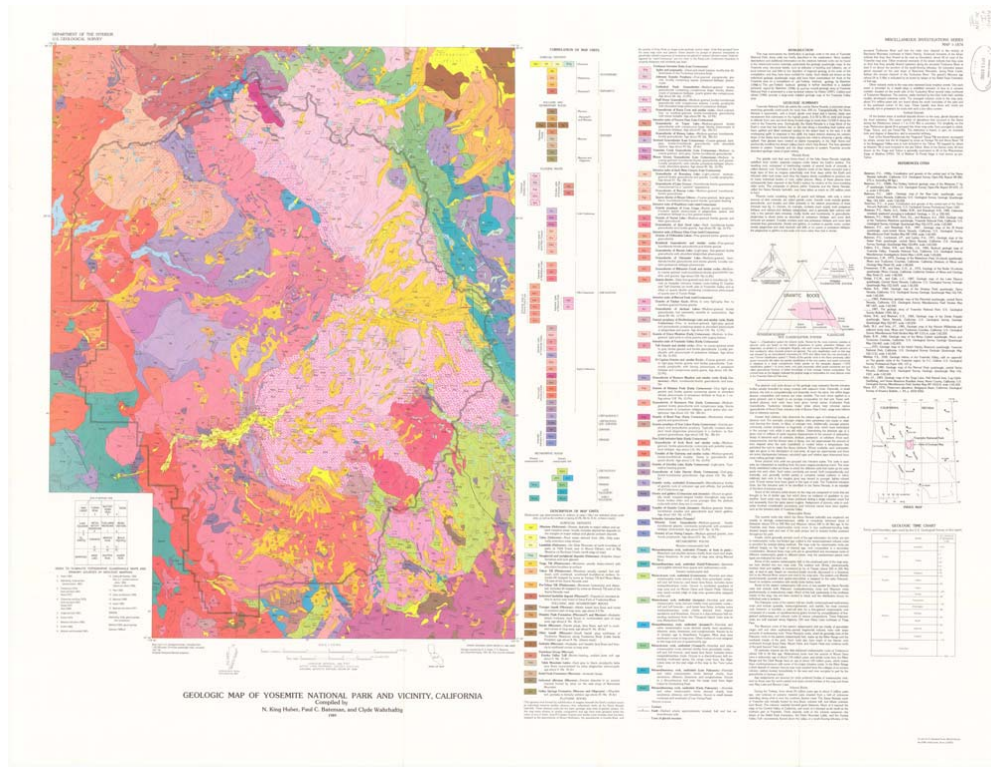


Illustration 4: Geologische Karte des Yosemite National Parks und Umgebung von N. King Huber (1989)

Die vorkommenden Gesteine lassen sich in den nachfolgenden Zeitaltern einordnen:

Trias (*Triassic*): 248-206 Millionen Jahre

Jura (*Jurassic*): 206-144 Millionen Jahre

Kreidezeit (*Cretaceous*): 144-65 Millionen Jahre

Paleozän (*Paleocene*): 65-54.8 Millionen Jahre

Eozän (*Eocene*): 54.8-33.7 Millionen Jahre

Oligozän (*Oligocene*): 33.7-23.8 Millionen Jahre

Miozän (*Miocene*): 23.8-5.3 Millionen Jahre

Pliozän (*Pliocene*): 5.3-1.8 Millionen Jahre

Pleistozän (*Pleistocene*): 1.8 Millionen Jahre - 10000 Jahre

Holozän (*Holocene*): 10000 Jahre – heute [UCMP BERKELEY]

Yosemite Valley wird oft als das “klassische” glaziale Tal bezeichnet. Der Sherwin Gletscher war der am längsten anhaltende und am weitesten ausgebreiteste Gletscher in

der Sierra Nevada. Er existierte wahrscheinlich 300 Tausend Jahre und schmolz vor ca. einer Millionen Jahren. Dieser Gletscher ist höchstwahrscheinlich für die Morphologie des Yosemite Tals innerhalb des Merced Canyons verantwortlich. Spätere Gletscher wie der Tahoe und der Tioga Gletscher waren weniger prägend für die Form. Die anormalen Spitzen und Türme konnten durch die Tahoe und Tioga Gletscher nicht geglättet werden, da die Schnee- und Eisdecke nicht so hoch reichte wie beim Sherwin Gletscher. Im Tenaya Canyon war das Eis dicker und konnte dort glätten und runden. Das Hetch Hetchy Tal, das sonst sehr ähnlich dem Yosemite Tal ist, hat mehr "Rundungen", da alle Gletscher das Tal bis zum Rand füllten. Als der Tioga Gletscher sich zurückzog, hinterliess er einen See im Tal. Die Arme des Gletschers zogen sich in die Tenaya und Merced Canyons zurück und das Schmelzwasser füllte den See mit Sedimenten. Der heutige relativ ebene Talboden stammt aus dieser Zeit.

Das Gestein im Yosemite besteht aus grossteils aus metamorphem Gestein. Tonschiefer, Sandstein, Kalkstein, und vulkanisches Gestein wurden zu Plutoniten umgewandelt. Die Hunderten von Plutoniten in der gesamten Sierra Nevada werden als der "Sierra Nevada Batholith" bezeichnet und könnten bis zu 130 Millionen Jahre für ihre Entstehung benötigt haben. Die plutonischen Gesteine des Tals und das angrenzende Oberland sind aus vielen Mineralien zusammengesetzt. Das Gestein besteht hauptsächlich aus Quarz, Kaliumfeldspat und Plagioklas, Biotit und Hornblende.

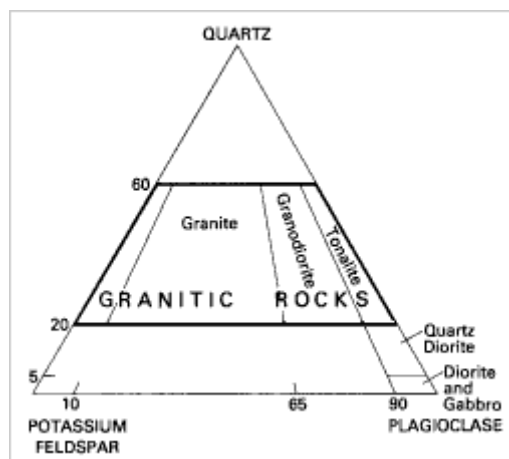


Illustration 5: Klassifikation der plutonischen Gesteinsnamen, nach Prozentanteilen von Quarz, Kaliumfeldspat und Plagioklas (Strecheisen, 1973)

Die ältesten plutonischen Gesteine befinden sich am westlichen Ende des Tals. Half Dome, das Wahrzeichen des Yosemite, besteht aus dem jüngsten plutonischen Gestein, ca. 87 Millionen Jahre alt. [HUBER 1985]



#### 4.1.1.3 Vegetation

Die Yosemite Region besitzt eine der grosse Vielfalt an Pflanzenarten. Sie hat Ökosysteme, die vom Zentralen Tal (Central Valley) bis in alpine Regionen und im Osten bis in die Great Basin Wüste reichen. Yosemite National Park alleine weist 23% der Pflanzenarten, die in Kalifornien vorkommen, auf. Auf dem Dana Plateau, oberhalb des Tioga Passes, kann man auf 2.6 qkm 50% aller alpinen Pflanzen, die in der Sierra Nevada Region vorkommen, finden. [LONELYPLANET 2002]

Die Bodenbedeckung (*Landcover*) wird in der Legende erklärt und führt folgende Klassen auf:

Bäume/Wald: *Trees/Forest*

Sträucher: *Shrubs*

Seggen: *Sedge*

Spärliche Vegetation: *Sparse Vegetation*

Geflutete krautartige Vegetation: *Flooded Herbaceous*

Geröll: *Scree*

Fels/Gestein: *Rock*

Wasser: *Water*

Schnee/Gletscher: *Snow/Glacier*

Siedlung: *Urban Area*

Beispiele für Bäume, die im Yosemite vorkommen:



Illustration 6: Lodgepole Pine (*Pinus contorta* ssp. *Murrayana*); © Br. Alfred Brousseau, Saint Mary's College



Illustration 7: Giant Sequoia (*Sequoiadendron giganteum*); © Andreas Neumann



Illustration 8: Ponderosa Pine (*Pinus ponderosa*); © Robert Potts, California Academy of Sciences



Illustration 9: Jeffrey Pine (*Pinus jeffreyi*); © Urs Mattle



Illustration 10: Whitebark Pine (*Pinus albicaulis*); © Robert Potts, California Academy of Sciences



Illustration 11: Sugar Pine (*Pinus lambertiana*); © Br. Alfred Brousseau, Saint Mary's College



Illustration 12: Red Fir (*Abies magnifica*); © Neva Snell



Illustration 13: Quaking Aspen (*Populus tremuloides*); © Andreas Neumann





Illustration 14: Live Oak (*Quercus virginiana*); © Andreas Neumann



Illustration 15: Black Oak (*Quercus kelloggii*); © Charles Webber, California Academy of Sciences

## Sträucher:



Illustration 16: Huckleberry Oak (*Quercus vaccinifolia*), © Steve Matson



Illustration 17: Greenleaf Manzanita (*Arcostaphylos patula*); © William R. Hewlett, California Academy of Sciences



Illustration 18: Gooseberries (*Ribes uva-crispa*); © Juliana Williams



Illustration 19: Whiteleaf Manzanita (*Arctostaphylos viscida*); © Br. Alfred Brousseau, Saint Mary's College



Illustration 20: Currents (*Ribes rubrum*); © Juliana Williams



Illustration 21: Deer Brush (*Ceanothus integerrimus*); © Christopher L. Christie



Illustration 22: Poison Oak (*Toxicodendron diversilobum*); © Charles E. Jones



Illustration 23: Mountain Misery (Kitkitdizze) (*Chamaebatia foliolosa*); © Albert P. Bekker, California Academy of Sciences

## Wildblumen:



Illustration 24: California Poppy (*Eschscholzia californica*); © Jo-Ann Ordano, California Academy of Sciences



Illustration 25: Bigelow's Sneezeweed (*Helenium bigelovii*); © Br. Alfred Brousseau, Saint Mary's College





Illustration 26: Jeffrey Shooting Star (*Dodecatheon jeffreyi*); © Dr. Samuel J. Pusateri



Illustration 27: Indian Paintbrush (*Castilleja minutata*); © Andreas Neumann



Illustration 28: Lupine (*Lupinus*); © Gary A. Monroe



Illustration 29: White Mountain Heather (*Cassiope mertensiana*); © John Game



Illustration 30: Sierra Primrose (*Primula suffrutescens*); © Steve Matson



Illustration 31: Sky Pilot (*Polemonium viscosum*); © Gerald and Buff Corsi

#### 4.1.2 Datensätze

Die Datensätze wurden grösstenteils aus der digitalen Datenbank des USGS (United States Geological Survey) [USGS] bezogen. Die Daten sind frei erhältlich und können jederzeit aus dem Internet heruntergeladen werden. Die Datenbasis für die ESRI ArcGIS-Bearbeitung befindet sich in einer ESRI Geodatabase. Die ursprünglichen Files des USGS lagen im ESRI e00-Format vor und mussten zuerst in die Personal Geodatabase eingelesen werden. Die restlichen Daten stammen aus der Datenbank des National Park Services oder wurden auf Anfrage direkt vom Yosemite National Park bezogen. Diese Daten wurden ebenfalls in die Personal Geodatabase importiert. Die Hydrologiedaten wurden vom NHD (National Hydrology Dataset) heruntergeladen und ebenfalls in eine Personal Geodatabase übernommen. Die Orthophotos befinden sich auf dem MS (Microsoft) Terraserver und werden von dort direkt über eine Netzwerkschnittstelle bezogen.

Die Datensätze befanden sich nicht alle in der gleichen Projektion und/oder waren von unterschiedlichem Ellipsoid-Datum<sup>9</sup>. Alle Daten mussten daher zuerst in die richtige

<sup>9</sup> Ein Ellipsoid-Datum ist eine Gruppe von Parametern und Kontrollpunkten um die dreidimensionale Form der Erde beschreiben zu können. Das Datum ist die Basis für ein planares Koordinatensystem. Beim USGS kommen v.a. zwei Ellipsoid-Daten zum Einsatz: das ältere NAD27 Datum, welches v.a. bei älteren GIS-Daten und gescannten topographischen Rasterdaten zum Einsatz kommt, sowie das neuere NAD83 Datum, welches bei Orthophotos und neueren GIS-Daten verwendet wird. NAD ist die Abkürzung für "North American Datum".



Projektion und in das korrekte Ellipsoid-Datum gebracht werden. Die gewählte Projektion ist UTM (Universal Transverse Mercator), in der Zone 11N. Die gewählte Längeneinheit ist Meter. Höhenangaben werden sowohl als Feet als auch als Meter angegeben. Der Kartenmassstab wird sowohl in Metern/Kilometern als auch in Meilen/Yards angegeben. Das Datum ist NAD83, das Ellipsoid ist GRS80 (Geodetic Reference System), welches praktisch idente Parameter wie das WGS84 Ellipsoid aufweist und daher auch in den meisten GPS Geräten zum Einsatz kommt. Die Umprojektion erfolgte ebenfalls in ESRI ArcGIS.

Nach der Projektion wurden alle Daten in ArcGIS nochmals überarbeitet und verbessert. Vielen Ebenen wurden zusätzliche Attributen hinzugefügt, um die Karteninformationen zu vervollständigen. Nachdem alle Daten bearbeitet wurden, wurden sie in das ESRI Shapefile exportiert und damit in die PostgreSQL/PostGIS Datenbank übertragen. In der Applikation werden sie in dieser Datenbank verwaltet und von dort abgefragt.

Um die Koordinaten gleichzeitig im UTM System (Meter) als auch in geographischen Koordinaten (Altgrad) anzeigen zu können, musste für die Umrechnung ein eigenes ECMAScript integriert und angepasst werden. [SNYDER 1987] [TAYLOR]

Hier eine Liste der verwendeten Datensätze und ihre Quellen:

- USGS
  - Hypsographie (Höhenlinien und Höhenkoten)
  - Geologie
  - Geländemodelle
  - Namensdatenbank
- National Park Service
  - Strassen
  - Wanderwege
  - Nationalparkgrenze
  - Gebäude
  - Vegetation (detailliertere Version)
  - handgezeichnetes Relief
- NHD: Hydrologie (High: nicht generalisierte Daten)
- US Terraserver: Orthophotos
- Tom Harrison Maps: Vegetation (gröbere Version)

## 4.2 Gewählte Systemarchitektur, Technische Grundlagen

### 4.2.1 XML: Markupsprache zum Abspeichern heterogener, hierarchischer Informationen

XML steht für “eXtensible Markup Language” und ist eine ausbaufähige Auszeichnungssprache, die 1998 vom W3C<sup>10</sup> definiert wurde. XML entspricht der Syntax von SGML (Standard Generalized Markup Language), ihr Vorgänger, wurde allerdings für das Internet v.a. in Hinsicht auf Verlinkung und Lokalisierung optimiert. [FIBINGER 2002]

XML ist eine Metamarkupsprache, was bedeutet, dass sie keine vordefinierten Elemente und Attribute hat. Jeder kann nach Bedarf seine eigenen Datenstrukturen definieren. Trotzdem gibt es feste Regeln wie und wo Attribute gesetzt werden dürfen, welche Elementnamen legal sind, wie Attribute an Elementen angehängt werden, etc. Dadurch können allgemein gültige Parser oder analysierende Systeme entwickelt werden, die XML lesen und verstehen können. Dokumente müssen “wohlgeformt” (well-formed) und “gültig” (valid) sein, also keine Fehler in der Grammatik der XML-Sprache haben, damit sie korrekt eingelesen werden können. Dokumente werden mit Hilfe eines XML-Schemas auf ihre Gültigkeit überprüft. In diesem Schema werden die gültigen Element- und Attributnamen, die korrekte Verschachtelung und die verwendeten Datentypen definiert. [HAROLD et al. 2001]

XML wird in dieser Applikation für die Wanderweginformationen und für die Photogalerie verwendet. Der Einsatz von XML lohnt sich an allen Stellen bei denen Inhalt und Präsentation sauber getrennt werden soll. Wenn diese Trennung konsequent eingehalten wird, so kann das System sehr einfach an neue Daten oder Regionen angepasst werden. Um das System etwa für einen neuen Nationalpark anzupassen, bedarf es daher wesentlich geringerem Aufwand als bei der erstmaligen Erstellung der Applikation.

Nachfolgend ein Beispiel der Wanderweg-Daten (XML Fragment):

```
<trail name="Sentinel Dome" distance="3.5/2.2" duration="2"
difficulty="moderate" location="Glacier Point Road" elev_gain="111/370"
season="May through October" map="USGS Half Dome " xmin="272056"
xmax="272892" ymin="4177532" ymax="4178490" campground="Bridalveil Creek
Campground">
  <start_point>On north side of Glacier Point Rd., 20.9 km (13 mi) from
  Chinquapin</start_point>
  <end_point>Sentinel Dome</end_point>
  <links>
    <link url="http://www.nps.gov/yose/trip/glacierhikes.htm"
    title="Glacier Point Road Day Hiking (NPS)"/>
  </links>
  <segments>
```

<sup>10</sup> W3C (World Wide Web Consortium) ist ein internationales Konsortium, das mit Hilfe zugehöriger Organisationen, festangestellte Mitarbeiter und der Öffentlichkeit daran arbeitet Webstandarde weiter und zu entwickeln.

```

<segment id="470" flip="false"/>
<segment id="460" flip="true"/>
<segment id="457" flip="true"/>
<segment id="450" flip="false"/>
<segment id="454" flip="false"/>
</segments>
<features rocks="1" water="0" nice_view="1" flowers="0"
less_crowded="0" children="1"/>
</trail>

```

### Ausschnitt des dazugehörigen XSD Schemas:

```

<xs:element name="trail">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="start_point"/>
      <xs:element ref="end_point"/>
      <xs:element ref="tips" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="photos" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="vista_points" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="swimming_spots" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="links" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="segments" maxOccurs="unbounded"/>
      <xs:element ref="features"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="distance" type="xs:string" use="required"/>
    <xs:attribute name="duration" type="xs:string" use="required"/>
    <xs:attribute name="difficulty" type="xs:string" use="required"/>
    <xs:attribute name="location" type="xs:string" use="required"/>
    <xs:attribute name="elev_gain" type="xs:string" use="required"/>
    <xs:attribute name="season" type="xs:string" use="required"/>
    <xs:attribute name="map" type="xs:string" use="required"/>
    <xs:attribute name="xmin" type="xs:integer" use="required"/>
    <xs:attribute name="xmax" type="xs:integer" use="required"/>
    <xs:attribute name="ymax" type="xs:integer" use="required"/>
    <xs:attribute name="ymin" type="xs:integer" use="required"/>
    <xs:attribute name="campground" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="trails">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="trail" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

#### 4.2.2 SVG: Textdateien zum Beschreiben von 2D Vektorgraphik

SVG steht für Scaleable Vector Graphics. SVG beschreibt zweidimensionale Graphiken in XML und kann Vektorgraphiken, Rastergraphiken und Text integrieren. Ausserdem

hat SVG noch die Möglichkeit Animationen zu integrieren. Diese graphischen Objekte können gruppiert, transformiert und in bereits erstellten Objekten integriert werden. Zeichnungen können interaktiv sein und animiert werden. Animationen können entweder deklarativ, d.h. durch eine Einbettung von SVG Animationselementen in den SVG-Inhalt, oder durch ein Skript definiert und kontrolliert werden. [W3C,SVG]

Mit SVG werden Vektoren gezeichnet. Diese Vektoren sind Pfade mit jeweils einem Anfangs- und Endpunkt, die durch kartesische Koordinaten definiert sind. Der Ursprung des Koordinatensystems befindet sich am Bildschirm links oben. Man beachte dabei, dass die positive y-Achse nach unten zeigt. Die Informationen zu den Vektoren befinden sich im Quelltext (Code) und können dadurch von jedem eingesehen, geöffnet und verändert werden. Diese Tatsache stellt einerseits einen Vorteil dar (es kann vom Code anderer Leute gelernt werden), andererseits einen Nachteil dar (einige Firmen befürchten, dass ihre Kartengraphik missbräuchlich kopiert werden könnte). In Fällen bei denen die Kartenausschnitte immer individuell und automatisch aus Datenbanken generiert und auch für den jeweiligen Massstab generalisiert werden, stellt dieser Umstand jedoch kein grösseres Problem dar. Der Kartennutzer bekommt lediglich einen in der Genauigkeit reduzierten Ausschnitt und nie den gesamten Datensatz in seiner vollen Auflösung zu Gesicht. Schliesslich könnten bei Bedarf auch digitale Wasserzeichen in die Vektordaten eingebaut werden. Für zukünftige SVG-Versionen ist dennoch ein "Digital Rights Management" angedacht.

Häufig haben SVG Anwendungen die folgende Dateistruktur: Am Anfang werden zentrale Definitionen (`<defs>`) wie etwa CSS Eigenschaften, Symbole, Marker, Farbverläufe, Muster etc. festgelegt. CSS (Cascading Stylesheets) werden benutzt, um global Formate für Farben, Texte, Strich- und Fülldefinitionen etc. zu definieren. In der eigentlichen Graphik oder Applikation wird dann auf diese Definitionen zurückgegriffen (`<use/>` oder `class`), damit ein Element oder dessen Eigenschaften nicht immer wieder von Grund auf neu definiert werden muss. Für Symbole gilt dasselbe. Sie werden global festgelegt und später instanziiert. Dabei können sie noch skaliert, rotiert, transformiert oder animiert werden. Im zentralen Teil der SVG Anwendung befindet sich das Layout und Benutzerinterface. Inhalte werden dynamisch aus XML-Dateien und/oder Datenbanken in die Applikation geladen. Interaktivität und die Logikkomponente der Anwendung werden in ECMAScript formuliert.

Warum SVG? SVG bietet einige Vorteile:

- SVG kann in jedem Texteditor erstellt und geändert werden. Zur Erstellung, Generierung und zum Abspielen von SVG-Dateien sind keinerlei Software oder Lizenzgebühren notwendig.
- SVG liegt im Text-Format vor und ist XML-Strukturiert
- SVG ist zu vielen anderen Webtechnologien kompatibel und integriert sich mit diesen sehr gut. Der Lernaufwand für Entwickler, die sich bereits mit anderen Webtechnologien auskennen, ist daher niedriger. Beispiele für mit SVG integrierte oder

integrierbare Webtechniken sind CSS, ECMAScript, DOM, XML, XSLT, DTD, Schema, HTML, JPEG, PNG, MPEG, MP3, etc.

- Graphiken und Animationsmechanismen können sehr einfach mit Programmiersprachen wie ECMAScript (besser bekannt als JavaScript), Perl, PHP generiert oder modifiziert werden. ECMAScript wird meist auf dem Client für die Manipulation und Steuerung der Applikation eingesetzt. Perl, PHP und andere Programmiersprachen werden auf dem Server zur Generierung von SVG Code verwendet.
- Viele Ressourcen (Rasterbilder, Javascripte, CSS, Definitionen etc.) können in externen Dateien abgelegt werden und von mehreren Anwendungen gleichzeitig genutzt werden.
- SVG unterstützt den standardisierten Farbraum sRGB und ICC<sup>11</sup>-Profile können referenziert werden, mit denen Farben sich in andere Farbräume transformieren lassen
- In SVG wird bei Animationen der Anfangs- und Endzustand, sowie die Interpolationsart festgelegt, denn es wird zwischen den Veränderungen interpoliert. Es sind noch die Key/Value-Pairs<sup>12</sup> von Bedeutung. Dadurch kommt es zu kompakterem und besser strukturiertem Code. Animationen in SVG/SMIL definieren im Gegensatz zu Macromedia Flash keine fixe Framerate. Computer berechnen so viele Zwischenframes wie ihnen Rechenkapazität zur Verfügung steht. Schnelle Computer berechnen daher flüssigere Animationen als langsamere Rechner. [FIBINGER 2002]
- SVG bietet zahlreiche gestalterische Möglichkeiten und graphische Effekte, wie z.B. Farbverläufe, vektorbasierte Muster, Textausrichtung entlang von Pfaden, Texteffekte, Filter, etc. SVG Viewer bieten meist sehr hohe graphische Qualität.
- Die Dokumentation von SVG ist sehr gut. Es existieren umfangreiche Spezifikationen mit Beispielen, Bücher, Online Ressourcen, Tutorials, etc. Eine aktive Entwicklergemeinschaft hilft bei Problemen bei der Programmierung und Erstellung von SVG Anwendungen.
- SVG ist nicht von einzelnen Herstellern/Firmen abhängig. Viele Firmen, Konsortien, Universitäten und Einzelpersonen engagieren sich für die Weiterentwicklung des SVG Standards und dafür benötigte Softwarekomponenten.
- SVG unterstützt das “Verschachteln” von Koordinatensystemen, sowie das Anzeigen von Ausschnitten von grösseren Datensätzen. So kann etwa ein oder mehrere “Weltkoordinatensysteme” innerhalb des Bildschirmkoordinatensystems definiert werden. Dieser Umstand erleichtert das Integrieren verteilter Geodatenquellen.

11 ICC (International Color Consortium): plattformübergreifendes Geräteprofil-Format zur farbmtrischen Kalibrierung der Geräte [HENNING 2001]

12 Key-Value Pair: Jeder Zeitpunkt hat einen Wert zwischen 0 und 1 und dem Zeitpunkt werden bestimmte Ortswerte zugewiesen.

- SVG unterstützt Netzwerkschnittstellen und das Nachladen von Geometrie. Im Gegensatz zu vielen HTML Anwendungen muss nicht die gesamte Website neu geladen werden um Geometrie zu ergänzen, sondern lediglich der neue Kartenausschnitt. Das Benutzerinterface bleibt jeweils ohne Neuladen erhalten.

Trotz allem ist SVG kein Allheilmittel für kartographische und graphische Probleme. Komplexe kartographische Symbolisierungen stellen trotz der vielen Möglichkeiten immer noch Herausforderungen dar. Die Unterstützung mancher Softwarehersteller könnte noch umfangreicher sein und die Kompatibilität von SVG Autorentools ist nicht immer gewährleistet. Gerade für komplexe SVG Anwendungen, bei denen sehr viel client- und serverseitig programmiert werden muss und mehrere Anwendungen integriert werden müssen, fehlen komfortable und effiziente Entwicklungsumgebungen.

Ein weiterer Nachteil ist, dass im Vergleich zu Macromedia Flash, die Anzahl der installierten SVG-Plugins kleiner ist. Allerdings ist die Wahrscheinlichkeit gross, dass dies sich ändern wird.

Schliesslich ist SVG derzeit noch ein rein graphisches Dateiformat ohne Unterstützung von Topologie oder Netzwerkstrukturen, wie sie bereits in vieler GIS Software oder räumlichen Datenbanken unterstützt werden. Topologie oder Netzwerkanalysen müssen daher meist auf dem Server berechnet werden oder müssten durch eigene Programm- und Datenlogik innerhalb der SVG Applikation selbst dazuprogrammiert werden. [NEUMANN et al. 2001]

Wie bereits erwähnt wird innerhalb von SVG ECMAScript zur Erstellung von Interaktivität und Programmlogik verwendet.

#### 4.2.3 ECMAScript (JavaScript): Clientseitige Programmiersprache

JavaScript, unter dem Namen ECMAScript standardisiert, ist eine objektorientierte Programmiersprache, die die dynamische Gestaltung einer HTML-Seite ermöglicht. Die Seite kann also Programme beinhalten, die mit dem Benutzer interagieren und den Webbrowser kontrollieren. [FLANAGAN 2002] Ausserdem kann sie als eine Makrosprache<sup>13</sup> für andere Software dienen.

Die Benutzung von ECMAScript ist nicht auf Webbrowser begrenzt. ECMAScript wird oft benutzt, um das SVG-Dokument zu manipulieren. Dabei können individuelle Funktionalitäten implementiert werden. Im SVG Dokument wird eine Referenz auf die jeweilige ECMAScript-Datei gesetzt z.B.

```
<script type="text/ecmascript" xlink:href="trails.js"/>
```

---

<sup>13</sup> Makrosprache: einfache Form einer Programmiersprache. Sie dient dazu, häufig verwendete Aktionen in Form von Makros zu automatisieren.

Damit weiss das SVG auf welche Scriptdatei es zugreifen muss, um eine Funktion ausführen zu können.

#### 4.2.4 PostgreSQL mit PostGIS spatial extension: Geographische Datenbank

PostgreSQL ist ein hochentwickeltes objektrelationales Datenbankverwaltungssystem das im Quellcode erhältlich ist und ohne Lizenzgebühren einsetzbar ist. PostGIS ist eine räumliche Erweiterung von PostgreSQL, erstellt von der Firma Refractions (<http://postgis.refractions.net/>). Die nachfolgenden Erklärungen stammen aus der PostGIS Dokumentation [POSTGIS] und der carto.net Website [CARTO.NET]. Alle Beispiele sind aus der Applikation entnommen.

##### 4.2.4.1 Eigenschaften von PostgreSQL und PostGIS

- SQL92 Kompatibilität<sup>14</sup>
- Unterstützung von:
  - Foreign Keys (Fremdschlüssel)<sup>15</sup>
  - Triggers<sup>16</sup>
  - Transaktionen, Commits und Rollbacks<sup>17</sup>
  - Views<sup>18</sup>, mit Schreibzugriffsmöglichkeit
- komplexe Abfragen mit Unterabfragen, auch geschachtelt
- Schnittstellen zu verschiedenen Programmiersprachen (Client)
- erweiterbar mit verschiedenen Programmier- und Scriptingsprachen (neue Datentypen, Funktionen oder Indizes können erstellt werden)
- geringe Systembelastung und hohe Geschwindigkeit
- problemloser Export und Import von Daten und Datenbankstrukturen

PostGIS ist eine Erweiterung der PostgreSQL Datenbank für geographische Daten. Dabei kann PostgreSQL als Datenserver für Geoinformationssysteme dienen. Folgende Funktionalitäten beinhaltet PostGIS:

---

<sup>14</sup> Standardisierte Datenbank-Abfrage- und -Manipulationssprache

<sup>15</sup> Ein Attribut (Spalte) einer Tabelle, das auf ein primäres Attribut einer anderen Tabelle verweist.

<sup>16</sup> Überwacht Änderungen in der Datenbank und führt bei Bedarf Funktionen aus in Abhängigkeit der geänderten Daten. Werden vor allem für die Datenkonsistenz benötigt.

<sup>17</sup> Eine Transaktion ist eine Abfolge von Operationen, die komplett oder gar nicht durchgeführt wird. Eine Transaktion ist dann abgeschlossen, wenn sie mit einem Commit bestätigt wurde. Im Falle eines Abbruchs einer Transaktion muss ein Rollback (Rückführung in den ursprünglichen Zustand) durchgeführt werden.

<sup>18</sup> Ausschnitte aus einer oder mehreren anderen Tabellen

- Unterstützung von Simple Features<sup>19</sup> gemäss der 2D und 3D Spezifikation des Open GIS Consortiums (OGC)
- Import vieler GIS-Formate (mittels OGR<sup>20</sup> Library)
- Räumliche Indizierung von GIS-Objekten
- Räumliche Abfragen
- Gängige GIS-Analysefunktionen, z.B.: Verschneidungen, Puffer berechnen, Flächen berechnen, Längen auf dem Ellipsoid berechnen, etc.
- Folgende Funktionalitäten sollen in Zukunft erscheinen:
  - Verbesserte Technologien für das Laden und Entfernen von Daten
  - Benutzerinterface Werkzeuge für den direkten Datenzugang und -manipulation
  - Support für Topologien, Netzwerke und Oberflächen [NETLEXIKON] [POSTGRESQL 2005]

#### 4.2.4.2 Aufbau der Datenbank

Die geographische Datenbank ist in einfachen Tabellen mit flacher Hierarchie aufgebaut. Pro Kartenebene existiert mindestens eine Tabelle. Falls für ein Thema mehrere Masstäbe vorhanden sind, kann es durch zwei oder mehrere Tabellen repräsentiert sein. Ein Beispiel dafür wäre das Flusssystem, das in zwei verschiedenen Auflösungen vorliegt. Relationen gibt es lediglich zu zwei Tabellen: eine SRID verknüpft zu einer Tabelle mit Projektionsinformationen (`spatial_ref_sys`) und der Tabellename verknüpft zu einer Tabelle mit Informationen zu den räumlichen Spalten (`geometry_columns`) in der u.a. die Geometrietypen festgelegt sind. Schliesslich sorgen Sequenzen für die Vergabe von eindeutigen Objektidentifikatoren. Nachfolgend eine Liste aller Tabellen:

List of relations			
Schema	Name	Type	Owner
public	buildings	table	williams
public	bus_stops	table	williams
public	contours_high	table	williams
public	contours_low	table	williams
public	elevation	table	williams
public	geology	table	williams
public	geology_gen	table	williams
public	geometry_columns	table	williams

<sup>19</sup> Geometrietypen: Polygon, Punkte, Linien, etc.

<sup>20</sup> Eine Opensource Simple Features Bibliothek, um verschiedene Vektordatenformate lesen und ich manchen Fällen schreiben zu können. (<http://www.remotesensing.org/gdal/ogr/>)



---

```

public | hydro_area      | table | williams
public | hydro_points        | table | williams
public | lakes                | table | williams
public | lakes_high           | table | williams
public | names                | table | williams
public | parkboundry          | table | williams
public | quad_100k            | table | williams
public | quad_24k             | table | williams
public | quad_250k            | table | williams
public | rivers               | table | williams
public | rivers_high          | table | williams
public | roads                | table | williams
public | ski_lifts            | table | williams
public | spatial_ref_sys      | table | williams
public | trails               | table | williams
public | vegetation            | table | williams
public | vegetation_har       | table | williams

```

#### 4.2.4.3 Import geographischer Daten

Die in ESRI ArcGIS fertig bearbeiteten Daten werden aus der ArcGIS Datenbank als Shapefile exportiert und in die PostGIS Datenbank importiert.

```
shp2pgsql -s 26911 ski_lift ski_lift >ski_lift.sql
```

Hier wird ein temporäres SQL-File generiert, das zuerst eine neue räumliche Tabelle kreiert, um die Daten danach mit SQL INSERT Befehlen einfügen zu können. `-s` kreiert und füllt Tabellen mit der angegebenen `SRID`<sup>21</sup>, hier `26911` (UTM Zone 11, NAD83). Dann wird der Name des Shapefiles eingegeben, der Name der Tabelle und der Name des temporären SQL-Files (Batch File). Am Ende der Datei wird noch ein Primärschlüssel erstellt sowie eine Sequenz zur Erstellung eindeutiger Objektidentifikatoren.

```
psql -d yosemite -f ski_lift.sql
```

Die in der Datei `ski_lift.sql` (`-f` für lesen aus einem File) enthaltenen SQL Befehle werden nun mit dem Befehl `psql` ausgeführt. Die Daten werden in die Datenbank (`-d`) `yosemite` eingefügt. Nachfolgend eine Beispiel SQL-Datei der Yosemite Skilifte (reduziert auf zwei exemplarische Lifte):

```
BEGIN;
```

---

21 SRID: Spatial Reference Identification, vom European Petroleum Survey Group definiert

```

CREATE TABLE "ski_lift" (gid serial, "shape_leng" numeric, "name"
varchar);
SELECT AddGeometryColumn('', 'ski_lift', 'the_geom', '26911',
'MULTILINESTRING', 2);
INSERT INTO "ski_lift" (gid, "shape_leng", "name", "the_geom") VALUES
('0', '3.03972997267e+002', 'Badger Pass Ski Lift', GeometryFromText
('MULTILINESTRING ((264646.272192171 4171520.59139499 ,
264930.158728917 4171629.25609635 ))', 26911) );
INSERT INTO "ski_lift" (gid, "shape_leng", "name", "the_geom") VALUES
('1', '2.54413791007e+002', 'Badger Pass Ski Lift', GeometryFromText
('MULTILINESTRING ((264723.695796334 4171350.80281244 ,
264858.168342047 4171566.77390195 ))', 26911) );
ALTER TABLE ONLY "ski_lift" ADD CONSTRAINT "ski_lift_pkey" PRIMARY
KEY (gid);
SELECT setval ('ski_lift_gid_seq', 3, true);
END;

```

Anschliessend werden räumliche Indizes in der Datenbank erstellt, damit die Abfrage bei grossen Datenmengen beschleunigt und optimiert wird. Beim Suchen werden die Daten mit Hilfe der Indizes in einer Art Nachschlageindex durchsucht und ausgelesen. Das Geometriefeld ist die Geometriespalte in der Datenbank und heisst per Default `the_geom`.

```

CREATE INDEX ski_lifts_the_geom ON public.ski_lifts USING gist
(the_geom);

```

Bevor die Daten extrahiert werden können, müssen noch die Zugriffsrechte der Tabelle gesetzt werden, damit der Webbenutzer die Daten selektieren darf. Er bekommt “select-only” Rechte, damit er die Daten nicht manipulieren kann. Zusätzlich zu den Geometrietabellen muss der Nutzer Zugriff auf die Tabellen “`spatial_ref_sys`” (Tabelle mit Definitionen der Projektionssysteme) und “`geometry_columns`” (Tabelle mit Informationen über Geometriespalten) haben.

#### 4.2.5 PHP: Serverseitige Programmiersprache

PHP steht für “Personal Home Page” oder “Hypertext Preprocessor” und ist eine Programmiersprache, die auf drei Arten verwendet werden kann: als serverseitige Programmiersprache, als Command-line<sup>22</sup> Programmiersprache oder für Client-side GUI (Graphic User Interface/Graphische Benutzeroberfläche) Applikationen. PHP wurde ursprünglich entwickelt, um dynamische Webinhalte kreieren zu können. Inzwischen wird PHP gerne verwendet, um XML-Dokumente, Graphiken, Flash-Animationen, PDF Dateien usw. herzustellen. [LERDORF et al. 2002]

In meiner Applikation verwende ich PHP um Daten aus der Datenbank extrahieren zu können. PHP unterstützt mehrere Datenbanken u.a. MySQL, PostgreSQL und Oracle.

<sup>22</sup> Mit command-line ist gemeint, dass mit Hilfe der Console Befehle gegeben werden.

#### 4.2.6 Extrahieren/Generieren von SVG Daten

Um die Daten, die in der Datenbank vorhanden sind, extrahieren und in SVG-Fragmente konvertieren zu können, muss ein serverseitiges Script erstellt werden. In dieser Applikation wird PHP für diesen Zweck verwendet. Im Folgenden eine Illustration wie die Anfrage und Extraktion von SVG-Daten erfolgt:

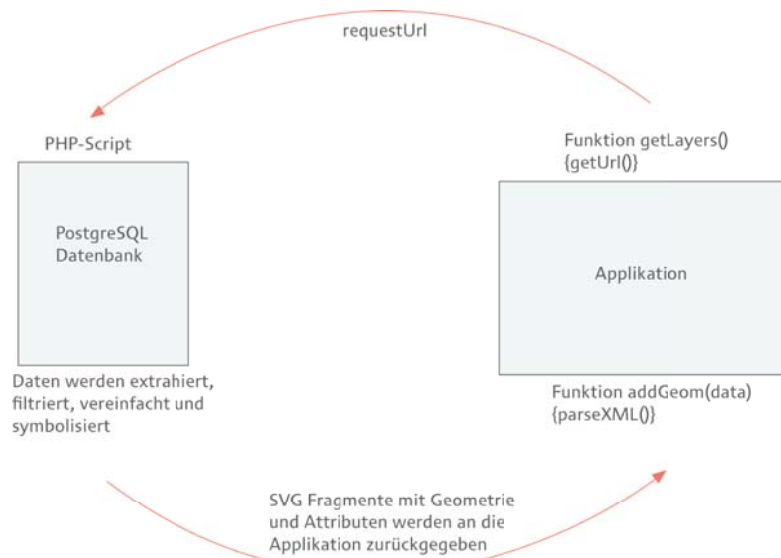


Illustration 32: Anfrage und Extraktion von SVG-Daten

Hier ein Beispiel, wie die Wanderwege extrahiert und SVG Geometrien daraus generiert werden. Es folgt Programmcode mit Erläuterungen. Der Code ist ein Ausschnitt aus der Datei "sendGeom.php", die für die Extraktion von SVG Daten aus der Datenbank zuständig ist.

```

if ($layername == "trails") {
    $tablename = "trails";
    print "<g id=\"".$myId."\" fill=\"none\" stroke=\"crimson\"
        stroke-linecap=\"round\" stroke-width=\"".$width * 0.002."\"
        xmlns:attrib=\"http://www.carto.net/attrib\"
        attrib:timestamp=\"".$timestamp.">\n";
  
```

Hier wird die Tabelle `trails` aus der Datenbank abgerufen. Danach werden die Eigenschaften der Pfade vergeben. Die Pfade haben keine Füllung (`fill="none"`), bekommen die Strichfarbe `crimson` (`stroke="crimson"`) zugewiesen, die Enden der Pfade werden abgerundet (`stroke-linecap="round"`) und die Breite der Pfade wird abhängig von der momentanen Breite des Kartenausschnitts gemacht (`stroke-width="".$width*0.002.""`). Das Attribut `timestamp="".$timestamp.">\n"` sorgt dafür, dass die Daten nur dann in die Kartenapplikation eingefügt werden, wenn sie noch benötigt werden, d.h. in der Zwischenzeit vom gleichen Layer nicht bereits neue Ausschnitte angefordert wurden. Dieser Mechanismus ist wichtig, da der Benutzer bei schnellem Zoomen oder Pannen den Kartenausschnitt schneller wechselt als die Da-

ten vom Server erhalten werden können. Es kann daher vorkommen, dass die eintreffenden Daten zum Zeitpunkt des Erhalts bereits nicht mehr benötigt werden. Falls der Zeitstempel veraltet ist, werden die eintreffenden Daten einfach ignoriert und nicht in die Kartenapplikation eingefügt.

```
$numRecs = 0;
if ($width <= 12000) {
    $addWhereClause = ' AND (class = "1" OR class = "2")';
}
```

Falls die Kartenbreite kleiner oder gleich 12000m ist, werden nur die Wanderwege mit den Klassen 1 oder 2 geladen. Diese Klassen wurden bereits bei der Datenaufbereitung in ESRI ArcGIS festgelegt. Die Variable `$addWhereClause` verwaltet Filter welche im "WHERE" Teil der finalen SQL Anweisung zum Zuge kommen.

```
$mySQL = 'SELECT id, name, type, AsSVG(the_geom,1,1) AS svg_geom
FROM '.$tablename.' WHERE the_geom && GeometryFromText("BOX3D
('.$xmin.' '.$ymin.', '.$xmax.' '.$ymax.')"::box3d,26911)'
.$addWhereClause.'ORDER BY class DESC';
$my_result_set = pg_Exec($my_pg_connect, $mySQL) or die
(pg_ErrorMessage());
```

Hier erfolgt die SQL-Abfrage. Es werden die Spalten `id`, `name` und `type` selektiert. Diese Spalten wurden bereits in ArcGIS bearbeitet. Die Boundingbox (BOX3D) des Kartenausschnitts wird aus den Variablen `$xmin` `$ymin`, `$xmax` `$ymax` und der SRID 26911 generiert. `&&` ist der "Overlap" Operator. Hier werden alle Geometrien selektiert die mit der angegebenen Boundingbox überlappen. Mit `AsSVG (the_geom,1,0) AS svg_geom` wird die Geometriespalte in SVG umgewandelt. Die Koordinaten sind relativ (1) und es gibt keine Kommastellen (0). `AS svg_geom` gibt den Aliasnamen des Resultats von `AsSVG` an. Es werden nur die d-Attribute<sup>23</sup> des SVG Pfad-Elements erzeugt. Für die korrekte Generierung des vollständigen Pfad-Elements ist PHP verantwortlich. Alle anderen Attribute müssen nicht in SVG umgewandelt werden. `$addWhereClause."ORDER BY class DESC"` gibt an, dass nur die Klassen, die in der Auswahl getroffen wurden, extrahiert werden und in absteigender Reihenfolge nach dem Klassenattribut sortiert werden. `pg_Exec` führt den SQL Befehl aus. Das Resultat der Abfrage wird in der Variable `$my_result_set` gespeichert. Die Datenstruktur des Resultats ist ein Feld und darin verschachtelt ein assoziatives Feld, bei dem der Feldindex vom Typ `String` und nicht vom Typ `Integer` ist.

```
$numRecs = pg_NumRows($my_result_set);
$i = 0;
while ($i < $numRecs) {
    $resultArray = pg_Fetch_Array($my_result_set, $i);
    $id = $layername.'_'. $resultArray[id];
    $mySvgString = $resultArray[svg_geom];
    print '<path id="'.$id.'" attrib:name="'.$resultArray[name].'"
```

23 d-Attribute beschreiben mit SVG gezeichneten <path>- Elemente (Linien oder Kurven)

```

        attrib:type="'.'.$resultArray[type].'" d="'.'.$mySvgString.'" />'
        ."\n";
    $i++;
}

```

Der Code in der oben aufgeführten `while` Schleife wird über alle Records des Resultat-Feldes (`$my_result_set`) ausgeführt. Das Abbruchkriterium der Schleife ist die Anzahl der Records im Ergebnis. Die Zählervariable `i` wird bei jedem Schleifendurchlauf inkrementiert. Innerhalb der Schleife wird ein einzelner Record mit dem Befehl `pg_Fetch_Array()` aus dem Result-Set extrahiert und in der Variable `$resultArray` abgelegt. Einzelne Werte aus dem Record werden mit eckiger Klammer und dem Index-String (in diesem Fall der Spaltenname) ausgelesen. Der `print` Befehl gibt ein einzelnes SVG Pfad-Element mit den Attributen und der Pfadgeometrie (d-Attribut) aus.

Sind die Datenmengen, wie z.B. bei den Ebenen für Vegetation oder Höhenlinien sehr gross, werden sie vereinfacht und auf die Grösse des Kartenausschnittes ausgeschnitten. Dadurch werden v.a. bei langsameren Netzwerkverbindungen die Ladezeiten verkürzt und die vom Client zu verarbeitenden Datenmengen reduziert. Es ist jedoch zu beachten, dass die Datenreduktions- und Clipping-Funktionen am Server zu einer Mehrbelastung des Prozessors führen.

```

SELECT elevation_ft, AsSVG(Simplify(Intersection
    (the_geom,$intersectPolygon),$simplifyTolerance),1,0) ...

```

In diesem Auszug einer SQL Anweisung werden die Daten für die Höhenlinien vereinfacht und geclippt. Die Variable `$intersectPolygon` beinhaltet ein Polygon, das verwendet wird, um die Daten auszuschneiden. Nur der Ausschnitt der in der Karte sichtbar ist, wird geladen. Die Variable `$simplifyTolerance` definiert eine Toleranz für die Generalisierungsfunktion (`Simplify()`), eine Implementation des Douglas-Peucker Generalisierungs-Filters [DOUGLAS et al. 1973]. Für diese Applikation wird der Filter nicht zum Generalisieren sondern zur Datenreduktion verwendet. Daher wird die Filtertoleranz nur sehr konservativ eingestellt, in der Regel kleiner als die Strichbreite der Geometrien.

#### 4.2.7 Higrd: ein Programm zur Extraktion von Geländemolldaten

Um das DTM (Digital Terrain Model, Geländemodell) erstellen zu können, wird ein zusätzliches CGI<sup>24</sup> (Common Gateway Interface) Programm verwendet. Diese Daten werden für die Anzeige der Höhenwerte und die Berechnung des Höhenprofils benötigt. Für diesen Zweck wurde das von Dr. Hans-Ruedi Bär<sup>25</sup> entwickelte Programm Higrd verwendet. Higrd ist die Kurzform für "Hierarchical Grid" und stellt ein Programm zur

24 CGI ist die Abkürzung für "Common Gateway Interface". CGI ist eine Schnittstelle zwischen Webserver und anderen Applikationen. Parameter an die Applikation werden mittels der Methode "Get" (Daten werden im URL codiert) oder "Post" (Daten werden über Standardinput übergeben) übermittelt. Bei der Methode "Get" kann es bei Browsern oder Servern zu Längenlimitierungen bei den zu übergebenden Parametern kommen. Bei grossen zu übertragenden Datenmengen ist "Post" vorzuziehen.

25 Mitarbeiter am Institut für Kartographie, ETH Zürich

Erstellung von Pyramidenstrukturen von Rasterdaten zur Verfügung, sowie ein Programm zum Extrahieren von Rasterdaten in beliebigen Auflösungen. Bei der Extraktion werden die Parameter xmin, xmax, ymin, ymax, die Zellgrösse und das Outputformat verlangt. In meinem Fall wurde das XML-Format als Ausgabeformat gewählt. Jede Zeile im Geländemodell ist dabei in einem XML-Element abgelegt, die einzelnen Zellen werden durch Trennzeichen abgetrennt. Nach Erhalt der Daten liest ein Javascript die Zeilen ein und erstellt ein zweidimensionales Feld daraus das die Höhenwerte enthält. Durch bilineare Interpolation kann aus diesem Feld zu jeder beliebigen Koordinate der Höhenwert ausgerechnet werden. Nachfolgend ein Beispiel für einen URL der Geländemodelldaten von Higrad anfordert:

```
http://www.carto.net/cgi-williams/higrad.cgi?dsname=yos.bin&
xmin=244000&ymin=4150000&xmax=308000&ymax=4231000&resolution=240&
format=text/svg+xml&get_data=Daten+liefere
```

Im Anschluss wird ein Codefragment gezeigt wie es von Higrad zurückgeliefert wird:

```
<g id="dtm" xmlns:dtm="http://www.karto.ethz.ch/dtm">
<dtm:metadata west="244000" south="4150120" east="246160"
north="4152040" nrows="8" ncols="9" resolution="240"/>
<dtm:row id="dtmRow0">734 746 756 745 742 728 700 701 733</dtm:row>
<dtm:row id="dtmRow1">721 729 732 734 732 715 696 685 729</dtm:row>
<dtm:row id="dtmRow2">732 713 717 721 715 704 687 711 715</dtm:row>
<dtm:row id="dtmRow3">729 720 710 704 702 684 682 697 706</dtm:row>
<dtm:row id="dtmRow4">712 693 701 703 684 674 673 687 692</dtm:row>
<dtm:row id="dtmRow5">710 685 692 694 683 669 676 677 683</dtm:row>
<dtm:row id="dtmRow6">702 685 684 696 678 666 658 669 678</dtm:row>
<dtm:row id="dtmRow7">690 677 669 681 682 678 683 663 664</dtm:row>
</g>
```

#### 4.2.8 Open GIS Consortium<sup>26</sup> WMS (Web Map Service)

Nach ISO/DIS 19128 produziert ein Web Map Service (WMS) Karten aus räumlich referenzierten Daten. Dieser Standard definiert eine Karte als eine Abbildung geographischer Informationen in digitaler Form, die gut auf einem Computerbildschirm angezeigt werden kann. Karten, die durch ein WMS erstellt wurden, werden meistens in einem Rasterformat wie z.B. PNG, GIF oder JPEG ausgeliefert. Seltener erscheinen sie als SVG (Vektorformat) oder WebCGM (Web Computer Graphics Metafile, Vektorformat).

Der Standard definiert weiterhin drei mögliche Abläufe:

- die Rückgabe von Metadaten über den Lieferbereitschaftsgrad (service-level) und die zur Verfügung stehenden Daten
- die Rückgabe einer Karte, deren geographische und dimensionale Parameter gut definiert

<sup>26</sup> OGC: gemeinnützige Organisation, die die Entwicklung von raumbezogener Informationsverarbeitung (v.a. Geodaten) als Basis allgemeingültiger Standards zum Zweck der Interoperabilität festlegt.



sind

- die Rückgabe von Informationen über bestimmte Eigenschaften, die auf der Karte gezeigt werden

Diese Abläufe können jeder Zeit durch ein Webbrowser abgerufen werden, indem URLs benutzt werden. Der Inhalt einer solchen URL ist abhängig davon welcher Ablauf aufgerufen wird und welche Informationen auf der Karte gezeigt werden sollen. Werden zwei oder mehrere Karten mit den gleichen geographischen Parametern und Outputgrößen generiert, können die Ergebnisse übereinander gelegt werden, um eine zusammengesetzte Karte zu kreieren. Ausserdem können individuelle Karten von verschiedenen Servern abgefragt werden. Dadurch erlaubt ein WMS die Bereitstellung eines Netzwerks von verteilten Map Servern, von denen der Benutzer massgeschneiderte Karten zusammensetzen kann.

Ein einfacher WMS klassifiziert seine geographischen Informationen in Ebenen (*Layers*) und offeriert eine Anzahl von vordefinierten Stilen (*Styles*), in denen die Ebenen angezeigt werden können. [OGC-ISO 19128 2004]

Die Integration von Rasterdaten, in diesem Fall die Orthophotos und die Reliefschattierung, erfolgt über einen Map Server. Die Orthophotos werden mit dem momentan freien Terraserver WMS in die Applikation eingebunden. Terraserver bietet USGS Orthophotos (DOQ: Digital Orthoimage Quadrangles) und gescannte topographische USGS Karten (DRG: Digital Raster Graphics) an. Die Daten liegen in verschiedenen Massstäben vor. Um Orthophotos vom Terraserver zu erhalten, muss man einen URL generieren, der wie folgt aussieht:

```
http://terrasservice.net/ogcmap.ashx?version=1.1.1&request=GetMap
&Layers=doq&Styles=&SRS=EPSG:26911
&BBOX=254028,4201714,257359,4205930&width=553&height=700
&format=image/jpeg
```

`http://terrasservice.net/ogcmap.ashx?version=1.1.1` Nennt den Servername, den Servicename und die Version

`request=GetMap&Layers=doq` Hier werden die Art der Abfrage (*GetMap*) und die gewünschte Ebene (*doq*), festgelegt.

`Styles=&SRS=EPSG:26911&BBOX=254028,4201714,257359,4205930` Hier wird die EPSG<sup>27</sup> = SRID Nummer 26911 angegeben, die bedeutet, dass die Daten in UTM Zone 11 mit dem Datum NAD83 und dem Spheroid GRS80 angefordert werden. Die Boundingbox gibt die Min- und Max Koordinaten von x und y an. Mit den oben genannten Koordinaten würde das westliche Ufer des Hetch Hetchy Reservoirs dargestellt werden.

27 European Petroleum Survey Group: eine Gruppe von Vermessern, Geodäten und Kartographen von Ölfirmen in Europa. Das Ziel ist es die Vermessung zu verbessern, effizienter zu machen, die Qualität und die Sicherheit zu erhöhen und die Umwelt zu schützen.



Illustration 33: Beispiel eines Terraserver Orthophotos:  
westliches Ufer des Hetch Hetchy Reservoirs

`width=553&height=700` Dies ist die Höhe und Breite des resultierenden Bildes in Pixel.

`format=image/jpeg` Das Bild wird im jpeg Format ausgegeben.

Als nächstes muss der WMS-Service in die SVG-Applikation eingebettet werden. Es wird ein ECMAScript Code hinzugefügt, der eine Referenz auf die Gruppe im SVG erstellt, die das Orthophoto enthalten soll. Als nächstes wird überprüft, ob ein Bild schon geladen ist. Ist dies der Fall, wird das alte Bild entfernt und durch das neue ersetzt. Dabei wird ein neuer URL nach dem oben erklärten Schema generiert, der den neuen Kartenausschnitt beschreibt.

Im Gegensatz zu den Orthophotos, die direkt von einem anderen WMS Server geladen werden, musste für die Reliefdaten ein eigener WMS eingerichtet werden. Der dafür verwendete Map Server ist der UMN (University of Minnesota) MapServer, der als OpenSource<sup>28</sup> Software erhältlich ist. Der UMN Map Server ist einer der populärsten Kartenserver weltweit. Er kann verschiedene GIS-Daten (Vektor- und Raster) lesen, symbolisieren und in verschiedenen Webformaten (derzeit v.a. Rasterformate) ausgeben. Er unterstützt Lese-Verbindungen zu geographischen Datenbanken (PostGIS), kann Daten umprojizieren und mittels MapScript auch serverseitige Interaktionen ausführen. Programmierschnittstellen existieren zu allen populären Programmier- und

<sup>28</sup> Open-Source: Der Quelltext eines Programms ist frei erhältlich; ein Werk, z.B. Software steht zur freien Verfügung und der Quelltext darf modifiziert werden

Scriptingsprachen.

Um Kartenebenen über den UMN MapServer anzeigen zu können, muss ein Mapfile konfiguriert werden, das alle relevanten Metadaten und Ebeneninformationen enthält. Die Reliefs sollten als \*.tif mit ihren dazugehörigen \*.wld (Worldfiles) abgespeichert sein. \*.wld-Files beinhalten die Informationen zur Georeferenzierung. Ohne diese Information, kann der MapServer die Bilder nicht im UTM Koordinatensystem ausliefern.

Das Mapfile ist wie folgt aufgebaut:

MAP

    OUTPUTFORMAT

    PROJECTION

    WEB

    LAYER

Im OUTPUTFORMAT werden das Format, der Farbmodus (RGB, Indexed Color) und die Qualität des Bildes festgelegt. Die PROJECTION definiert die Projektion (UTM Zone 11, NAD83). Im Unterverzeichnis METADATA in WEB wird neben Titel und Projektion der URL zum WMS festgelegt:

```
http://www.carto.net/cgi-williams/mapserv?  
map=/home/williams/www/yosemite/mapserver/hillshade.map&SERVICE=WMS  
&VERSION=1.1.1&REQUEST=GetMap&LAYERS=hillshade&STYLES=&  
SRS=EPSG:26911&BBOX=244000,4150000,308000,4231000&WIDTH=553&  
HEIGHT=700&FORMAT=image/jpeg
```

<http://www.carto.net/cgi-williams/mapserv?> Hier ist der Servername mit dem CGI-Directory und dem CGI-Applikationsnamen

[map=/home/williams/www/yosemite/mapserver/hillshade.map](#) Hier der absolute Pfad zum Mapfile. Die anderen Bestandteile des URL wurden oben erklärt.

LAYER gibt Informationen zu den Ebenen an. Hier wird der Name der Ebene angegeben und zu welcher Gruppe diese gehört. Alle Ebenen, die die gleiche Gruppe ([GROUP hillshade](#)) besitzen, werden also zu einer gemeinsamen neuen Ebene zusammengefasst. Die [MINSCALE](#) und [MAXSCALE](#) Parameter bestimmen welche Ebene in Abhängigkeit vom Massstab verwendet wird. Diese Werte müssen bestimmt werden, wenn eine Gruppe definiert wird. METADATA und PROJECTION werden so behandelt wie oben erklärt. Nachfolgend ein Ausschnitt aus dem Mapfile. Hier wird die zweite Ebene der drei Auflösungsstufen des Reliefs definiert.

```
LAYER  
    NAME "hillshade_middle"
```

```
GROUP "hillshade"
DATA "relief_detailed.tif"
TYPE RASTER
STATUS ON
# display layer for scale values: 50000 < scaleValue < 100000
MINSCALE 50000
MAXSCALE 100000

METADATA
  WMS_TITLE "Yosemite Shaded Relief Analytical, Normal"
  WMS_ABSTRACT "Shaded Relief of the Yosemite National Park,
    Analytical Hillshading, normal resolution"
  WMS_SRS "EPSG:26911"
END
PROJECTION
  "init=epsg:26911"
END
END
```

## 4.3 Kartengestaltung

### 4.3.1 Massstabsabhängige Symbolisierung

Zur korrekten massstabsabhängigen Symbolisierung müssen bestimmte Regeln beachtet werden. Die Symbolisierung geht Hand in Hand mit der Generalisierung und hat das Ziel ein lesbareres und aufschlussgebenderes Kartenbild zu erzeugen. Die automatische und halbautomatische Generalisierung ist ein aktuelles Forschungsgebiet in der Kartographie, das sich immer noch in der Entwicklungsphase befindet. Es ist schwierig Generalisierungsregeln zu formalisieren zumal nicht nur einzelne Elemente, sondern die gesamte Kartengraphik im Kontext betrachtet werden muss. Folgende Teilschritte sollten in der Regel bei der Generalisierung und massstabsabhängigen Symbolisierung beachtet werden:

- Auswahl von Elementen
- Zusammenfassen von Elementen
- Vereinfachung von Geometrie
- Verdrängung von Elementen
- Vergrössern und Betonen von Elementen (Beachtung der Minimaldimensionen)
- Zuweisung von Strichstärken, Symbolgrössen und Schriftgrössen
- Überführung von Geometrien zu Signaturen beim Massstabswechsel
- Klassifizieren, Typisieren

[HAKE/GRÜNREICH/MENG 2002] unterscheiden zwischen geometrischer Generalisierung und sachlicher Generalisierung mit geometrischer Wirkung. Zusätzlich kann es noch eine temporale Generalisierung geben. Zahlreiche Beispiele zur Generalisierungspraxis befinden sich in "Topographische Karten- Kartengraphik und Generalisierung", von der Schweizerischen Gesellschaft für Kartographie ausgegeben [SGK 1990]. Eine gute Zusammenfassung des Generalisierungsprozesses befindet sich unter "Der Generalisierungsbegriff in der Kartographie und anderen Disziplinen" [BRASSEL 1988].

Auf Grund der Komplexität des Generalisierungsprozesses sowie des Zusammenspiels verschiedener Generalisierungsoperatoren kommen in diesem Beispiel nur ausgewählte Regeln zum Einsatz. Diese sind: Auswahl von Elementen, Vereinfachung von Geometrie, Klassifizieren und Typisieren und Zuweisung von Strichstärken, Symbolgrößen und Schriftgrößen. Ebenso werden Kartensignaturen verwendet. Falls vorhanden, kommen auch unterschiedliche Massstäbe bei den einzelnen Kartenebenen zum Einsatz. So liegt etwa das Gewässernetz in zwei verschiedenen Generalisierungsstufen vor. Beim Überschreiten eines definierten Schwellwertes wird auf den jeweils anderen Massstabsbereich gewechselt.

Die massstababhängige Zuweisung der Symbole erfolgt schon serverseitig bei den Datenbankabfragen und der Generierung des SVG-Codes.

```
if ($resultArray[levels] == 15) {
    $fill= ' fill="aliceblue" stroke="cornflowerblue" stroke
    width="'. $width * 0.001.'";
}
```

Hier wird einem Pfad, der einen See mit der Klasse 15 darstellt, die Füllfarbe `fill="aliceblue"` zugewiesen. Die Linie hat die Strichfarbe `stroke="cornflowerblue"`. Die Strichstärke ist abhängig von der momentanen Kartenbreite (Zoomstufe), hier ein Tausendstel davon `width="'. $width * 0.001.'"`.

Im folgenden Beispiel wird ein Symbol auf die gewünschte Grösse, wieder abhängig von der Zoomstufe, skaliert.

```
$i = 0;
while ($i < $numRecs) {
    $resultArray = pg_fetch_array($my_result_set, $i);
    $mySvgString = str_replace('x=', 'transform="translate
    (' , $resultArray[svg_geom]);
    $mySvgString = str_replace('" y=', ' ', $mySvgString)
    $mySvgString = rtrim($mySvgString, '"').', rotate(''. $resultArray
    [angle].'), scale(''. $width * 0.002.'')";
```

Es werden die X und Y Koordinaten aus einem bestehenden Transformstatement eingelesen `'transform="translate` und in der letzten Zeile mit einem Rotations- `rotate` (`$.resultArray [angle]."`) und einem Skalierungsfaktor `scale("$.width * 0.002.")` versehen.

Beim Setzen der Werte wurde wenn möglich auf die gängigen kartographischen Richtlinien für Punktgrößen, Linienstärken und Flächengrößen geachtet. Dabei erscheint z.B. das Symbol für die Bushaltestellen erst ab einer bestimmten Zoomstufe, oder je weiter in der Karte eingezoomt wird, desto detaillierter ist das Namensgut. Die Stufen der Genauigkeit wurden bereits bei der Bearbeitung der Daten in ESRI ArcGIS festgelegt. Die Schwellwerte und Symbolisierung wird im PHP-Script geregelt und eine Ordnung und Auswahl wird mit SQL getroffen. Diese Auswahl erscheint massstabsabhängig in der Karte. Geometrische Vereinfachung von Linien und Polygonelementen erfolgt mit der bereits oben beschriebenen PostGIS Funktion `Simplify()`.

### 4.3.2 Textplatzierung

Die Platzierung des Textes folgt automatisch. Die Tabelle mit den geographischen Namen wird mit dem PHP-Script aus der Datenbank gelesen und anhand von Punktkoordinaten platziert. Es gibt in dieser Applikation zwei verschiedene Beschriftungstypen: Punktbasierte (z.B. Gipfel- oder Passnamen) und Flächenbasierte (z.B. Flurnamen und andere Geländebezeichnungen).

```
while ($i < $numRecs) {
    $resultArray = pg_Fetch_Array($my_result_set, $i);
    $id = $tablename.'_'. $resultArray[gid];
    $mySvgString = str_replace('x="','transform="
        translate(','.$resultArray[svg_geom]);
    $mySvgString = str_replace('" y="',' ',$mySvgString)
    $mySvgString = rtrim($mySvgString,'"').'),rotate
        ('.$resultArray[angle].'),scale('.$width * 0.0014.'))";

    if ($resultArray['class'] == 8) {
        $symbolType = 'pass';
    }
    else {
        $symbolType = 'names';
    }
}
```

Das Symbol wird mit Hilfe von `x` und `y` an der Punktkoordinate platziert. Mit `scale` ändert sich die Grösse des Symbols in Abhängigkeit von der Breite (`.$width * 0.0014.`) der Karte. Hat das jeweilige Symbol die Klasse “8” (ein Pass), wird das in der SVG Symbolliste abgelegte Symbol `pass` verwendet. Die Drehrichtung (`angle`) des Symbols, entsprechend der Richtung des Wanderweges, wurde in ArcGIS festgelegt und wird an dieser Stelle des PHP-Scripts abgefragt. Das Symbol wird anhand der Drehrichtung in der Karte platziert. Gehört die Beschriftung einer der anderen Gruppen an, wird das Symbol `names` (unsichtbares Hilfssymbol) hinzugezogen und platziert. Da es sich um ein Flächensymbol handelt, das um einen Punkt zentriert wird, ist hier keine Drehung notwendig.

Um die Beschriftung von der darunterliegenden Kartengraphik hervorheben zu können, musste eine weisse Umrandung um die Buchstaben gelegt werden. Der Adobe SVG-



Viewer hat einen “Fehler” und kann die übliche Art Texte hervorzuheben nicht durchführen. Normalerweise wird der Text referenziert und bekommt eine Strichstärke zugewiesen, mit einer weissen Strichfarbe. Der referenzierte Text wird unter das ursprüngliche Textelement gelegt. Diese Methode kommt beim Apache SVG-Viewer zum Einsatz. Beim Adobe SVG-Viewer wird stattdessen ein SVG Filter verwendet. Der Filter wird global in den `<defs>` definiert.

```
<filter id="textshadow">
  <feMorphology id="morphDilate" operator="dilate"
    in="SourceGraphic" radius="1.2" result="dila"/>
  <feMerge>
    <feMergeNode in="dila"/>
    <feMergeNode in="SourceAlpha"/>
  </feMerge>
</filter>
```

`feMorphology` mit dem `operator="morphDilate"` erweitert die Ränder von graphischen Elementen. Die Referenz `in="SourceGraphic"` verweist auf die Graphik, auf die der Filter angewendet werden soll. Mit einem Radius grösser als eins, legt man fest, dass der Filter um die Graphik, in diesem Fall mit 1.2, erweitert wird. Das temporäre Resultat wird unter dem Namen `result="dila"` zwischengespeichert. Schliesslich wird das Originalbild mit `<feMerge>` mit dem zwischengespeicherten Filterresultat kombiniert. Mit `SourceAlpha` wird nur der Alphakanal der Eingangsgraphik referenziert (der Alphakanal ist per default auf 1 gesetzt), was zur Folge hat, dass die Originalschrift schlussendlich schwarz erscheint. Die durch den Filter entstandene Umrandung erscheint gemäss dem Textelement zugewiesenen Füllfarbe weiss. Im Folgenden das ECMAScript Fragment mit der Zuweisung der Füllfarbe und des Filters.

```
label.setAttributeNS(null,"fill","white");
label.setAttributeNS(null,"filter","url(#textshadow)");
```

[BADER 2004]

Damit die Schriften sich nicht überlappen oder aus dem Kartenblatt hinauslaufen, müssen sie mit Hilfe eines Algorithmus so platziert werden, dass sie den kartographischen Regeln weitgehend folgen. Es wird versucht dieses Problem mit einer automatischen Schriftplatzierung anzugehen.

Das Textelement wird zunächst an seiner Defaultposition (oben rechts) platziert. Danach wird mit Hilfe der Boundingbox geprüft, ob die Schrift in den Kartenrand hinausläuft. Ist dies der Fall, wird die Defaultposition durch die nächstbessere Positionierung nach den acht Schriftplatzierungsmöglichkeiten (siehe Graphik unten) ersetzt. Die verschiedenen Optionen werden solange getestet, bis die Beschriftung vollständig innerhalb des Kartenausschnittes platziert werden kann.

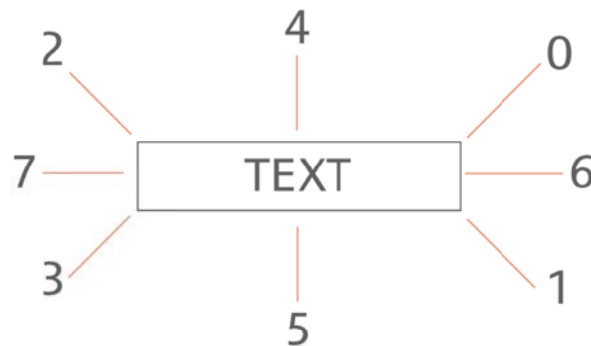


Illustration 34: Kartographische Textplatzierungsreihenfolge

Hier die Parameter der acht möglichen Textpositionen, relativ gesehen zur Punktkoordinate des Symbols:

```
var placementOptions = new Array();
placementOptions[0] = {x:myMainMap.curWidth * 0.007,y:
  (myMainMap.curWidth * 0.007)*-1,textanchor:"start"};
placementOptions[1] = {x:myMainMap.curWidth *
  0.007,y:myMainMap.curWidth * 0.02,textanchor:"start"};
placementOptions[2] = {x:(myMainMap.curWidth * 0.007)*-1,y:
  (myMainMap.curWidth * 0.007)*-1,textanchor:"end"};
placementOptions[3] = {x:(myMainMap.curWidth * 0.007)*-1,
  y:myMainMap.curWidth * 0.02,textanchor:"end"};
placementOptions[4] = {x:0,y:(myMainMap.curWidth * 0.011)*-1,
  textanchor:"middle"};
placementOptions[5] = {x:0,y:myMainMap.curWidth * 0.025,
  textanchor:"middle"};
placementOptions[6] = {x:myMainMap.curWidth * 0.007,
  y:0,textanchor:"start"};
placementOptions[7] = {x:(myMainMap.curWidth * 0.007)*-1,
  y:0,textanchor:"end"};
```

Als nächstes wird geprüft, ob die Schrift das Symbol eines anderen Textes oder ein anderes Textlabel überschneidet. Hier werden wieder die obengenannten kartographischen Regeln zur Bestimmung der nächstbesseren Position angewandt. Zur Überprüfung der Überlappungen wird ein Feld mit verbotenen rechteckigen Flächen geführt, gegen dieses das neu zu platzierende Text getestet wird. In diesem Feld befinden sich sämtliche Flächen, die von den Punktsymbolen eingenommen werden. Kann ein Textelement erfolgreich platziert werden, so wird es ebenfalls in diesem Feld mit aufgenommen. Kann eine Schrift trotz der Verschiebungen nicht platziert werden, ohne dass sie eine andere überlappt, wird sie, als letzte Möglichkeit, weggelassen.

Bei Schriften, die nicht punkt- sondern flächenbezogen sind, wie z.B. Wiesen oder Siedlungen, werden andere Regeln verwendet. Diese Schriften werden per Default zentriert

auf die jeweiligen Punktkoordinaten gesetzt. Sie werden ebenfalls zuerst geprüft, ob sie aus dem Kartenausschnitt hinauslaufen. Wenn dies der Fall ist, wird getestet, ob sie mit einer Verschiebung in X- oder Y-Richtung nicht mehr den Kartenrahmen überschneiden. Die Schrift darf maximal bis zur halben Länge oder Höhe der Textboundingbox verschoben werden. Andernfalls wird dieses Textlabel nicht platziert.

Anschliessend werden sie ebenfalls mit den verbotenen Flächen getestet. Sollten sie mit anderen Elementen kollidieren, wird nochmals eine Verschiebung in X- und/oder Y-Richtung bis zur halben Länge oder Höhe der Textlabelboundingbox versucht. Dabei wird zeitgleich auch getestet, ob der Text durch die neue Position immer noch vollständig im Kartenrahmen zuliegen kommt. Als letzte Option, falls die Schrift nicht platziert werden kann, wird sie weggelassen.

Die Reihenfolge in der die Schriften gesetzt werden, hängt von der Reihenfolge ab wie sie aus der Datenbank extrahiert werden. Schriften die zuerst platziert werden erhalten potenziell bessere Positionen als Schriften die später platziert werden. Deshalb empfiehlt es sich die Schriften bei der Datenbankabfrage bereits nach Wichtigkeit zu sortieren.

Derzeit besitzt der Algorithmus folgende zwei Limitierungen: Es wird kein iteratives Verfahren zur Verbesserung der Resultate verwendet, sondern nur von der Reihenfolge der zu platzierenden Texte ausgegangen. Die Positionierungen im Falle der Punktsymbole werden nur an den acht fixen Positionen getestet. Würden auch leichte Verschiebungen gegenüber den fixen Positionen toleriert werden, so liessen sich wahrscheinlich mehr Textelemente optimaler platzieren. Schliesslich können derzeit auch keine gebogenen Schriften, etwa entlang von Flussläufen, platziert werden.

Der Computer kann nur die Regeln ausführen, die ihm im Algorithmus gegeben werden. Würde man aber die Schriften per Hand setzten, ergäben sich sicherlich mehr Möglichkeiten der Platzierung. In manchen Fällen hätte man durch eine weitere, kleine Verschiebung per Hand eine Schrift mehr oder deutlicher setzen können. Manche Gipfelnamen können nicht eindeutig zugewiesen werden, können aber im Zweifelsfall noch mit der Maus abgefragt werden.

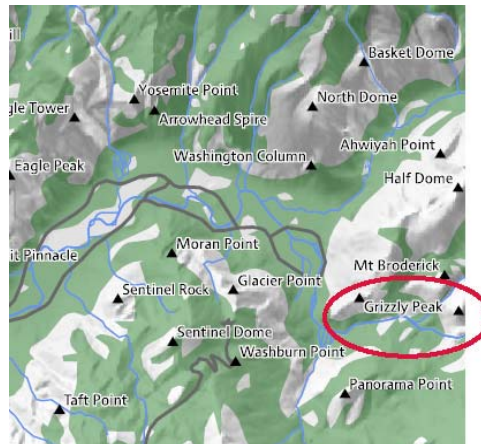


Illustration 35: Nicht eindeutige  
Textplatzierung

Die verwendete Schriftart ist Cisalpin. Diese Schrift wurde von Felix Arnold neu entwickelt und eignet sich speziell für Karten, da sie eine relativ enge Laufweite aufweist und auch in sehr kleinen Grössen gut lesbar ist. Ausserdem sind alle Buchstaben deutlich voneinander zu unterscheiden. Auch wenn die Schrift über komplexe Kartenhintergründe platziert wird, bleibt sie leserlich. [LINOTYPE]

#### 4.3.3 Reliefschattierung

Die Reliefebene besteht aus drei Inputreliefdatensätzen mit verschiedenen Auflösungen, die je nach Zoomfaktor gezeigt werden. Das erste Relief in der ersten Zoomstufe ist ein handgemaltes Relief des National Park Services. Die beiden anderen sind digital aus den Daten des USGS DTM (30 Meter Auflösung) erstellt worden. Bevor sie in die Applikation eingebunden worden sind, mussten sie mit Shadow<sup>29</sup> und Adobe Photoshop bearbeitet werden.

Mit Shadow ist es möglich einzelne Regionen selektiv zu verbessern. Damit können z.B. markante Gebirgszüge hervorgehoben werden, indem die Lichtrichtung in diesen Regionen abgedreht wird. Stellen, die zu dunkel sind oder wenig Kontrast aufweisen, können aufgehellt werden.

Zu verändernde Regionen werden als Vektoren erfasst. Es gibt verschiedene mögliche Algorithmen, die in den durch die Vektoren ausgewählten Rastergebieten ausgeführt werden können:

- Schattierung: Formen können abgedunkelt werden, ohne dass viel Kontrast verloren wird

<sup>29</sup> Shadow: von Bernhard Jenny (Institut für Kartographie) erstelltes Programm zur Bearbeitung von digitalen Reliefschattierungen

- Reduktion von Licht- und Normalvektor auf die Horizontalebene: flache Gebiete können mit einem Ebenenton abgedeckt werden
- Atmosphäre: hier werden atmosphärische Aufhellungen simuliert
- Kontrast: der Kontrast kann angepasst werden, d.h. reduziert oder erhöht
- Ebenenton: der Ebenenton kann an die Umgebung angepasst werden
- Lokale Einstellungen: innerhalb eines Polygons können die Beleuchtung, die Lichtrichtung, die Überhöhung des Geländes, die Atmosphäre, der Kontrast und der Ebenenton verändert werden
- Überlagerung von Rasterdaten: Rasterbilder können über das Relief gelegt werden
- Import und Export von Daten: Höhenmodelle in verschiedenen Formaten können konvertiert und eingelesen werden. Einzelne Ausschnitte können in neuen Dateien abgelegt werden.
- Median-Filter: Artefakte im Höhenmodell können geglättet werden
- Lichtton: hier kann ein Lichtton berechnet werden

[JENNY 2000]

Das Relief des Yosemite konnte mittels Shadow erheblich verbessert werden. Nach der Bearbeitung mit Shadow musste das gesamte Relief aufgehellt werden. Dies geschah mit Hilfe des Programms Adobe Photoshop. Das Relief soll nämlich als dezente Basis dienen und nicht die Karte übertönen



Illustration 36: Digital erstelltes Relief des USGS vor der Bearbeitung



Illustration 37: Nach der Bearbeitung mit Shadow und Photoshop

#### 4.4 Graphische Benutzeroberfläche

Jede vollständige Karte weist neben der Karte auch eine Kartenrandausstattung auf: Kartentitel, Legende, Massstabsangabe, Angabe über die verwendeten Daten und ihre Quellen, Kartenautor.

Um die Einzelteile des Inhalts auseinanderzuhalten, wurde die Bildschirmgestaltung in verschiedene Bereiche gegliedert. Die Karte steht natürlich im Mittelpunkt der Applikation. Die umliegenden Bereiche steuern die Applikation oder geben Informationen aus. Die untenstehende Abbildung soll die Aufteilung verdeutlichen und wird im Nachfolgenden erklärt.

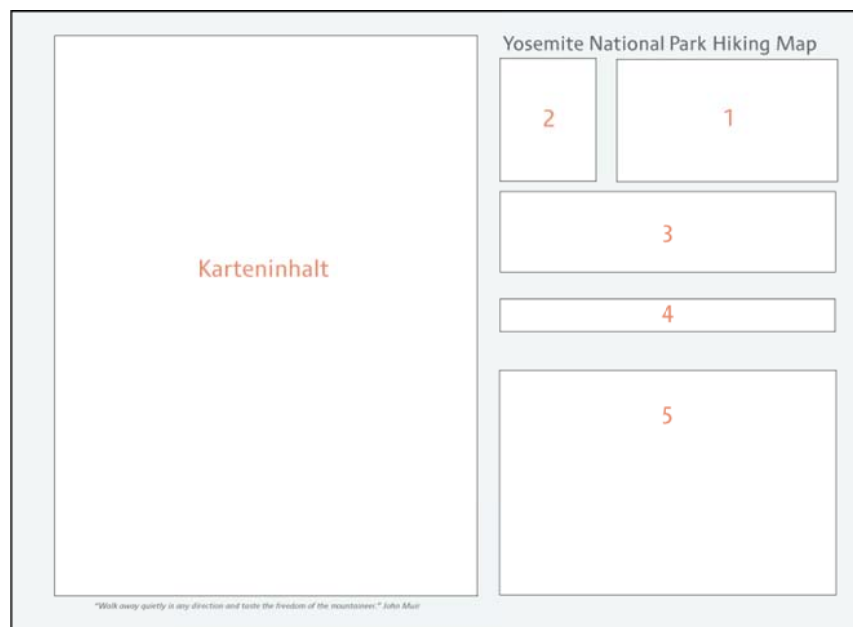


Illustration 38: Schema des Layouts der Applikation

Räumliche Navigation (1)

Referenzkarte (auch ein Teil der Navigation) (2)

Ebenenkontrolle (3)

Attributenanzeige (4)

Thematische Navigation und Information (5)



Die Applikation sollte in Bezug auf die kartographische Gestaltung ein paar grundsätzlichen Regeln folgen. Zuerst sollte die Seite einfach zu lesen sein, d.h. die Schrift darf weder zu gross noch zu klein sein und sie sollte sich klar von der Hintergrundfarbe abheben. Dabei ist zu beachten, dass serifen-lose Schriften am Bildschirm einfacher zu lesen sind als Schriften mit Serifen. Ausserdem sind dünne bzw. fette und kursive Schriften in Bildschirmkarten nicht zu empfehlen, da diese auch schwieriger zum Lesen sind.

Als nächstes ist die Farbwahl von grosser Bedeutung. Nicht nur für die Hintergrundfarbe der Website, sondern auch für die Karte. Ausserdem sollte man beachten, dass die Farben der Karte nach Möglichkeit nicht mit der Hintergrundfarbe in Konflikt geraten. Für die Kartenfarben orientiert man sich an den traditionellen Farben der analogen Karten, wobei zu beachten ist, dass Farben auf dem Bildschirm oft dunkler erscheinen, als auf dem Papier. Für die Hintergrundfarbe ist es wichtig auf die Assoziationen von Farben zu achten, z.B. grün ist beruhigend, rot vermittelt Wärme und blau wirkt kühl. Das Layout wurde mit wenigen, harmonisierenden Farben gestaltet:

- `aliceblue rgb(240,248,255)`: Hintergrundfarbe
- `dimgray rgb(105,105,105)`: Schrift, Symbole, Linien im Layout
- `lightskyblue rgb(135,206,250)`: Farbe des Ausschnitts in der Referenzkarte
- `darkblue rgb(0,0,139)`: Kreuz in der Referenzkarte (bei hoher Zoomstufe)
- `white rgb(255,255,255)`: Hintergrundfarbe der Legende
- `darksalmon rgb(233,150,122)`: Farbe der Schrift in der thematischen Navigation bei Mouseovereffekts, Aufziehrefteck

Bei der Symbolisierung ist es wichtig, sowohl die kartographischen Regeln, als auch die technischen Grenzen von Bildschirm und Software zu berücksichtigen. In analogen Karten sind die Minimalgrössen viel kleiner als bei Bildschirmkarten. Die Minimalgrösse ist abhängig von der Technik, die verwendet wird um ein Symbol darzustellen und von der Grösse des Bildschirms. Deshalb ist es nicht möglich, bestimmte Werte zu nennen. [RÄBER, JENNY 2000]

Die einzelnen Interaktionsobjekte werden im folgenden Kapitel im Detail erläutert.

## **4.5 Interaktivität**

### **4.5.1 Räumliche Navigation**

Navigation erlaubt dem Benutzer seinen Weg innerhalb dem Informationsangebot zu finden. Gelungene Navigation führt den Nutzer einerseits, steht ihm aber gleichzeitig grosse Flexibilität zu. Sie zeigt dem Benutzer gleichzeitig den Kontext (Übersicht) wie auch Details, dies wenn möglich gleichzeitig graphisch wie numerisch. In dieser Ap-

plikation wird zwischen räumlicher und thematischer Navigation unterschieden. Die räumliche Navigation ermöglicht das Navigieren in der Karte, die thematische Navigation erlaubt das Explorieren von Themen, in diesem Fall die Wanderwegeinformationen und damit verbundene Themen.

Zoomen und Navigieren brauchen geeignete Steuerelemente. Es empfiehlt sich, eine Übersicht mit dem momentan sichtbaren Ausschnitt zu zeigen. Es ist ausserdem wichtig, dass der Benutzer die Möglichkeit hat, die Navigation durch geeignete Funktionen (Lupe, Verschiebehand, etc.) möglichst einfach selbst durchzuführen.

Die Elemente der räumlichen Navigation befinden sich oben rechts in der Karte.

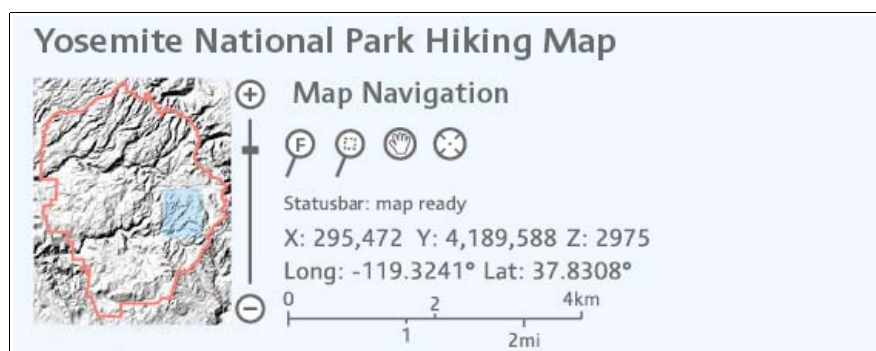


Illustration 39: Räumliche Navigation

Das Lupensymbol mit dem “F” bedeutet, dass man von einem eingezoomten Kartenausschnitt durch einfaches Klicken wieder auf die ganze Karte (*Fullscreen*) herauszoomen kann. Die Lupe mit dem angedeuteten Kästchen lässt den Nutzer manuell in der Karte einen Ausschnitt auswählen zu dem er hinzoomen möchte. Dazu klickt man auf das Symbol, geht in die Hauptkarte und zieht ein Rechteck auf. Das Rechteck weist zu Beginn immer automatisch die minimale Grösse und während dem Ziehen die korrekten Seitenverhältnisse der Hauptkarte auf, egal wie der User das Rechteck aufzieht. Die Hand lässt den Nutzer die Karte verschieben (*pan*) und mit dem “X” kann man den Mittelpunkt der Karte festlegen. Dabei klickt man auf das Symbol und definiert einen neuen Mittelpunkt durch klicken in der Karte. Alle Werkzeuge müssen bei mehrmaliger Benutzung auch mehrmals angeklickt werden.

Die Statusleiste (*status bar*) befindet sich unterhalb der Navigationselemente und zeigt Funktionen und Bedeutungen der Navigationselemente an. Ausserdem zeigt sie an, ob die Karte vollständig geladen ist, oder ob noch Elemente am Laden sind, z.B. das DTM.

Die nächsten zwei Angaben sind die Koordinaten der momentanen Position der Maus in der Karte. Die Koordinaten werden angezeigt, wenn sich die Maus in der Hauptkarte oder in der Referenzkarte befindet. Dies erfolgt sowohl in UTM Koordinaten als auch in Geographischen Koordinaten. Ist die Ebene DTM in der Ebenenkontrolle eingeschaltet, werden zusätzlich die Z-Koordinaten (Höhe) angegeben. Um diese Koordinaten anzeigen zu können, werden im SVG die Koordinaten definiert. Damit dies funktioniert, wird angenommen, dass das Verhältnis zwischen Breite und Höhe der Karte in Pixel

gleich ist mit dem Verhältnis zwischen Breite und Höhe der Karte in Meter.

```
<svg id="mainMap" x="0" y="15" viewBox="244000 -4231000 64000 81000"
width="553" height="700">
```

`x` und `y` beschreiben die linke obere Ecke der Karte auf dem Bildschirm und `width` und `height` geben die Höhe und Breite der Karte im Koordinatensystem des Bildschirms (Pixel) an. Die `viewBox` definiert das innere, kartenorientierte Koordinatensystem (UTM Zone 11): X- und Y-Koordinaten der linken oberen Ecke, Breite und Höhe der Karte in Metern.

Zuletzt gibt es noch eine Massstabsleiste, die ihre Angaben sowohl in Kilometer bzw. Meter als auch in Meilen bzw. Yards macht. Der effektive numerische Massstab einer Webkarte kann wegen der verschiedenen Bildschirmauflösungen und Monitorgrößen nicht berechnet werden. Dazu müsste die effektive Grösse des Monitors oder der Projektionsfläche des Kartennutzers bekannt sein. Die Massstabsleiste in dieser Applikation ändert sich proportional zur Zoomstufe der Karte. Die Länge der Leiste wird immer auf die nächste runde Längeneinheit abgerundet. Ist die angegebene Länge weniger als ein Kilometer, werden Meter angezeigt. Dasselbe gilt für die Meilen. Ist die Länge unter einer halben Meile, werden Yards angegeben.

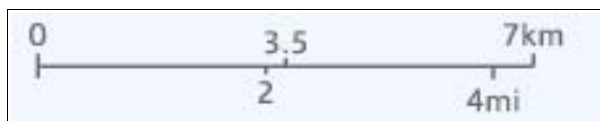


Illustration 40: Massstabsleiste mit Kilometer- und Meilenangabe

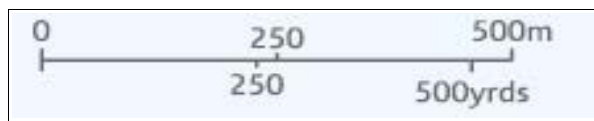


Illustration 41: Massstabsleiste mit Meter- und Yardsangabe

#### 4.5.2 Räumlicher Bezug

Die Referenzkarte zeigt an, welcher Ausschnitt in der Karte momentan dargestellt wird. Mit Hilfe der räumlichen Navigation (*Map Navigation*), kann der Ausschnitt beliebig verändert werden. Auf der rechten Seite der Referenzkarte befindet sich ein Schieberegler, der graphisch anzeigt, in welcher Zoomstufe die Karte sich befindet. Mit den Plus- und Minussymbolen oder indem der Zeiger durch Anklicken und/oder Ziehen mit der Maus bewegt wird, kann die Zoomstufe verändert werden. Es ist auch möglich das blaue Rechteck, momentaner Ausschnitt der Karte, anzuklicken und zu verschieben, oder in der Referenzkarte durch Klicken den neuen Kartenmittelpunkt zu bestimmen. Dadurch ändert sich der Kartenausschnitt in der Hauptkarte. Das blaue Rechteck, das den momentanen Kartenausschnitt anzeigt, wird bei starkem Einzoomen durch ein zusätzliches Symbol unterstützt. Das repräsentative Rechteck in der Karte wäre sonst zu klein um vom Nutzer wahrgenommen zu werden.

### 4.5.3 Ebenenkontrolle



Illustration 42: Ebenenkontrolle

Die Ebenenkontrolle (*map layers*) dient dazu, dem Nutzer die Möglichkeit zu geben, die gewünschten Ebenen selbständig ein- und ausschalten zu können. Dies kann jederzeit erfolgen. Verfügbare Ebenen sind die Hydrologie (*Hydrology*), die Vegetation (*Landcover*), die Höhenlinien (*Contour Lines*), die Beschriftung (*Names*), der Verkehr (*Transportation*), die Wanderwege (*Trails*), die Gebäude (*Buildings*), die Geologie (*Geology*), das Relief (*Shaded Relief*), das Orthophoto (*Orthophoto*), und das DTM (Digital Terrain Model), um die Z- oder Höhenwerte anzeigen zu können. Nach dem Laden der Applikation sind die Ebenen Hydrology, Landcover, Names, Transportation und Shaded Relief bereits aktiviert. Die Wanderwege können jeder Zeit eingeschaltet werden, werden allerdings erst ab einer Kartenbreite kleiner als 12000 Meter angezeigt. Ebenso wechselt die Geologiekarte von generalisiert auf detailliert bei einem Schwellwert von 10000 Metern und die Vegetationskarte bei einer Breite von 5000 Metern.

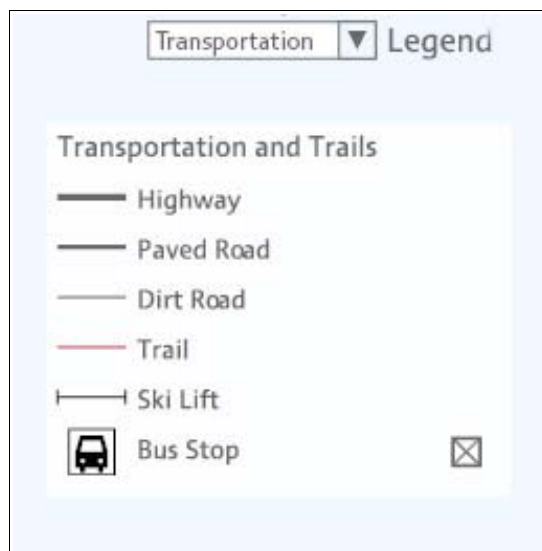


Illustration 43: Beispiel einer Legende (Verkehr)

Für die Ebenen Hydrologie, Vegetation, Beschriftung, Verkehr, Geologie und die Eigenschaften (*Features*) der Wanderwege können Legenden angezeigt werden. Die Legende

für die Geologie gibt es in einer generalisierten und einer detaillierten Form. Die Darstellung der Geologie-Legende wechselt automatisch, wenn die Karte einen bestimmten Zoomfaktor überschreitet. Die Legende der Vegetation beinhaltet alle Klassen, unabhängig von der Zoomstufe und dem Grad der Generalisierung.



Illustration 44: Legende für die generalisierte Geologiekarte

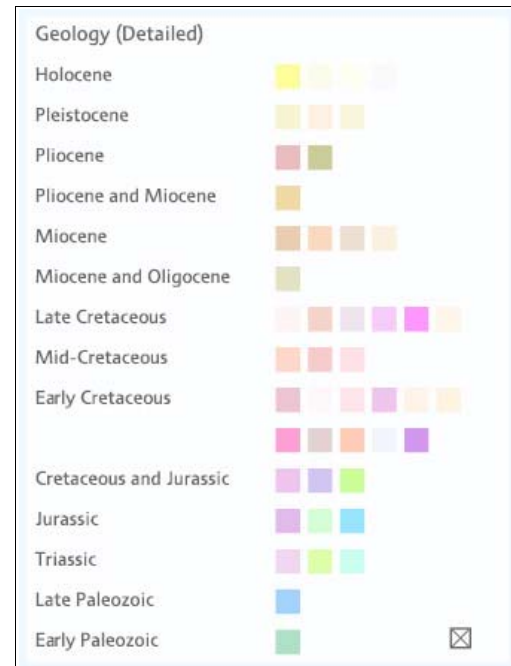


Illustration 45: Legende für die detaillierte Geologiekarte

Wenn in der Karte über die Vegetation bzw. Geologie mit der Maus gefahren wird, wird die zugehörige Legendenklasse durch eine graue Umrandung hervorgehoben. Die Legenden haben einen transparenten Hintergrund und können unten rechts mit dem Kreuz wieder geschlossen werden.

Legenden sind für die qualitative und quantitative Bedeutung der Signaturen von grosser Wichtigkeit. Sie müssen den gesamten Karteninhalt darstellen und erklären. Ausserdem sollten sie sinnvoll gegliedert sein, um dem Nutzer das Lesen und Interpretieren zu erleichtern.

#### 4.5.4 Attributdatenanzeige

<b>Infopanel</b> Height (m): 3048 Height (ft): 10000	<b>Infopanel</b> Name: Yosemite Creek Granodiorite Period: Late Cretaceous
--	--

In der Attributdatenanzeige (*Infopanel*) werden alle Daten, ausgelöst von einem *mouseover-event*, angezeigt, d.h. Informationen zu den Ebenen Strasse, Hydrologie, Vegetation, Geologie, Höhenlinien etc. Die Höhenlinien werden sowohl in Meter, als auch in Feet angezeigt. Die Information zur Geologie gliedert sich in den Namen und die Periode des Gesteins. Die Gebäudeinformationen werden in Name und Art des Gebäudes unterteilt.

#### 4.5.5 Thematische Navigation

In diesem Bereich kann der Nutzer aus den Themen Wanderwege (*Trails*), Photogalerie (*Photo Gallery*), allgemeine Informationen (*General Information*) oder Links wählen. Quellenangaben und Danksagungen (*Acknowledgements*) sind hier ebenfalls zugänglich. Der eigentliche Teil der Applikation (Wanderwege und Photogalerie) spielt sich hier ab. Um einen Themenbereich wieder zu verlassen und zur vorhergehenden Auswahl zurückzukehren, stehen “back-buttons” zur Verfügung. Die nachfolgenden Kapitel beschreiben die Themen ausführlicher.

#### 4.5.6 Wanderwege

Die Wanderwege lassen sich nach verschiedenen Kriterien auswählen und werden anschließend angezeigt und animiert. Ausserdem werden die relevanten Daten zu den einzelnen Wegen aufgelistet.



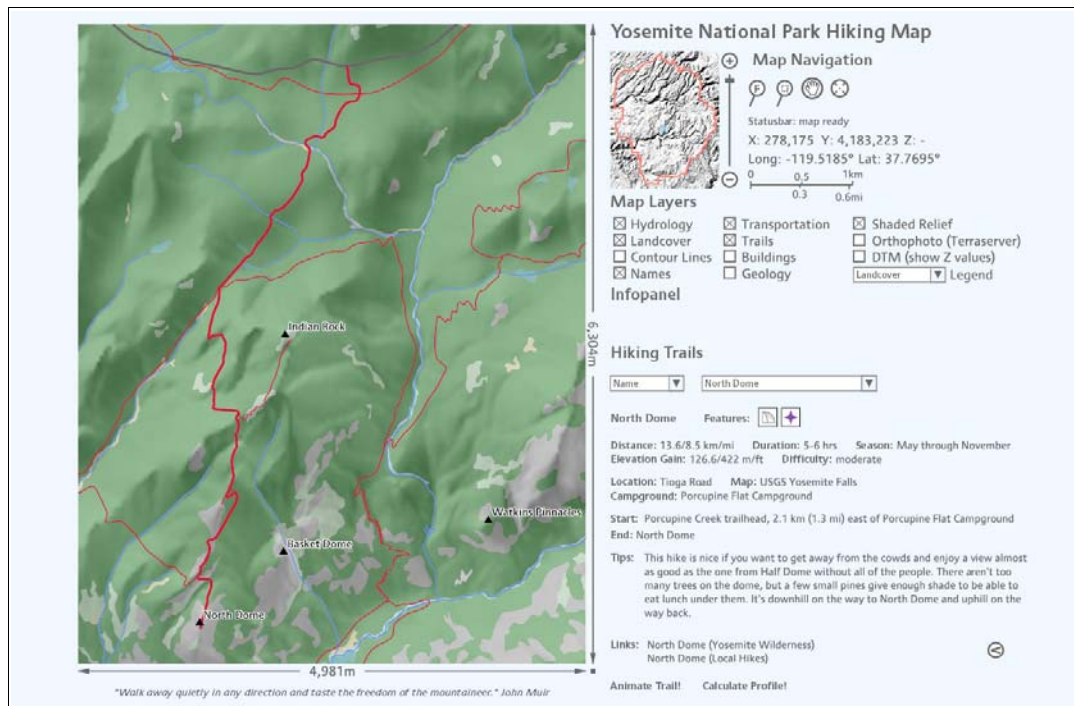


Illustration 46: Beispiel eines Wanderweges mit Informationen

#### 4.5.6.1 Auswahl nach Benutzerdefinierten Kriterien

Die Auswahl der Wanderwege erfolgt über den Namen, den Schwierigkeitsgrad, die Dauer und den Ort. Die Namen werden alphabetisch sortiert. Beim Schwierigkeitsgrad kann der Nutzer die gewünschte Schwierigkeit auswählen, bevor die dabei gefilterten Wege angezeigt werden. Das gleiche erfolgt bei der Auswahl nach Dauer und der Region.

Die Auswahl wird über Selektionslisten getroffen. Diese Listen werden aus den Informationen des XML-Files `trails.xml` generiert und je nach Abfrage, entsprechend der Kriterien gefiltert. XML bietet hier den Vorteil, dass die Wanderwege zentral, mit allen Eigenschaften verwaltet werden und strikt vom Benutzerinterface entkoppelt werden. Sollte man Wanderwege hinzufügen oder wegnehmen wollen, ist dies jeder Zeit möglich, ohne dass Veränderungen in der Applikation durchgeführt werden müssen.

#### 4.5.6.2 Anzeige und Animation des gewählten Wanderweges

In dem Script für die Wanderwege (`trails.js`) werden zuerst alle Daten des Wanderweges aus der XML-Datei gelesen (`function parseTrails()`) und im Objekt `trail` abgespeichert. Anschliessend werden die Auswahllisten anhand der Eigenschaften der Wanderwege kreiert. Diese Auswahllisten, die in der Funktion alphabetisch gereiht werden, lassen den Nutzer eine Wanderung nach Name, Schwierigkeitsgrad

(*Difficulty*), Länge (*Duration*) oder Ort (*Location*) aussuchen. Nachdem gewählt wurde, erscheint die nächste zugehörige Auswahlliste, d.h. wählt der Nutzer “Difficulty”, erscheint die nächste Auswahlliste mit den verschiedenen Schwierigkeitsgraden (*easy*, *moderate*, *difficult*, etc.). Nachdem er wiederum ausgewählt hat, z.B. “moderate”, erscheinen alle Wanderwege, die als moderat gelten in einer dritten Auswahlliste.

Wurde vom Nutzer ein Wanderweg ausgewählt, so werden Text- und Graphikelemente zur Anzeige der Wanderweginformationen erzeugt. Zuerst der Titel und die Eigenschaften, dargestellt mit Symbolen, dann weitere Informationen zum Wanderweg (siehe Kapitel “Wanderweginformationen“ ab Seite 67) und die Texte “Animate Trail!” und “Create Profile!”. Darauf folgend kommt der Befehl an die Karte je nach Wanderweg zum festgelegten Ausschnitt hin zu zoomen.

```
myMainMap.setNewViewBox(trailCollection[id].xmin, trailCollection[id].ymin, trailCollection[id].xmax, trailCollection[id].ymax);
```

Ist die Ebene *Trails* noch nicht eingeschaltet, wird sie durch das Script automatisch aktiviert.

```
if (myMainMap.dynamicLayers["trails"].value == "no"){
  myMainMap.dynamicLayers["trails"].value = "yes";
  document.getElementById("trails").setAttributeNS(
    (null, "visibility", "visible");
  document.getElementById("checkCrosstrails").setAttributeNS(
    (null, "visibility", "visible");
}
```

Der Nutzer hat die Möglichkeit den Verlauf des Wanderwegs zu animieren. Wenn er auf “Animate Trail!” klickt, wird der Wanderweg von Anfang des Wanderweges bis zum Ziel progressiv gezeichnet. Die Animation wird mit Hilfe eines ECMAScripts und SMIL<sup>30</sup>-Elementen ermöglicht. Sollte der Ausschnitt der Karte sich geändert oder die Ebene Trails ausgeschaltet haben, springt die Karte wieder zum vordefinierten Ausschnitt zurück und schaltet die Ebene wieder ein.

Um den Wanderweg animieren zu können, müssen zuerst alle Segmente des Wanderwegs zu einem Pfad zusammengefügt werden und ggf. die Richtungen einzelner Segmente geändert werden. Die Informationen zu den Wanderwegen werden direkt aus dem GIS an die Datenbank übergeben und von dort übernommen. Die Richtung der Wanderwege ist abhängig von der Reihenfolge der Digitalisierung. Es gibt nur wenige Wanderwege, deren Segmente alle in die “richtige” Richtung zeigen.

Nachdem der Pfad zusammengesetzt wurde, werden einige Strichattribute gesetzt und die neue Geometrie oberhalb des Wanderwegelayers in den Dokumentenbaum eingefügt. Die Strichstärke wird dicker als die Ursprungsdicke der Wanderwege gesetzt. Der neue Pfad ist zunächst unsichtbar, da eine Strichlierung mit einem sehr langen Intervall (doppelte Weglänge) und einem Strichlierungsoffset von der einfachen negativen Weg-

<sup>30</sup> SMIL (Synchronized Multimedia Integration Language): ein auf XML basierender, vom W3C entwickelter Standard für zeitsynchronisierte, multimediale Inhalte und Animationselemente. [NETLEXIKON]

länge definiert wurde. Der Strichlierungsoffset wird so gesetzt, dass zu Beginn die gesamte Weggeometrie von einer Lücke ausgefüllt ist (siehe Graphik, Situation rot). Danach wird die Animation gestartet und der Strichlierungsoffset wird über die Zeitdauer der Animation vom ursprünglichen Wert der negativen Weglänge bis zum Wert Null animiert. Dadurch entsteht der Eindruck eines progressiven Zeichnens.

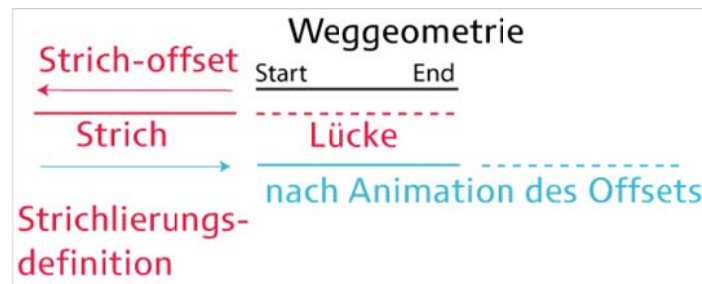


Illustration 47: Darstellung des Offsets für die Animation




Damit nicht der Anschein erweckt wird, alle Wanderwege wären gleich lang, werden Klassen definiert und die Animationsdauer wird proportional zur Länge des Pfades festgelegt.

```
function animateTrail(newHikePath) {
  if (window.navigator) { //only for Adobe viewer
    var segmentLength = Math.round(newHikePath.getTotalLength() * 10) / 10;
    newHikePath.setAttributeNS(null, "stroke-dasharray",
      segmentLength + " " + segmentLength);
    newHikePath.setAttributeNS(null, "stroke-dashoffset",
      segmentLength * -1);
    var animateElement = document.createElementNS(svgNS, "animate");
    animateElement.setAttributeNS(null, "attributeName", "stroke-dashoffset");
    animateElement.setAttributeNS(null, "from", segmentLength * -1);
    animateElement.setAttributeNS(null, "to", 0);
    var duration = Math.round(myMainMap.pixWidth / myMainMap.curWidth
      * segmentLength / 100);
    animateElement.setAttributeNS(null, "dur", duration + "s");
    animateElement.setAttributeNS(null, "fill", "freeze");
    animateElement.setAttributeNS(null, "id", "hikeAnim");
    animateElement.setAttributeNS(null, "begin", "indefinite");
    newHikePath.appendChild(animateElement);
    window.setTimeout("startAnimation('hikeAnim')", 500);
  }
}
```

### 4.5.6.3 Wanderweginformationen

## Hiking Trails

Duration ▼
4-6 ▼
hours
Vernal and Nevada Falls ▼

Vernal and Nevada Falls
Features:




Distance: 11.2/7 km/mi
Duration: 5-6 hrs
Season: Year-round
Elevation Gain: 570/1900 m/ft
Difficulty: moderate-difficult

Location: Yosemite Valley
Map: USGS Half Dome
Campground: Sunnyside Walk-In Campground

Start: Happy Isles
End: Happy Isles

**Tips:** This is THE hike in Yosemite. And the waterfalls look different everytime you go. Try to go early to avoid too many people. Going up the mist trail can be very wet when there is a lot of water and always misty, so watch your step.

**Links:**
[Yosemite Valley Day Hiking \(NPS\)](#)  
[Vernal Falls \(Yosemite Wilderness\)](#)  
[Mist Trail \(Yosemite Wilderness\)](#)  
[Nevada Falls \(Yosemite Wilderness\)](#)

Animate Trail!
Calculate Profile!

Illustration 48: Informationen eines Wanderweges

Jeder Wanderweg, der aufgerufen wird, zeigt eine Reihe von Informationen an. Diese Informationen wurden aus “Yosemite National Park” von Lonely Planet [LONELY-PLANET 2002], “100 Hikes in Yosemite National Park” von Marc Soars [SOARS 2003] und von der Website des Yosemite National Parks [NPS] bezogen. Die Informationen wurden miteinander verglichen und ergänzt, damit ein einigermaßen vollständiges Profil der Wanderwege entsteht.

Bei jedem Wanderweg werden der Name des Wanderwegs und die “Eigenschaften” des Wanderweges als Symbole angezeigt. Die Symbole für die Eigenschaften Fels (*Rocks*), Wasser (*Water*), schöne Aussicht (*Nice View*), Blumen (*Flowers*) und für Kinder geeignet (*Child Friendly*) sind in der auswählbaren Legende erklärt.

Anschliessend wird die Länge des Wanderweges (hin und zurück) in Kilometern und Meilen angegeben. Parallel dazu, die dafür kalkulierte Zeit und in welcher Jahreszeit die Begehung des Weges empfohlen ist. Ausserdem wird angegeben wie viele Höhenmeter zu bewältigen sind und in welcher Schwierigkeitsklasse der Weg eingestuft wird.

Im nächsten Abschnitt wird angegeben in welcher Region des National Parks der Weg liegt (Glacier Point Road, Hetch Hetchy, Tioga Road, Tuolumne Meadows, Wawona oder Yosemite Valley) und welche USGS topographische(n) Karte(n) im Massstab 1:24000 empfohlen ist/sind. Ausserdem wird der naheliegendste Zeltplatz für den Aus-

gangspunkt des Weges angegeben. Als nächstes wird beschrieben wo sich Anfangs- und Endpunkt der Wanderung befinden.

Falls es Tipps zum Wanderweg gibt, werden diese aufgeführt. Das gleiche gilt für eventuelle Links zu aussenstehenden Webseiten, die beim Anklicken in einem separaten Fenster geöffnet werden.

Zum Schluss gibt es die Möglichkeit den Weg zu animieren, siehe “Anzeige und Animation des gewählten Wanderweges“ ab Seite 64, und das Profil berechnen und anzeigen zu lassen, siehe “Profilzeichnung“ ab Seite 68.

#### 4.5.6.4 Profilzeichnung

Die Applikation erlaubt die Darstellung eines interaktiven Höhenprofils, das entlang des Wanderweges aus dem digitalen Höhenmodell berechnet wird.

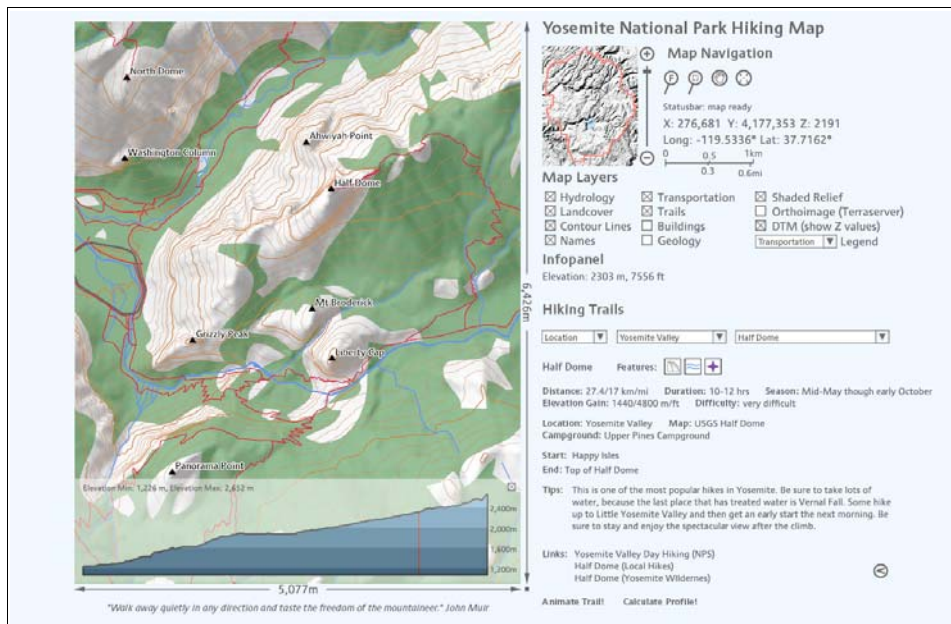


Illustration 49: Beispiel eines Wanderweges mit Profil

Wenn “Calculate Profile!” aktiviert wird, wird der Wanderweg hervorgehoben und nach einer kurzen Verzögerung (Rechenzeit) das Profil angezeigt. Im Profil werden Höhenstufen von dunkel (niedrige Höhenstufen) zu hell (höhere Lagen) dargestellt. Oben links im Profil wird die minimale und maximale Höhe angezeigt. Die Höhenstufenskala auf der rechten Seite des Profils gibt die Höhe in fixen Intervallen an. Fährt man mit der Maus über das Profil, läuft auf dem Wanderweg ein weißer Kreis mit, der die momentane Lage in der Karte korrespondierend zum Profil anzeigt. Gleichzeitig wird im Profil die gegenwärtige Position hervorgehoben. Die Angaben über die Höhe werden in Metern und Feet im Infopanel angezeigt.

Zuerst wird der Kartenausschnitt entsprechend der Ausdehnung der Wanderroute fest-



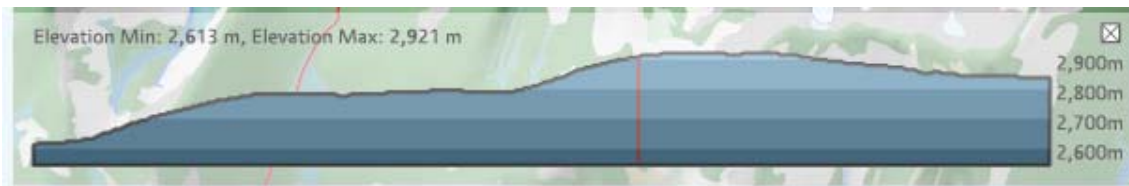


Illustration 50: Detail Profil

gelegt. Ist die Ebene *DTM*, die die Höhenwerte anzeigt, nicht aktiviert, wird sie per Script eingeschaltet. Zum Zusammensetzen der einzelnen Segmente zu einem gemeinsamen Pfadelement, wird die gleiche Funktionen wie bei der Animation der Wege verwendet.

Um das Profil auszurechnen, wird zuerst der Abstand der Höhenpunkte entlang des Pfades ausgerechnet. Die Anzahl der Höhenpunkte ist fest. Daraus ergibt sich ein kürzeres Intervall bei kurzen Wanderrouen und daher auch ein detaillierteres Höhenprofil. Entlang der Wanderweggeometrie werden nun in fixen Intervallen die Positionen zur Bestimmung der Höhenwerte berechnet. Dafür wird die SVG-DOM Methode der Pfad-elemente `.getPointAtLength()` verwendet. Da die errechneten Positionen meist nicht exakt mit den Stützpunkten des Geländemodellgitters zusammenfallen, wird zwischen den vier Nachbarstützpunkten bilinear interpoliert. Diese Funktion ist eine Methode des Geländemodellobjektes und wird auch bei der Anzeige der Höhenwerte in der Karte verwendet. Während der Berechnung der Höhenwerte werden gleichzeitig minimale und maximale Höhenwerte bestimmt. Dieser Bereich ist relevant für die Skalierung des Profils und die Bestimmung der Höhenstufenintervalle.

Nachdem diese Werte festgelegt wurden, wird das Profil gezeichnet. Ein weisses, leicht transparentes Hintergrundrechteck wird zur besseren Lesbarkeit zwischen Karte und Profil eingefügt. Das Profil wird am unteren Ende der Karte angezeigt. Da das Profil eine Mindesthöhe aufweisen muss, um es angemessen betrachten und bedienen zu können, wird das Profil meist überhöht.

```
if (profileRange < 100) {
  myMeterHeight *= 0.25;
}
else if (profileRange >= 100 && profileRange < 500) {
  myMeterHeight *= 0.5;
}
```

Ist die Höhe des Profils kleiner als 100, wird die Höhe mit 0.25 multipliziert. Wenn die Höhe des Profils grösser oder gleich 100 oder kleiner als 500 ist wird das Profil mit 0.5 multipliziert. Dies ist notwendig, um die Überhöhung bei geringer Reliefeenergie nicht zu übertreiben.

Es werden kleine, zweieinhalb Pixel breite, vertikale Rechtecke leicht überlappend im Abstand von zwei Pixeln gezeichnet. Die Überlappung dient zur Vermeidung von Renderingartefakten. Gleichzeitig werden die Events für die Mouseovereffekte<sup>31</sup> an die

<sup>31</sup> Mouseovereffekt: Ein Effekt, das durch eine Bewegung der Maus über ein Objekt ausgelöst wird



Rechtecke angehängt und eine Pfadgeometrie erzeugt die zum Schluss das Profil umranden wird und über die vertikalen Rechtecke gelegt wird. Dieser Pfad dient zum Überdecken des Treppenstufeneffektes der vertikalen Rechtecke (siehe Graphik unten), aber auch als Clipping-Pfad für die horizontalen Rechtecke, die später die Höhenstufen darstellen. Durch eine Schleife über alle Höhenwerte, wird dieser Pfad bei jedem Durchgang länger, bis alle Werte durchlaufen worden sind. Darauf folgend wird die Linie entlang der Profilbasis weiter gezeichnet bis zurück zum Ausgangspunkt, damit das Profil "eingerahmt" wird. Die Länge des Profils in Pixel beträgt immer 500 Einheiten, egal wie lang der Weg effektiv ist.

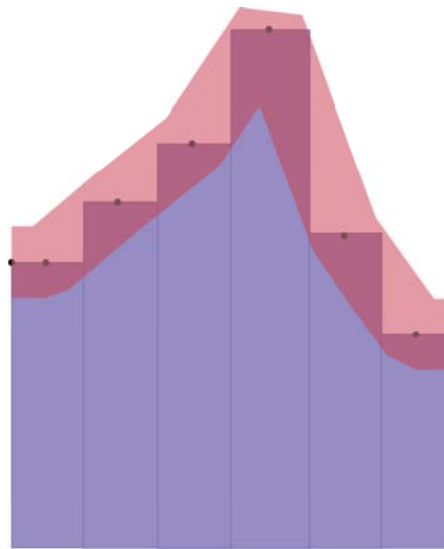


Illustration 51: Stark übertriebene Schemadarstellung der vertikalen Profilrechtecke und der darüberliegenden Abschlusslinie

Als nächstes wird das Symbol kreiert, das die momentane Position entlang des Wanderweges in der Karte anzeigt. Das Symbol wurde im SVG als Symbol definiert und wird im Script mit einem `use`-Element aufgerufen. Mit `translate("+point.x"+"point.y+")` wird die Position des Symbols bestimmt.

```
var point = newHikePath.getPointAtLength(0);
var profileSymbol=document.createElementNS(svgNS,"use");
profileSymbol.setAttributeNS(xlinkNS,"href","#symbolOnProfile");
profileSymbol.setAttributeNS(null,"transform","translate("+point.x+"
    "+point.y+")",scale("+ (myMainMap.curWidth*0.002)+")");
profileSymbol.setAttributeNS(null,"pointer-events","none");
profileSymbol.setAttributeNS(null,"id","symbOnProfile");
newHikePath.parentNode.appendChild(profileSymbol);
```

Die Höhenstufen folgen als nächstes in der Funktion. Je nach Höhendifferenz des Profils, werden die Intervalle festgelegt. Z.B. bei einem Höhenunterschied von unter

1000m sind die Intervalle im Abstand von 200 Metern oder wenn der Höhenunterschied weniger als 150m beträgt, ist der Abstand bei 50 Metern. Die Höhenstufen werden entsprechend der Profilbreite als transparente horizontale Rechtecke über die zuvor generierten vertikalen blauen Rechtecke gelegt. Jede Stufe besteht aus Grauwerten. Die unterste Stufe ist schwarz und die oberste ist weiss. Daraus entstehen blau-graue Mischfarben. Die Höhenstufenrechtecke werden mit dem zuvor erstellten Clippingpath auf die Fläche des Profils reduziert.

Schliesslich legt die Funktion `showProfileData()` die Verhaltensweise des Mouseovereffekts im Profil fest. Wenn die Maus sich im Profil bewegt, bewegt sich das Symbol entlang des Wanderwegs und die Höhenwerte werden im Infopanel angezeigt. Die Koordinaten werden wie folgt errechnet: Das Attribut `distFrStart` kann bei jedem Rechteck ausgelesen werden und liefert die Distanz des jeweiligen Punktes gegenüber dem Startpunkt der Wanderung. Die X- und Y-Koordinaten des Standpunkts werden ebenfalls mit der Methode `getPointAtLength` errechnet und an die translate-Definition des Symbols weitergegeben. Somit kann das Symbol an der richtigen Stelle platziert werden.

Um das Profil jeder Zeit wieder schliessen zu können, erscheint in der rechten oberen Ecke ein Kreuz, das beim daraufklicken, das Profil schliesst. Wird ein neuer Wanderweg gewählt, wird das alte Profil automatisch geschlossen.

#### 4.5.7 Photogalerie

Geht der Nutzer in die Photogalerie, hat er die Möglichkeit die Bilder, die sich in verschiedenen Kategorien befinden, auszuwählen. Die Kategorien sind: Geomorphologie (*Geomorphology*), See/Fluss (*Lake/River*), Wiese (*Meadow*), Gipfel (*Peak*), Vegetation (*Vegetation*), Aussicht (*View*) und Wasserfall (*Waterfall*). Da manche Bilder in mehrere Kategorien eingeordnet wurden, können sie mehrmals vorkommen. Nachdem eine Kategorie ausgewählt wird, werden Thumbnails der Bilder (jeweils vier auf einmal) angezeigt. Wenn mehrere Bilder vorhanden sind, kann noch auf der rechten Seite geblättert werden. Nachdem der Nutzer ein Thumbnail gewählt und auf ihn geklickt hat, wird das Bild vergrössert. Unterhalb des Bildes (Querformat) oder auf der rechten Seite des Bildes (Hochformat) wird der Titel, der Photograph, das Aufnahmedatum (falls vorhanden), die UTM-Koordinaten (falls vorhanden) und das Azimut (falls vorhanden) des Bildes angezeigt. Gleichzeitig zoomt die Karte auf den Punkt ein, wo das Bild aufgenommen wurde. Im Mittelpunkt der Karte (Aufnahmestandpunkt) wird das Symbol einer Kamera platziert, die in die Richtung zeigt (Azimuth), in der das Bild aufgenommen wurde.

Technisch gesehen werden die Daten der Bilder erst aus der XML-Datei mit Hilfe eines ECMAScripts eingelesen. Dann werden die Daten des jeweiligen Bild als `photo` Objekt abgelegt. Anschliessend entsteht die Auswahlliste anhand der Keywords (Geomorphologie, Wiese, Gipfel, etc.), die dann alphabetisch von A bis Z sortiert erscheint. Wird ein Thema ausgewählt, geht eine Schleife über alle Photos und sortiert die Photos mit passendem Keyword aus.

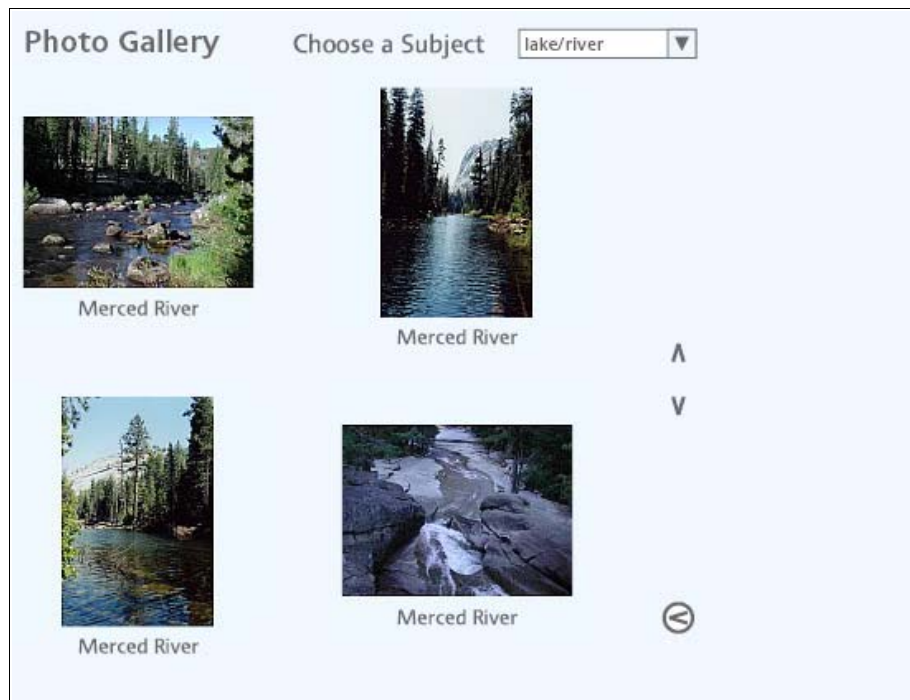


Illustration 52: Photogalerie

In der Galerie werden die Photos in Vierer-Gruppen in einer Grösse von 130x130 Pixel angezeigt. Wenn mehr als vier Photos in der gesamten Gruppe vorhanden sind, dann erscheinen rechts von den Bildern Pfeile zum Weiterblättern, damit die nächste Gruppe von vier oder weniger Bildern erscheint.

Sind genauere Angaben zum jeweiligen Bild erwünscht, kann man auf das gewollte Bild klicken. Dabei wird ein grösseres Bild entweder in Quer- oder Hochformat dargestellt. Unterhalb des Bildes (Querformat) oder rechts des Bildes (Hochformat) werden der Titel des Photos, der oder die Photograph/in und das Aufnahmedatum angezeigt. In der zweiten Zeile befinden sich die UTM Koordinaten und der Azimuth der Aufnahme-richtung. Die Karte wird auf den Ausschnitt der Aufnahme-Position gezoomt. Die Aufnahme-koordinaten befinden sich in der Mitte der Karte, die jeweils einen Ausschnitt von 3000 Metern darstellt. Das Kamerasymbol wird referenziert und an diesen Koordinaten platziert. Das Symbol wird zusätzlich durch ein animiertes Fadenkreuz hervorgehoben. Um das Bild wieder schliessen und zur Galerie zurückkehren zu können befindet sich unten rechts ein "Close-Button".



Illustration 53: Georeferenziertes Photo mit Angaben

Die Photos sind hauptsächlich ausgewählte Photos, die von CalPhotos<sup>32</sup> [CALPHOTOS] stammen. Die restlichen stammen aus dem Fundus von U. Mattle, A. Neumann und J. Williams. Die zwei integrierten Webcams werden von der Yosemite Association zur Verfügung gestellt.

#### 4.5.8 Touristische Informationen

Die touristischen Informationen gliedern sich in Anfahrts-, Übernachtungs- und Verpflegungsmöglichkeiten. Es besteht die Möglichkeit per Flugzeug, Bus, Zug oder Auto nach Yosemite zu reisen. Die meisten Besucher des Parks ziehen es vor mit dem Auto durch einen der vier Eingänge einzureisen. Wenn man mit dem Flugzeug oder Zug ankommt, muss zum Schluss noch in einen Bus umgestiegen werden.

<sup>32</sup> CalPhotos beinhaltet über 80000 Bilder von Pflanzen, Tieren, Fossilien, Menschen und Landschaften, die von Gruppen und Einzelpersonen beigetragen werden. Diese Seite wird von der University of California, Berkeley verwaltet.

**General Information: Getting There**

Air: Fresno-Yosemite Intl Airport (90 mi)  
 Train: Amtrak to Merced, transfer to YARTS or VIA  
 Bus:  
 YARTS(Yosemite Area Transport System) from Merced or  
 Mammoth Lakes (summer)  
 Greyhound: Buses to Merced, continue with YARTS  
 Car:  
 South Entrance on Hwy 41 (Wawona Rd)  
 Arc Rock Entrance on Hwy 140 (El Portal Rd)  
 Big Oak Flat Entrance on Hwy 120 (Big Oak Flat Rd)  
 Tioga Pass Entrance on Hwy 120 E (Tioga Rd): open Jun-Oct



Die Rubrik “Places to Stay” ist eine Liste von Campingplätzen, Hotels und anderen Möglichkeiten, in denen der Besucher übernachten kann.

**General Information: Places to Stay**

<b>Camping</b>	<b>Lodging</b>
Bridalveil Creek	Ahwahnee Hotel
Camp 4	Curry Village
Crane Flat	Housekeeping Camp
Hadgdon Meadow	Tuolumne Meadows Lodge
Lower Pines	Wawona Hotel
North Pines	White Wolf Lodge
Porcupine Flat	Yosemite Lodge
Tamarak Flat	
Tuolumne Meadows	<b>Other Possibilities</b>
Upper Pines	High Sierra Camps (Jul, Aug)
Wawona	Wilderness Permit
White Wolf	
Yosemite Creek	



Die Liste mit Verpflegungsmöglichkeiten listet auf, an welchen Orten der Besucher eine Möglichkeit hat sich zu verpflegen.

**General Information: Places to Eat**

<b>Yosemite Village</b>	<b>Wawona</b>
Degnan's Deli	Wawona Hotel
Degnan's Café	Wawona General Store
Degnan's Loft	Pine Tree Market
Village Grill	<b>Tuolumne Meadows</b>
Village Store	Tuolumne Meadows Lodge
<b>Curry Village</b>	Tuolumne Meadows Grill
Taco Stand	Tuolumne Meadows Store
Pizza Patio	
Curry Dining Pavillion	<b>Ahwahnee Hotel</b>
Coffee Corner	<b>Yosemite Lodge</b>
Terrace Bar	<b>White Wolf</b>
Curry Village Store	



#### 4.5.9 Externe Links

Diese kleine Liste gibt dem Wanderer die Möglichkeit ergänzende Informationen von anderen Webseiten einzuholen. Die Seiten öffnen jeweils in einem neuen Fenster.





## 5 Schlussfolgerungen

### 5.1 Zusammenfassung und Beurteilung der Ergebnisse

Das Ziel der Arbeit, eine interaktive Wanderkarte des Yosemite National Parks mit Hilfe von SVG zu entwickeln, ist als gelungen zu betrachten. Im Zuge des Projekts wurde eine der ersten interaktiven, vektorbasierten Onlinekartenapplikationen mit Datenbankanbindung entwickelt. Die Vektorbasis ermöglichte es neue Qualitätsmassstäbe in Bezug auf graphische Qualität und Interaktivität zu setzen. Die weitgehend konsequente Trennung von Inhalt und Präsentation ermöglicht eine rasche Anpassung der Applikation an andere Regionen. Diese Trennung wurde v.a. durch den Einsatz von Datenbanken und XML umgesetzt. Diese Technologien erlauben schnelle Anpassungen und Korrekturen der Inhalte ohne die Programmlogik verändern zu müssen. Daher wäre es durchaus denkbar in dieser Applikation tagesaktuelle Informationen in die Karte und die umliegenden Informationsbereiche zu integrieren.

Für die Gestaltung der Applikation erwies sich SVG als gut geeignet. Grosse Teile der Clientapplikation wurden mit Hilfe von ECMAScript umgesetzt. Eine Umsetzung einer derartigen Webapplikation ohne clientseitige Programmiersprache wäre unmöglich gewesen. Die Anwendung wurde während der Programmierung und in der Endphase des Projekts jeweils im Adobe SVG-Viewer (Versionen 3 und 6) und im Apache Batik SVG-Viewer getestet. Sie läuft in allen aktuellen Webbrowsern in denen das Adobe SVG Plugin installiert ist. Sowohl die Pluginvariante als auch die standalone Batikvariante wurde in den Betriebssystemen Linux, MacOSX und Windows (2000, XP) getestet.

Keiner der oben angeführten SVG-Viewer unterstützt heute bereits den vollen SVG 1.1 Sprachumfang. Jedoch reicht die Funktionalität der Viewer bereits aus, um ansprechende interaktive Kartenapplikationen zu programmieren. Eine vollständige Umsetzung der SVG 1.1 Features würde die Erstellung dieser Applikation erleichtern und beschleunigen. So fehlen etwa in den Adobe Viewern einige DOM-Methoden zum Zugriff auf Pfadgeometrien und im Batik Viewer sind die SMIL-Animationen noch nicht implementiert. Die Version 3 des Adobe Viewers unterstützt noch keine benutzerspezifischen Cursor und ist bei Animationselementen über komplexem Kartenhintergrund langsamer als die Version 6.

An wenigen Stellen mussten daher im Code Verzweigungen angebracht werden um sowohl die Adobe Viewer (mit Animationsmöglichkeiten) als auch den Batik Viewer (ohne Animationen) zu unterstützen. Trotzdem erwies sich der Test in verschiedenen SVG- und Browserumgebungen als sinnvoll, da Fehlermeldungen in den unterschiedlichen Viewern unterschiedlich informativ ausfielen. So half mal der eine, mal der andere Viewer um die Fehler schneller aufzufinden. Als sehr nützlich erwies sich ausserdem der integrierte Javascript Debugger in Apache Batik und der Umstand, dass Batik einen validierenden XML Parser integriert, der den Programmierer zu korrektem XML/SVG Code zwingt.

Zeitmässig fällt auf, dass die Datenaufbereitung den grössten Teil der Arbeitszeit beanspruchte. Die Datensätze mussten zuerst aufgefunden bzw. von verschiedensten Quellen angefordert werden und lagen in unterschiedlichen Projektionssystemen und Qualitätsklassen vor. Einige der USGS Daten hatten unzureichende Attribute für die Erstellung interaktiver Kartenapplikationen. So mussten viele Namen und Klassenattribute für die Generierung verschiedener Massstäbe ergänzt werden. Zahlreiche Gipfelpunkte wurden manuell ergänzt. Die Redaktion der Wanderwege und Photodaten nahm ebenfalls beträchtliche Zeit in Anspruch.

Eine grössere Hürde bei der automatischen Generierung der SVG Kartendaten stellten die Datenmengen dar. Dieses Problem konnte erst unter Zuhilfenahme der PostGIS Funktionen `Simplify`, `Intersection` und `AsSVG` gelöst werden. Als sehr hilfreich in diesem Zusammenhang erwies sich das Apache Webservermodul `mod_gz`, das eine on-the-fly Komprimierung der Datenströme erlaubt. Das Extrahieren der Geländemodell-daten in verschiedenen Auflösungen konnte mit dem von H. Bär zur Verfügung gestellten Programm Higrd bewältigt werden.

Bei der Programmierung erwies sich die automatische Namensplatzierung als das schwierigste Problem. Die Resultate sind für eine erste Version als brauchbar zu bezeichnen, hier gäbe es aber durchaus Verbesserungsmöglichkeiten, falls man einen iterativen oder einen “Sliding Label” Ansatz verfolgen würde. Einen solchen Ansatz verfolgte etwa Marc van Kreveld [KREVELD 1998]. Dies wäre v.a. bei sehr dichtem Namensgut notwendig. Die Umrechnung von UTM nach geographischen Koordinaten (v.a. für GPS Nutzer interessant), sowie die Berücksichtigung der exotischen amerikanischen Längeneinheiten erwiesen sich als kleinere Hürden.

Einige Interaktivitätsvorgänge könnten in ihrem Ablauf verbessert werden. Benutzerstudien würden hier unterstützend wirken. Als Beispiel wäre hier die Legende zu erwähnen, die momentan nicht verschiebbar ist. Bei den Navigationstools wäre es empfehlenswert eine Mehrfachbenutzung zu ermöglichen, ohne dass das Tool jedes Mal neu aktiviert werden muss. Ebenso wäre hier ein Hin- und Zurückknopf sinnvoll, um die letzten Kartenausschnitte betrachten zu können. Eines der grössten Mankos ist derzeit das Fehlen der Druckfunktion.

## 5.2 Weiterführende Möglichkeiten der Arbeit

Eine derartige Applikation kann nie als “fertig” bezeichnet werden. Es müssten alle Wanderwege vollständiger mit Informationen ausgestattet werden. Die georeferenzierten Bilder sollten im Umfeld des Wanderweges eingeblendet werden. Zusätzliche georeferenzierte Informationen wie Schwimmmöglichkeiten, Aussichtspunkte, etc. würden den Informationsgehalt zusätzlich anreichern. Die Standorte der Verpflegungs- und Übernachtungsmöglichkeiten sollten in der Karte visualisiert werden. Ein geographisches Ortsverzeichnis würde das schnelle Aufwenden von Lokalitäten ermöglichen.

Es ist denkbar, dass eine derartige Applikation in den Visitor Centern des Yosemite National Parks für Wanderer und andere Interessenten bereit stehen könnte. Hier könnten sich potentielle Wanderer schnell und bequem über Wanderwege informieren. Mit einer zusätzlichen Druckfunktion wäre es möglich, den gewählten Wanderweg über ein PDF-File auszudrucken und auf der Wanderung als Hilfsmittel mitzunehmen. Das Layout liesse sich, wenn erwünscht, einfach an das Layout des National Parks anpassen. Doch nicht nur Yosemite sondern auch andere National Parks und Wandergegenden würden von einer solchen Applikation profitieren.

Eine andere, und sicherlich erstrebenswerte Erweiterung, wäre eine offline Version dieses Projekts. Somit könnten Wanderwege ohne Internetanschluss abgerufen werden. Das grösste Problem bei der Erstellung einer offline Applikation wäre die räumliche Datenbank. Derzeit gibt es kaum geographische Datenbanken, die ohne Installationsroutinen und Datenbankkenntnisse der Nutzer lauffähig sind.

In der heutigen Zeit sind Navigationsfunktionen auf mobilen Geräten populär. In Gegenden, wo Empfang möglich ist, wäre es denkbar dass PDA- oder Mobiltelefonbesitzer während der Wanderung ihr mobiles Gerät konsultieren, um Geo-Informationen abzurufen. Solche Location Based Services sind beispielsweise in Japan schon weit verbreitet. Allerdings müsste das Benutzerinterface und die Kartensymbolisierung dafür neu designed werden. Mit Hilfe der Sub-Profile SVG-Tiny<sup>33</sup> oder SVG-Basic<sup>34</sup> wäre es möglich die Applikation auf mobile Telephone oder PDAs zu portieren.

Ein zusätzliches GPS-Modul könnte das Erfassen von Wegpunkten in der Kartenapplikation ermöglichen. Ebenso wäre es interessant die vom GPS-Empfänger aufgezeichneten Routen und Wegpunkte interaktiv zu visualisieren. Weg/Zeit-Diagramme könnten aus dieser Datengrundlage ebenfalls generiert werden.

---

<sup>33</sup> Ein vom W3C definiertes mobiles Profil für mobile Telephone. [W3C SVGMOBILE]

<sup>34</sup> Ein vom W3C definiertes mobiles Profil für PDAs

## 6 Bibliographie

- [**BADER 2004**] **Bader, Herbert**: SVG Reporting: Vektorgraphiken im Web einsetzen, Software & Support Verlag, Frankfurt, 2004
- [**BRASSEL 1988**] **Brassel, Kurt**: Der Generalisierungsbegriff in der Kartographie und anderen Disziplinen , 1988
- [**CRAMPTON 2002**] **Crampton, Jeremy W.**: Interactivity Types in Geographic Visualization , 2002
- [**DOUGLAS et al. 1973**] **Douglas, Davis H.; Peucker, Thomas K.**: Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature , 1973
- [**FIBINGER 2002**] **Fibinger, Iris**: SVG-Scaleable Vector Graphics, Markt und Technik Verlag, München, 2002
- [**FLANAGAN 2002**] **Flanagan, David**: JavaScript- The Definitive Guide, O'Reilly & Associates, California, USA, 2002
- [**HAKE/GRÜNREICH/MENG 2002**] **Hake, Günter; Grünreich, Dietmar; Meng, Liqiu**: Kartographie, Walter de Gruyter, Berlin, 2002
- [**HAROLD et al. 2001**] **Harold, Elliott Rusty; Means, W. Scott**: XML in a Nutshell- A Desktop Quick Reference, O'Reilly & Associates, California, USA, 2001
- [**HENNING 2001**] **Henning, Peter A.**: Taschenbuch Multimedia, Fachbuchverlag Leipzig, Leipzig, 2001
- [**JENNY 2000**] **Jenny, Bernhard**: Computergestützte Schattierung in der Kartografie, 2000
- [**JIANG 1996**] **Jiang, Bin**: Cartographic Visualization: Analytical and Communication Tools , 1996
- [**KIRSCH 1997**] **Kirsch, David**: Interactivity and MultiMedia Interfaces; 1997
- [**KRAAK et al. 2003**] **Kraak, Menno-Jan; Ormeling, Ferjan**: Cartography: Visualization of Geospatial Data, Prentice Hall, Harlow et al., 2003
- [**KREVELD 1998**] **Kreveld, Marc van, Strijk, Tycho, Wolff, Alexander**: Point Set Labeling with Sliding Lables, 1998

- [LERDORF et al. 2002] **Lerdorf, Rasmus; Tatroe, Kevin**: Programming PHP, O'Reilly & Associates, California, USA, 2002
- [LONELYPLANET 2002] **Lonely Planet**: Yosemite National Park, Lonely Planet Publications, Victoria, Australia, 2002
- [NEUMANN et al. 2001] **Neumann, Andreas; Winter, André**: Time for SVG- Towards High Quality Interactive Web-Maps, 2001
- [RÄBER, JENNY 2000] **Räber, Stefan; Jenny, Bernhard**: Attraktive Webkarten- ein Playdoyer für gute Kartengraphik; 2000
- [SGK 1990] **Schweizerische Gesellschaft für Kartographie**: Kartographisches Generalisieren, Schweizerische Gesellschaft für Kartographie, Zürich, 1990
- [SNYDER 1987] **Snyder, John P.**: Map Projections- A Working Manual, US Government Printing Office, Washington DC, USA, 1987
- [SOARS 2003] **Soars, Marc J.**: 100 Hikes in Yosemite National Park, The Mountaineers Books, Seattle, USA, 2003
- [CALPHOTOS] **CalPhotos** : <http://elib.cs.berkeley.edu/photos/>
- [CARTO.NET] **carto.net** : <http://www.carto.net/papers/svg/samples/>
- [NETLEXIKON] **Netlexikon** : <http://www.lexikon-definition.de/>
- [LINOTYPE] **Linotype** : <http://www.linotype.com/>
- [NPS] **National Park Service** : <http://www.nps.gov/yose/>
- [OGC-ISO 19128 2004] **Web Map Service- Version 1.3** : <http://www.opengeospatial.org/specs/?page=specs>
- [POSTGIS] **PostGIS Documentation** : <http://www.postgis.org/docs/>
- [POSTGRESQL 2005] **PostgreSQL 8.0.0 Documentation** : <http://www.postgresql.org/files/documentation/>
- [TAYLOR] **Converstion Table** : <http://www.taylor-made.com.au/billspages/>
- [UCMP BERKELEY] **Web Geological Time Machine** : <http://www.ucmp.berkeley.edu/help/timeform.html>
- [USGS] **United States Geological Survey** : <http://www.usgs.gov/>

**[W3C,SVG] W3C Recommendation Scalable Vector Graphics 1.1 Specification :**  
<http://www.w3.org/TR/SVG11/>

**[W3C SVGMOBILE] Mobile SVG Profiles: SVG Tiny and SVG Basic :**  
<http://www.w3.org/TR/SVGMobile/>

**[YOSEMITE CA] Yosemite CA :** <http://www.yosemite.ca.us/>

**[HUBER 1985] Bedrock Geology of Yosemite Valley :**  
[http://www2.nature.nps.gov/usgsnps/yos/l\\_1639.html](http://www2.nature.nps.gov/usgsnps/yos/l_1639.html)

Weitere verwendete Webseiten:

[www.carto.net/papers/svg/samples](http://www.carto.net/papers/svg/samples)

- Manipulating the SVG-Document with ECMAScript
- SVG SelectionsList Object
- Map Navigation Tools
- Animation Along Path
- Integrating OGC WMS Services with SVG Mapping Applications
- Dynamic Loading of Vector Geodata for SVG Mapping Applications
- Interactive Topography Map Tuerlersee

<http://www.w3.org/TR/SVG/types.html#ColorKeywords> (Farben)

<http://www.ice.ucdavis.edu/cnps/specieslistb.asp> (Pflanzenarten in Kalifornien)

<http://www.yosemite.ca.us/faq/1.html#climate> (Klimawerte des Yosemite Tals)



## 7 Abkürzungsverzeichnis

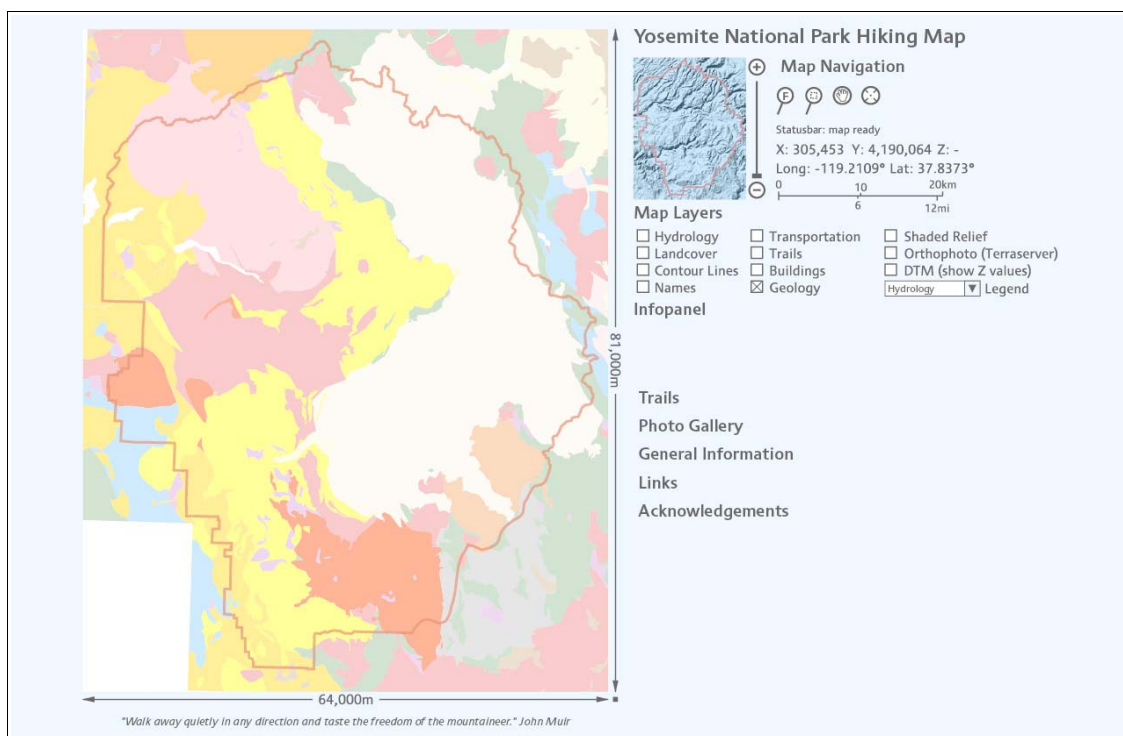
ASV	Adobe SVG Viewer
CGI	Common Gateway Interface
ETH	Eidgenössische Technische Hochschule
GIS	Geographisches Informationssystem
GUI	Graphical User Interface (Benutzeroberfläche)
HTML	Hypertext Markup Language
JPEG	Joint Photographers Expert Group
OGC	Open GIS Consortium
PDA	Personal Digital Assistant
PHP	Hypertext Preprocessor
RGB	Rot-Grün-Blau-Farbmodus
SMIL	Synchronized Multimedia Integration Language
SQL	Structured Query Language
SVG	Scaleable Vector Graphics
TIFF	Tagged Image File Format
USGS	United States Geological Survey
W3C	World Wide Web Consortium
WMS	Web Map Service
XML	Extensible Markup Language

## 8 Anlagen

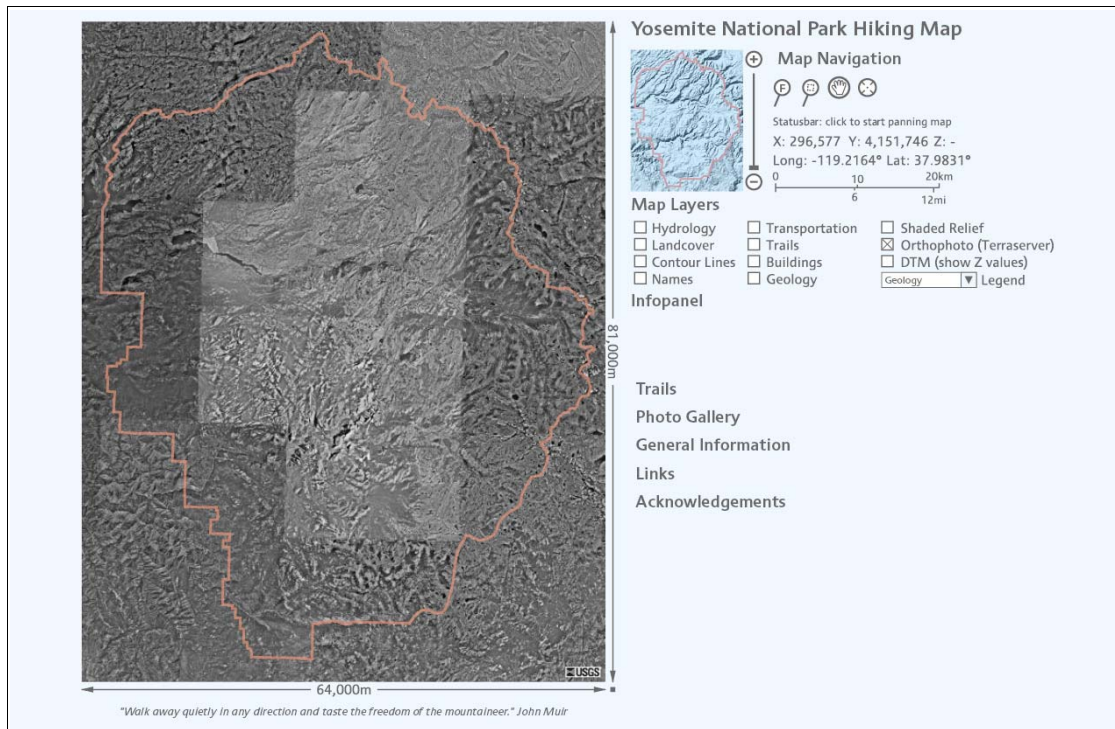
## Anlage 1: Eingangsseite der Applikation



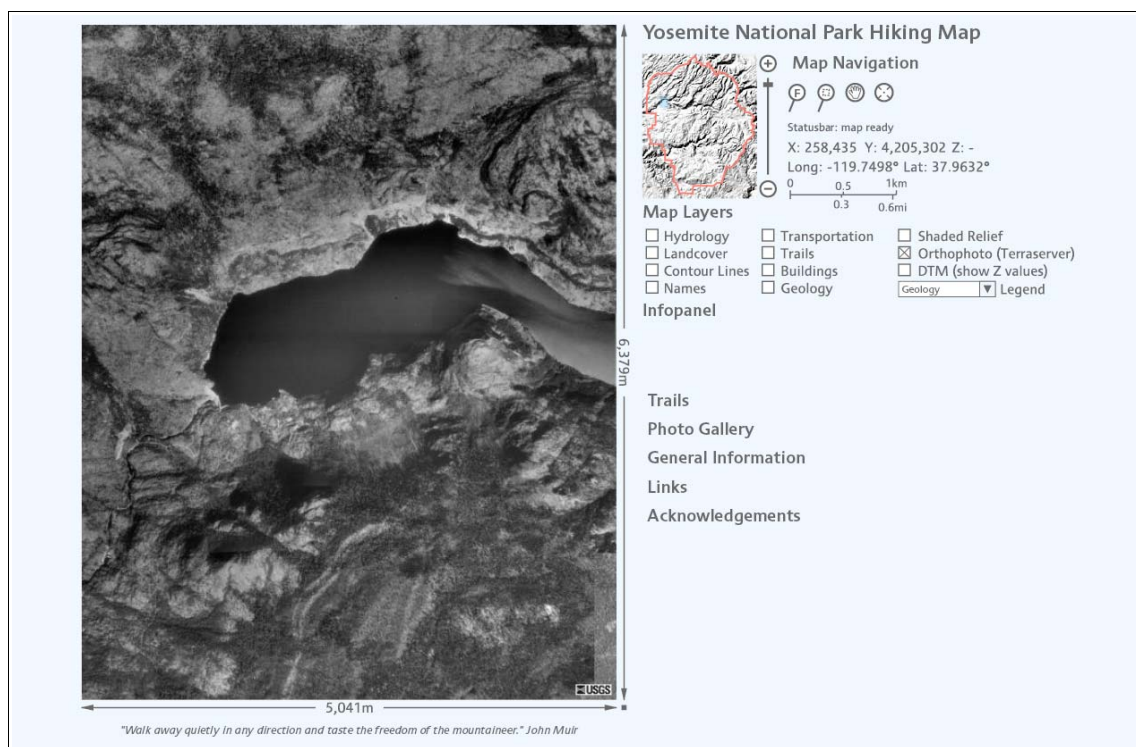
## Anlage 2: Geologiekarte (generalisiert) der Applikation



### Anlage 3: Orthophoto der Applikation

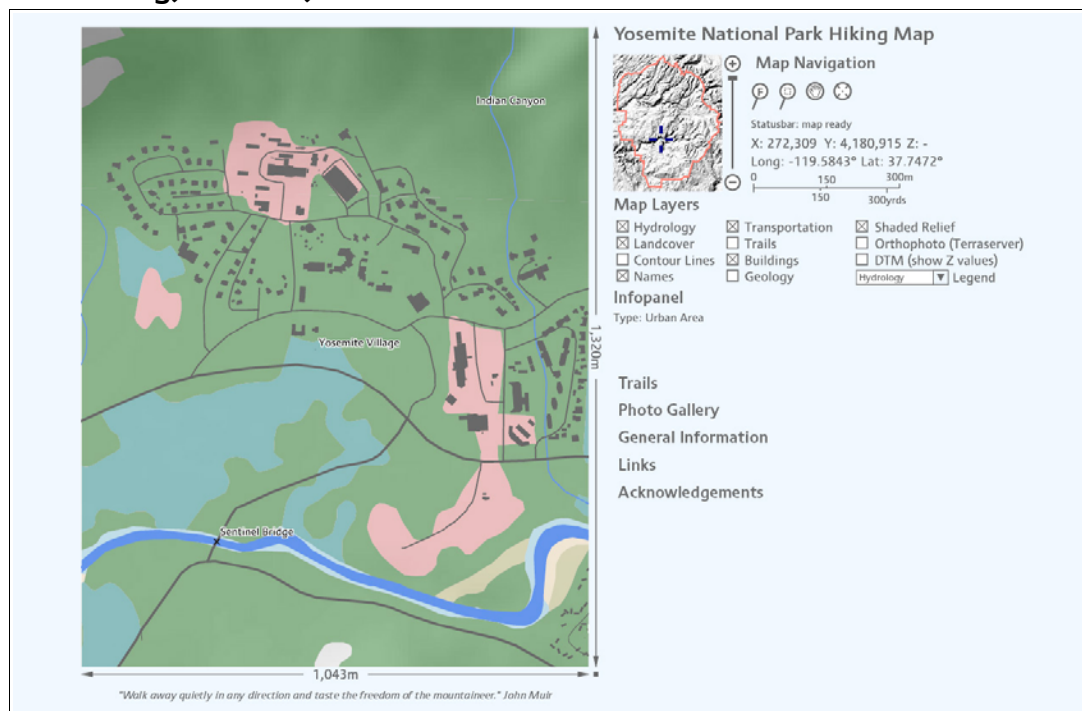


### Anlage 4: Orthophoto (detail) der Applikation





### Anlage 5: Ausschnitt aus der Applikation mit den Ebenen Hydrologie, Vegetation, Beschriftung, Verkehr, Gebäude und Relief



### Anlage 6: Ausschnitt um Half Dome mit den Ebenen Hydrologie, Höhenlinien, Wanderwege, Relief und DTM



## Anlage 7: Legenden

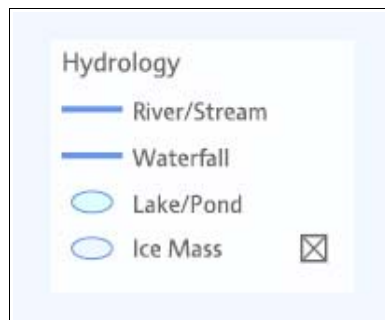


Illustration 54: Legende der Hydrologie



Illustration 55: Legende der Vegetation

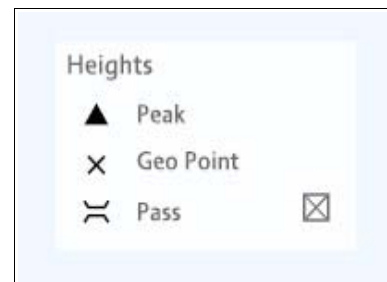


Illustration 56: Legende der Höhenkoten



Illustration 57: Legende des Verkehrssystems



Illustration 58: Legende der Geologie (generalisiert)



Illustration 59: Legende der Geologie (detailliert)



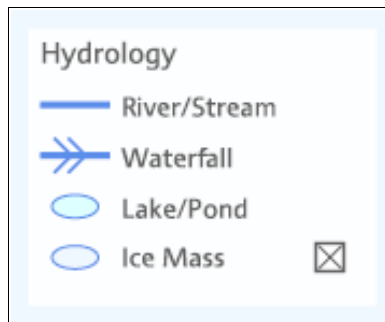


Illustration 60: Legende der Eigenschaften eines Wanderweges

## **9 Erklärung gemäss § 31 Abs. 5 RaPO**

Hiermit erkläre ich, dass ich die Diplomarbeit selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benutzt, sowie wörtliche und sinngemässe Zitate als solche gekennzeichnet habe.

Zürich, den 11.04.2005

Juliana Williams  
31.12.1979  
KA9 im WS 2004/2005