



Hochschule Neubrandenburg
University of Applied Sciences

Studiengang Geoinformatik

Visualisierung von Signalverläufen zur GNSS-Fernerkundung mit Google Earth

Bachelorarbeit

vorgelegt von: Maik Nergert

Zum Erlangen des akademischen Grades
"Bachelor of Engineering" (B.Eng.)

Erstprüfer: Prof. Dr.-Ing. Andreas Wehrenpfennig
Zweitprüfer: Dr. Ralf Stosius
Eingereicht am: 01.06.2010

urn:nbn:de:gbv:519-thesis2010-0135-9

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Bachelorarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Neubrandenburg, den 01.06.2010

Maik Nergert

Kurzfassung

Diese Arbeit befasst sich mit der Darstellung von Messungen aus der GNSS-Fernerkundung mit Hilfe von Google Earth. Es wird ein Visualisierungstool entwickelt, welches in der GNSS-Reflektometrie und Radio-Okkultation Anwendung findet. Die GNSS-Reflektometrie nutzt an der Erdoberfläche reflektierte GNSS-Signale als Messsignale. Dadurch können die Höhe und die Eigenschaften der Oberfläche ermittelt werden. Während einer Radio-Okkultation geht ein GNSS-Satellit am Horizont unter bzw. auf. Bei diesem Vorgang durchläuft das Radiowellen-Signal die Troposphäre und Ionosphäre und gibt dabei Aufschluss über den Zustand, z.B. Temperatur, Druck und Dichte, der Atmosphäre. Über die in dieser Arbeit entwickelte grafische Nutzerschnittstelle (Webinterface) werden vom Nutzer alle notwendigen Einstellungen getroffen und die verschiedenen Darstellungsmöglichkeiten gewählt. Die selektierten Daten werden in Google Earth mit Grafiken, 3D-Modellen und Linien dargestellt. Die Daten können darüber hinaus auch zu einer Animation zusammengefügt werden. Dieses Projekt dient der Planung und Veranschaulichung von Messkampagnen des GFZ Potsdam.

Abstract

This thesis addresses the visualization of GNSS remote sensing measurements with Google Earth. A visualization tool was developed to be used in the GNSS reflectometry and radio occultation. GNSS reflectometry uses the GNSS signals reflected by the Earth's surface to determine the height and properties of the surface. During an occultation a GNSS satellite sets behind or raises from the Earth's horizon. The radio wave signal crosses the troposphere and the ionosphere and gives information about the state like temperature, pressure and density of the atmosphere. A graphical user interface (web interface) developed will allow the user to change all settings and miscellaneous display options. The selected data are displayed with graphics, 3D models and lines in Google Earth. The data can also be joined to an animation. The animation can be played or saved as a video by the user. This project is used for planning and demonstration of measurement campaigns of the GFZ Potsdam.

Inhalt

1	Einleitung und Ziele	1
2	Grundlagen	3
2.1	GNSS	3
2.2	GNSS-Reflektometrie	4
2.3	Radio-Okkultation	5
2.4	weitere Messungen	5
3	Analyse und Planung	7
3.1	Anforderungen	7
3.1.1	Grundfunktionalitäten	7
3.1.2	sinnvolle Erweiterungen	8
3.1.3	Präsentation der Daten	8
3.2	Stand der Technologien	8
3.2.1	Google Earth	9
3.2.2	Google Earth API	10
3.2.3	Google SketchUp	10
3.2.4	3D-Tool	11
3.2.5	KML / WKML	11
3.3	Messdaten	12
3.3.1	Datei eines Senders	12
3.3.2	Datei eines Empfängers (Satellit)	13
3.3.3	Datei eines Empfängers (Bodenstation oder bodennah)	13
3.3.4	Qualitätsdaten	13
3.4	Entwurf	14

4	Umsetzung	17
4.1	Arbeit mit WKML	17
4.2	Funktionsweise des Programms	20
4.2.1	Ermitteln der Richtungswinkel	20
4.2.2	Verarbeiten und Anlegen der KML Datei	22
4.2.3	Darstellung von Reflexions- und Okkultationsmessungen	23
4.2.4	Die Tour	27
4.3	Parameter-Datei	28
4.4	Webinterface	31
5	Installation und Tests	34
5.1	Installation	34
5.2	Tests	35
6	Schlussbetrachtung	37
	Abkürzungsverzeichnis	39
	Quellenverzeichnis	40

1 Einleitung und Ziele

Für viele Menschen ist ein Navigationssystem im Auto oder sogar im Handy nicht mehr weg zu denken. Die Benutzung ist bereits zum Alltag geworden, auch wenn ein Großteil dieser Personen nicht weiß, wie die Positionsangabe in ihrem Navigationsgeräte zustande kommt. Einigen mag der Begriff "GPS" geläufig sein und wird automatisch mit einer Positionsbestimmung assoziiert. Doch das System kann noch viel mehr. Die Sektion 1.1 des deutschen GeoForschungsZentrums Potsdam befasst sich mit der GPS/Galileo-Erdbeobachtung und nutzt diese Satellitensignale für wissenschaftliche Zwecke. Die Satellitensignale werden beim Durchqueren der Atmosphäre verändert und es lassen sich Informationen über den Zustand der Atmosphäre gewinnen (Radio-Okkultation). Mit diesen Daten können die vertikalen Temperatur- und Wasserdampfverteilungen innerhalb der Atmosphäre bestimmt werden und tragen somit zur Verbesserung von globalen Wettervorhersagen bei [GFZAtmo2010]. Außerdem werden die Signale der GNSS-Satelliten von Wasser- und Eisflächen gut reflektiert und es lassen sich u.a. Rückschlüsse über Meeresspiegelhöhe und Windgeschwindigkeiten bzw. die Verbreitung von Meereis ziehen (GNSS-Reflektometrie und Scatterometrie). Wenn diese Daten schnell verarbeitet und übermittelt werden, können sie z.B. für ein Tsunami-Frühwarnsystem genutzt werden. Im Rahmen von GITEWS wird bereits eine Machbarkeitsstudie durchgeführt, die genau diesen Ansatz verfolgt. Aktuelle Simulationen zeigen, dass der Tsunami von 2004 mit einer Konstellation von 18 Satelliten innerhalb von 15 Minuten detektiert hätte werden können.

Eine weitere Studie ist der Kleinsatellit MicroGEM. Er dient der präzisen Orbitbestimmung mittels GNSS-, SLR- und VLBI-Messungen und wird sowohl in der Radio-Okkultation als auch in der Reflektometrie eingesetzt. Okkultationen und Reflexionen setzen jeweils eine bestimmte Konfiguration der sendenden GNSS-Satelliten und des GNSS-R Empfängers voraus. Der erfasste Atmosphären- oder Oberflächenbereich ändert sich sehr schnell auf globalem Maßstab. Deshalb ist eine Visualisierung sowohl der Satellitenbahnen als auch der Okkultations-/Reflexionsereignisse sehr wichtig, um einen Eindruck des jeweils erfassten Bereiches zu erhalten [GFZRefl2010]. Die genutzten Signale der Navigationssatelliten sind kostenfrei zu empfangen. Sie stehen an jedem Punkt der Erde und zu jedem Zeitpunkt zur Verfügung, da diese Systeme vor allem kommerziell und vom Militär genutzt

werden. Durch den ständigen Ausbau und Verbesserungen stehen in den nächsten Jahren durch die drei Systeme GPS, Galileo und GLONASS über 80 Satelliten zur Verfügung. Dies garantiert in absehbarer Zeit eine höhere Genauigkeit in der Altimetrie und bessere Ergebnisse in der Atmosphärensondierung.

Ziel dieser Arbeit ist es, für das GFZ Potsdam ein neues Programm zur Visualisierung für die GNSS-Fernerkundung zu erstellen. Es dient der Planung von Messkampagnen und zur Veranschaulichung des Arbeitsfeldes der Sektion 1.1. Außerdem soll ein grafisches Interface für das Programm erstellt werden, das es ermöglicht, das Programm mit wenig Aufwand zu bedienen und mit der offen zugänglichen Darstellungssoftware Google Earth ein schnelles Ergebnis zu präsentieren. Das Interface soll das entwickelte Programm mit der Konfiguration des Nutzers ausführen und anschließend die erzeugten Daten zurück liefern oder automatisch in Google Earth laden.

2 Grundlagen

2.1 GNSS

Das Globale Navigationssatellitensystem GNSS (Global Navigation Satellite System) setzt sich zusammen aus GPS, GLONASS und den im Aufbau befindlichen Galileo Navigationssystem. Alle GNSS Konstellationen sind so genannte Walker Konstellationen [Walker1984]. **GPS** ist eine Kurzform für das US-amerikanische NAVSTAR-GPS (NAVigation by Satellite Timing And Ranging - Global Positioning System), das seinen Ursprung als Ortungssystem des Militärs hat. Das GPS ist so konzipiert, dass es jederzeit verfügbar ist und somit den Empfang der Signale von mindestens vier Satelliten an jedem Punkt der Erde garantiert. Zurzeit befinden sich in der Erdumlaufbahn 30 Satelliten, wobei 24 für die flächendeckende Positionierung benötigt werden. Theoretisch können bis zu 32 Satelliten verwendet werden [GPSconst2009]. Die restlichen Satelliten dienen als Reserve, falls einer ausfällt, oder ältere werden zu Testzwecken weiter genutzt. Sie umkreisen die Erde in sechs Bahnen in einer Höhe von ca. 20.200 km (Abb. 2.1). Die Satelliten werden unterschiedlich nummeriert, z.B. mit der SVN (Space Vehicle Number) oder PRN (Pseudo Random Noise). Die PRN gibt die Zuordnung innerhalb der aktuellen Gesamtkonstellation an und wird neu vergeben, sobald ein Satellit nicht mehr aktiv ist. Es beschreibt den für jeden Satelliten individuellen Signalcode. Bei den Codes handelt es sich um so genannte GOLD-codes und sind eine mehr oder weniger willkürliche Abfolge von 1 und -1, die sich nach 1023 Schritten wiederholt. Während die Amerikaner ihr GPS entwickelten, hat Russland, damals UdSSR, ein eigenes Navigationssystem in den Orbit geschickt.

GLONASS (GLObalnaja NAWigazionnaja Sputnikowaja Sistema, Globales Satellitennavigationssystem) dient dem gleichen Zweck wie das amerikanische Gegenstück und besteht aus 21 funktionsfähigen Satelliten. Die vollständige Konstellation umfasst wie beim GPS 24 Satelliten und alle weiteren dienen der Ergänzung bzw. Aufrechterhaltung der Verfügbarkeit. 2010 ist geplant noch 7 weitere Satelliten zu starten, um das System vor Ausfällen zu schützen. Alle Satelliten umkreisen im Gegensatz zu GPS die Erde auf drei Bahnebenen in einer Höhe von ca. 19.100 km. Messungen haben ergeben, dass die Leistungsfähigkeit des GLONASS mit der des GPS verglichen werden kann.

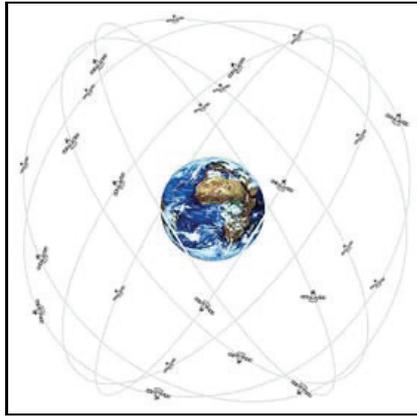


Abb. 2.1: Umlaufbahnen der GPS-Satelliten [PicGPS]

Das neue **Galileo** Satellitennavigationssystem befindet sich derzeit noch in der Planungs- und Testphase und ist ein Projekt der EU und der ESA. Geplant sind 30 Satelliten, wobei drei als Reserve dienen, die die Erde auf 3 Bahnen in einer Höhe von ca. 23.200 km umkreisen. Das System soll eine Echtzeit-Positionierungsgenauigkeit von 4 Meter erreichen und somit eine doppelte Genauigkeit im Vergleich zu GPS erzielen [Wickert2002].

2.2 GNSS-Reflektometrie

Die GNSS-Reflektometrie (kurz GNSS-R) nutzt an der Erdoberfläche reflektierte GNSS-Signale als Messsignale. GNSS-R ist eher ein Oberbegriff für GNSS-R Altimetrie und GNSS-R Scatterometrie. Dabei wird durch einen GNSS-R-fähigen Empfänger auf erhöhter Position, z.B. Gipfel, Flugzeug oder Satellit sowohl das direkte als auch das reflektierte GNSS-Signal empfangen. Durch den Vergleich der beiden Signale können die Höhe (Altimetrie) und die Eigenschaften (Scatterometrie) der Oberfläche ermittelt werden. Am besten für die Reflektometrie geeignet ist Wasser oder Eis. Wie bei der Positionsbestimmung beim GPS gibt es auch in der Reflektometrie zwei Methoden zur Höhenbestimmung. Bei der Codemessung werden die Laufzeitunterschiede des direkten und indirekten Signals gemessen und bei der Phasenmessung werden mittels Auswertung der Trägerphase die Interferenz der Phasenunterschiede beider Signale gemessen. Eine Phasenmessung findet nur bei sehr flachen Einfallswinkeln statt, weil dieses Verfahren nur mit kohärent reflektierten Signalen funktioniert. Die reflektierten Signale enthalten außerdem Informationen über die Ionosphäre und können auch zur globalen Ionosphärensondierung (siehe Kap. 2.3) verwendet werden [Komjathy1996]. Meeresspiegelhöhen werden mit Genauigkeiten im Zentimeter- bis Dezimeterbereich gemessen [Helm2008]. Aus diesem Grund läuft im Rahmen des GITEWS-Projekts bereits eine Studie zur Verwendung der GNSS-R bei der Tsunamidetektion [Stosius2010].

2.3 Radio-Okkultation

Während einer Okkultation wird ein Himmelskörper komplett durch einen anderen verdeckt. In der Radiookkultationsmessung bedeutet das, dass von einer erhöhten Position aus betrachtet ein Satellit hinter einem Himmelskörper verschwindet. Bewegt sich ein Satellit auf seiner Flugbahn hinter dem umkreisten Planeten (Erde) durchläuft das Radiowellen-Signal während des Eintauchens und beim Austritt aus der Verdeckung (vorstellbar wie Sonnenunter- und aufgang) die Troposphäre und Ionosphäre. Dabei wird die Welle gebrochen und phasenverschoben. Durch Beobachtung des direkten Signals lassen sich Faktoren, wie Temperatur, Druck und Dichte, in der oberen Troposphäre und Ionosphäre ermitteln [UNIBW2010].

In den letzten Jahren wurden verschiedene Satellitenmissionen zur atmosphärischen Fernerkundung durchgeführt. Die Okkultationsdaten von CHAMP, GRACE-A, COSMIC und MetOp werden bereits zur Verbesserung globaler Wettervorhersagen genutzt. Der CHAMP-Satellit hat bisher den längsten Datensatz geliefert, der zur Untersuchung klimatologischer Veränderungen in der Erdatmosphäre genutzt wird [Healy2007].

2.4 weitere Messungen

In dieser Arbeit wird das MicroGEM-Projekt als Beispiel herangezogen und diese Satellitenmission kann neben den GNSS-Methoden auch noch weitere Messverfahren durchführen. Um die aufgezeichneten Daten der GNSS-R und Radio-Okkultationen von einem Satelliten zur Bodenstation zu transferieren, wird ein **Downlink** verwendet. Die Übertragung der Daten kann beginnen, sobald ein Satellit auf seiner Umlaufbahn in den Empfangsbereich der zuständigen Station eintritt. Meist muss die Übertragung durch einen Operator ausgelöst und überwacht werden. Die Datenmenge und somit die Übertragungsdauer kann variieren.

SLR (Satellite Laser Ranging) ist eine präzise Methode in der Bahnbestimmung von Satellitenorbits. Mit der Messung der Laufzeit eines Laserstrahls wird die Entfernung zwischen einer Bodenstation (Abb. 2.2) und einem Satelliten gemessen. SLR dient der genauen Umlaufbahnbestimmung eines Satelliten oder einer Punktbestimmung auf der Erde. Bei bekannter Umlaufbahn eines Objektes können Änderungen des Erdkörpers und der Erdrotation festgestellt werden [Seeber1989].

VLBI (Very Long Baseline Interferometry) findet für hoch präzise Positionsbestimmungen Verwendung in der Radioastronomie. Ebenso wie bei SLR können die Messungen auch für die Erdvermessung genutzt werden. Durch präzises Erfassen der Signale von mindestens zwei Radioteleskopen (Abb. 2.2) und einer Zeitmarke kann eine Laufzeitmessung durchgeführt werden. Die gespeicherten Daten

werden solange korreliert, bis sie annähernd übereinstimmen. Das Ergebnis ergibt einen Wegunterschied vom anvisierten Objekt zu den Radioteleskopen. Die Genauigkeit beträgt dabei wenige Zentimeter.



Abb. 2.2: SLR-Station GFZ Potsdam [PicSLR], VLBI-Radioteleskop Effelsberg [PicVLBI]

Die parallele Verwendung beider Verfahren ermöglicht eine höhere Genauigkeit bei Bahnbestimmungen, da systematische Fehler der einzelnen Systeme aufgedeckt werden können. Zusammen mit GNSS, SLR und VLBI kann ein hoch genaues Referenzsystems realisiert werden. Das daraus entstehende ITRF (International Terrestrial Reference Frame) ist von großer Bedeutung bei der Erfassung kleinster Änderungen auf der Erde, z.B. dem globalen Meeresspiegelanstieg [MicroGEM2009].

Über eine GNSS-Zweifrequenzmessung durch die Ionosphäre kann entlang des Signalweges der integrierte Elektronengehalt (**TEC** - Total Electron Content) bestimmt und zur tomographischen Rekonstruktion der Elektronendichteverteilung über dem Satelliten verwendet werden. Der TEC setzt sich zusammen aus Elektronendichte und Weg, er wird in Elektronen pro Quadratmeter gemessen [Stolle2001]. Solche GNSS-TEC-Messungen sind bedeutend für klimatologische Untersuchungen und geben eine Auskunft über den aktuellen Zustands der Ionosphäre.

3 Analyse und Planung

3.1 Anforderungen

3.1.1 Grundfunktionalitäten

Die Hauptfunktion des zu entwickelnden Programms soll darin liegen, die Reflexionen von GNSS-Signalen und Okkultationen zu visualisieren. Da die Satelliten ständig in Bewegung sind, sollen die empfangenen Daten mit Datum und Uhrzeit versehen werden, damit sie später auseinander gehalten werden können. Die Animation dieser dynamischen Daten soll ein weiterer Schwerpunkt des Programms sein. Die grafische Darstellung in Form von Bildern oder 3D-Modellen der sendenden Satelliten, und besonders des Empfängers, ist für die Verständlichkeit und Anschaulichkeit der Animation maßgebend. Um einen Vergleich und die Kombination verschiedener Sender zu ermöglichen, soll das Programm unterschiedliche Technologien des GNSS verarbeiten. Darin eingeschlossen sind GPS, GLONASS und Galileo.

Ein GNSS-R-Empfänger kann auf verschiedenen Plattformen betrieben werden:

- Satelliten (z.B. CHAMP oder MicroGEM)
- Luftfahrzeuge (z.B. Flugzeug oder Zeppelin)
- Bodenstationen (z.B. an erhöhten Küsten von Meeren oder Seen)

Die Bedienung des Programms wird jedem Nutzer ob außenstehend oder projektintern so einfach wie möglich gemacht. Aus diesem Grund soll das Programm eine separate Konfigurationsdatei erhalten. Diese kann über die Linux-Konsole oder eine grafische Nutzerschnittstelle (GUI) erstellt werden. Um die Wartung und Erweiterung zu gewährleisten, muss das Programm strukturiert aufgebaut und übersichtlich sein. Die Nutzerschnittstelle zur Bedienung des Programms ist übersichtlich zu gestalten und leicht verständlich zu machen. Änderungen am Programm und der Nutzerschnittstelle sollen durch autorisierte Personen erfolgen.

3.1.2 sinnvolle Erweiterungen

Wenn sich der Empfänger auf einem Satelliten oder Flugzeug bzw. Luftschiff befindet, ist die Darstellung der Flugbahn sinnvoll. Zusätzlich zu der Animation der Signale und der Sender/Empfänger soll dem Betrachter die Möglichkeit gegeben werden aus einer Totalen, bei der sich der Empfänger im Fokus befindet, die aktuellen Ereignisse (Reflexionen oder Okkultationen) zu verfolgen. Die Reflexionen der unterschiedlichen PRN eines einzelnen GNSS sollen mit verschiedenen Farben dargestellt werden, um eine bessere Übersicht zu schaffen. Zur Unterscheidung der einzelnen GNSS der Systeme GPS, GLONASS oder Galileo wird alternativ jedem Satelliten eine andere Farbe zugewiesen. Die verschiedenen PRN bzw. GNSS werden in einer Legende aufgelistet. Weitere Messungen von MicroGEM, wie TEC, VLBI, SLR und der Datentransfer, soll der Nutzer nach Bedarf anzeigen lassen können. Eine andere Möglichkeit ist die Auswertung von Qualitätsdateien. Diese beinhalten Angaben zum Ausschluss bestimmter Reflexionen, die anhand bestimmter Kriterien im Vorfeld selektiert werden. Die Kompatibilität zu anderen Messkampagnen soll gegeben sein, damit für neue Projekte nicht extra ein neues Programm entwickelt werden muss. Da die Darstellungen von Reflexionen und Okkultationen auf Linien beruhen, sollen diese in der Strichstärke angepasst werden können.

3.1.3 Präsentation der Daten

Der Grundgedanke zur Präsentation der Daten ist die Ausgabe in einem Format, das jede Plattform, ob Windows, Linux oder Mac, lesen kann. Eine Erweiterung dazu wäre die Darstellung der Daten direkt in der Nutzerschnittstelle ohne die Installation einer speziellen Software. Zum Versenden, Präsentieren in Vorträgen oder Einbauen der animierten Daten in eine Internetseite wäre die Möglichkeit zum Erstellen von Videos eine gute Alternative.

3.2 Stand der Technologien

Google Earth ist eine freie Software, mit der sich schnell eigene Daten auf einen digitalen Globus darstellen lassen. Sie bietet eine offene Schnittstelle über das KML-Dateiformat an. Das Format ist angelehnt an XML, es ist logisch aufgebaut und bietet einen großen Funktionsumfang. Google Earth ist für jede Plattform frei verfügbar. Daraus ergibt sich die Möglichkeit die erstellten Visualisierungen an jeden beliebigen Nutzer weiter zu geben. Außerdem lassen sich die Dateien mittels Google Earth API in eine Internetseite einbinden und können somit einem breiten Publikum zur Verfügung gestellt werden. Für die originalgetreue Darstellung von Objekten, z.B. Satelliten, können 3D-Modelle aus CAD-Anwendungen oder anderen Modellierungsprogrammen geladen werden.

Dies sind die Hauptgründe, warum Google Earth für die Visualisierung der Messdaten ausgewählt wird. Folglich müssen alle anderen Komponenten (Tab. 3.1) mit Google Earth kompatibel sein. Um ein 3D-Datenformat zu erhalten, das Google Earth verarbeiten kann (COLLADA - XML basiertes Austauschformat für 3D-Programme), wird SketchUp verwendet. Es lassen sich Modelle erstellen, bearbeiten und in ein dae-Format speichern. Fertige Modelle können direkt nach Google Earth exportiert werden. Modelle, die SketchUp nicht importieren kann, müssen im Vorfeld in ein unterstütztes Format konvertiert werden, z.B. 3ds (Dateiformat 3D Studio Max). Das Programm 3D-Tool ist eine kommerzielle Lösung. Es kann alle gängigen Formate lesen und in ein anderes speichern. Die Schnittstelle, die Google Earth für das Laden von Daten verwendet, ist das KML-Format. Um die am GFZ viel genutzte Programmiersprache Fortran das Erstellen von KML-Dateien zu ermöglichen, wird die freie WKML-Library genutzt.

Tab. 3.1: Übersicht verwendete Software Versionen

Software	Version
Google Earth	5.1
Google Earth API	1
Google SketchUp	7.1
3D-Tool	9 (Trial)
KML	2.2
WKML	RC2

3.2.1 Google Earth

Es können Satellitenbilder, Karten, Geländeinformationen und 3D-Modelle von Gebäuden betrachtet werden. In einigen Ländern der Erde können über die Street-View-Funktion reelle Umgebungen angesehen werden. Hierzu werden Straßen abgefahren und mit einer 360°-Kamera Aufnahmen gemacht. Seit der Google Earth Version 5 lassen sich entfernte Galaxien sowie Mond und Mars betrachten. Es können nicht nur Landmassen erkundet werden, sondern auch der Unterwasserbereich der Ozeane. Für die Darstellung der Erdoberfläche liegen nicht nur aktuelle Satellitenbilder vor, sondern auch die vorherigen Versionen und sogar historisches Bildmaterial. Die Entwicklung eines Gebietes kann somit ohne Probleme nachvollzogen werden. Um die Suche eines Ortes zu erleichtern, bietet Google Earth eine umfangreiche Suchfunktion. Es können Orte, Postleitzahlen, Straßennamen oder die Koordinaten eines Punktes gesucht werden. Mittels Ortsmarke kann das Suchergebnis gespeichert werden, um den Ort zu einem späteren Zeitpunkt schneller zu finden. Die Ortsmarken können mit Kommentaren versehen und an andere Nutzern verschickt werden. Ein weiteres Werkzeug ist die Messfunktion. Es lassen sich Wegunterschiede zwischen zwei Punkten oder ganze Pfade messen.

Durch die umfangreichen Darstellungsmöglichkeiten können, z.B. Wetterkarten oder sogar Radarfallen geladen und in der Karte angezeigt werden. Die kostenpflichtige Version Google Earth Pro bietet zudem die Möglichkeit, GIS-Dateien und Tabellen zu importieren. Diese umfangreichen Datensätze können hochauflösend gedruckt oder es kann über die MovieMaker-Funktion ein Videoclip erstellt werden [GoogleEarth2010].

3.2.2 Google Earth API

Das bekannte Programm Google Earth hat sich in kurzer Zeit zu einem beliebten Tool für Heimwerker und im kommerziellen Bereich entwickelt. Dieses Tool kann auch ohne Installation auf dem PC in die eigene Website eingebaut werden, nämlich mit der Google Earth API. Mit dem Google Earth Plug-In wird der digitale Globus in die eigene Website integriert. Mittels der API können Markierungen und Linien eingetragen, Bilder über das Gelände drapiert, 3D-Modelle hinzugefügt oder über eine importierte KML-Datei komplexe Kartenanwendungen generiert werden. Die Integration in die eigene Internetseite ist für nicht kommerzielle Zwecke kostenlos und erfordert lediglich die Registrierung der Website-URL. Im Anschluss wird eine Quellcode-Zeile eingeblendet, die in die Website eingefügt werden muss [EarthAPI2010].

3.2.3 Google SketchUp

Die kostenlose Version von Google SketchUp ist eine leistungsfähige, professionelle 3D-Software zum Konstruieren von Objekten, wie Häusern, Flugzeugen oder Landschaften. Diese Objekte werden mit Hilfe einfacher Zeichenwerkzeuge, wie Linie, Rechteck, Kreis oder Bogen gezeichnet. Die Hilfefunktion ermöglicht Anfängern den schnellen Einstieg in das Arbeiten mit SketchUp. Eine Schnellhilfe zeigt zu jedem gewählten Zeichenwerkzeug eine Beschreibung und eine Schritt für Schritt Anleitung zur Handhabung an. Die Einarbeitung in die verschiedenen Tools gelingt schnell und motiviert zum Weitermachen. Neben dem Zeichnen sind Bemaßungen ebenso möglich wie dreidimensionale Beschriftungen oder das Skalieren, Drehen oder Verschieben von Objekten. Zusätzlich zu Zeichen- und Bearbeitungswerkzeugen bietet das Programm auch vorgefertigte Komponenten in den Bereichen Architektur, Personen, Landschaft, Transport und andere. Es können anhand von importierten Fotos Objekte nachmodelliert und diese im Anschluss nach Google Earth exportiert werden. Es gibt auch die Möglichkeit, sich die Werke anderer Personen anzusehen und weiter zu bearbeiten. Dazu bietet Google für SketchUp eine gratis Online-Datenbank an, in der jeder Nutzer seine Modelle zur Verfügung stellen kann [Sketchup2010].

3.2.4 3D-Tool

3D-Tool ist ein dynamischer Viewer für 3D und 2D CAD-Dateien, der es ermöglicht viele gängige 3D und 2D CAD-Dateiformate zu importieren. Er bietet die Möglichkeit grafische Analysen und Visualisierungen von CAD-Projekten durchzuführen. Baugruppen, Explosionszeichnungen und Animationen können mit einer sehr einfachen Schnittstelle erstellt werden. 3D-Tool erzeugt auch ausführbare Präsentations-Dateien von CAD-Projekten, die per E-Mail verschickt und auf jedem PC ohne Installation oder zusätzliche Lizenzierung angeschaut werden können. Die Benutzeroberfläche erlaubt es, ohne spezielle CAD-Kenntnisse die Modelle zu betrachten, in Details zu zoomen oder einfach in die Modelle hineinzuschauen. Ansichten aus jedem Winkel können ausgedruckt oder als Bild gespeichert werden [3DTool2010].

3.2.5 KML / WKML

KML (Keyhole Markup Language) baut auf die XML-Syntax auf und dient der Darstellung strukturierter Daten in Form von Textdateien. Es lassen sich grundlegende Visualisierungselemente, wie z.B. Linien, Punkte, Modelle oder Bilder, speichern und in Google Earth und Google Maps anzeigen. Diese Elemente lassen sich mit Zeitinformationen und Beschreibungen versehen. KML-Dateien können zum Austausch von Informationen an jeden Nutzer, der Google Earth oder Google Maps verwendet, weiter gegeben werden. Die Verarbeitung der KML-Dateien erfolgt ähnlich wie ein HTML-Dokument im Browser. Sie wird sequenziell ausgelesen und die Tag-basierte Struktur mit Namen und Attributen ausgewertet [GEMan2010].

WKML ist ein Ausgabemodul, das auf FoX (Fortran Projekt um das Erstellen von XML Dateien zu vereinfachen) aufbaut. Es wurde dazu entwickelt KML-Dateien mit Fortran zu erstellen, um die einfache Visualisierung von Geodaten, die mit Fortran Programmen erzeugt wurden, zu ermöglichen. Viele häufig verwendete wissenschaftliche Programme im Geodatenbereich sind in Fortran geschrieben und eine Menge Geowissenschaftler programmieren in Fortran, weil diese Sprache nach wie vor eine der schnellsten Verarbeitungsmethoden für große Datenmengen ist. Die Visualisierung der Daten in Google Earth erfordert die Verwendung von KML, welche für viele Geowissenschaftler ungewohnt ist und für die keine bestehende Fortran-Schnittstelle existiert. Die WKML-Bibliothek bringt die Leistung von Google Earth in die Reichweite der Fortran-gebildeten Wissenschaftler und ermöglicht mit vorhandenen Fortran-Code und einem geringen Aufwand die Ausgabe von Daten in Google Earth [WKML2010].

3.3 Messdaten

Alle Daten, die in diesem Projekt verarbeitet werden, stammen aus realen Messungen oder wurden unter realistischen Bedingungen simuliert. Die vorliegenden Daten befinden sich alle im ASCII-Format und sind leicht verständlich aufgebaut. Sie können mit jedem beliebigen Texteditor geöffnet und leicht mit Fortran eingelesen werden. Im Folgenden werden die Inhalte der einzelnen Spalten und ihre Funktionen aufgelistet. Für jedes Projekt gibt es Daten über die Sender, z.B. GPS-Satelliten, den Empfänger und ggf. noch zusätzliche Informationen über die Verwendbarkeit der Reflexionen. Alle Uhrzeiten haben das Format *Stunde Minute Sekunde*. Die Positionen beziehen sich auf die geographische Breite und Länge, die Höhenangaben liegen in Meter über Meeresspiegel vor. Einige Beispiele enthalten den Eintrag NaN, dieser ist ein Platzhalter für Daten die für diesen Zeitraum nicht vorhanden sind.

3.3.1 Datei eines Senders

Spalten	Funktion
1-3	Uhrzeit
4	PRN
5-7	Position Sender
8-10	Position der Okkultation
11-13	Position der Reflexion
14-16	Position Empfänger (für TEC-Messung, nur bei Satelliten)

Beispiel

```
9 22 47 4 -37.12545182 -137.78964726 20200000 38.02442353 -150.37972140 105356.38 36.82533526 -150.35375158 0 NaN NaN NaN
9 22 48 4 -37.11977035 -137.78606629 20200000 38.03865671 -150.34993922 103284.22 36.86092222 -150.32931577 0 NaN NaN NaN
```

3.3.2 Datei eines Empfängers (Satellit)

Spalten	Funktion
1-3	Uhrzeit
4-6	Position Empfänger
7-9	Position VLBI-Station 1
10-12	Position VLBI-Station 2
13-15	Position SLR-Station
16-18	Position Downlink-Station

Beispiel

```
9 36 23 69.708266 -3.042156 700000 49.145199 12.877275 651 57.39306 11.91776 58 52.38302 13.06143 79 78.91785 11.92344 41
9 36 24 69.647957 -3.001581 700000 49.145199 12.877275 651 57.39306 11.91776 58 52.38302 13.06143 79 78.91785 11.92344 41
```

3.3.3 Datei eines Empfängers (Bodenstation oder bodennah)

Spalten	Funktion
1-3	Uhrzeit
4-6	Position Empfänger

Beispiel

```
6 32 37 47.67580584 9.51293498 508
6 32 38 47.67587133 9.51300237 510
```

3.3.4 Qualitätsdaten

Spalten	Funktion
1-3	Uhrzeit
4-35	Anzahl variiert je nach System, jeder Satellit hat eine Spalte

Beispiel

```
7 6 17 NaN NaN NaN 0 NaN NaN NaN NaN NaN NaN 1 0 0 0 NaN NaN 1 NaN 0 1 NaN NaN 1 NaN NaN NaN NaN 0 NaN NaN 0 1
7 6 18 NaN NaN NaN 0 NaN NaN NaN NaN NaN NaN 1 0 0 0 NaN NaN 1 NaN 0 1 NaN NaN 1 NaN NaN NaN NaN 0 NaN NaN 0 1
```

3.4 Entwurf

Bisher wurde geklärt, was das System leisten soll und welche Software und Daten dafür zur Verfügung stehen. Im nächsten Schritt ist zu klären, wie die Messdaten veranschaulicht und in Google Earth angezeigt werden können. Nach dem EVA-Prinzip (Einlesen, Verarbeiten, Ausgeben) werden die Messdaten aus den ASCII-Dateien zusammen mit Zeit und Darstellungsoptionen in das KML-Format gebracht. Abb. 3.1 zeigt ein Beispiel, wie Reflexionen eines einzelnen GNSS-Satelliten vom Programm verarbeitet werden. Im Vorfeld wird bereits die Datei des Empfängers eingelesen.

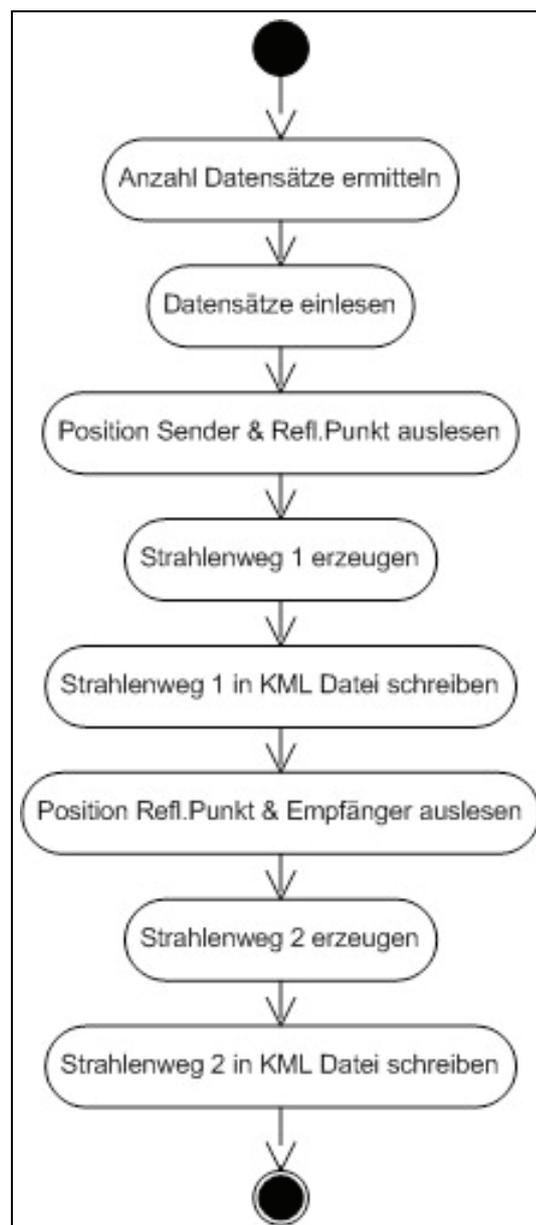


Abb. 3.1: UML-Aktivitäts-Diagramm - Verarbeiten von Reflexionen

Für die Darstellung von Positionen stehen in Google Earth Punkte und Linien zur Verfügung. Da die Messungen immer einen Weg, z.B. zwischen einem Sender und einem Reflexionspunkt, darstellen wird die Funktion zur Anzeige von Punkten dafür genutzt, die Position der Sender und Empfänger in Google Earth darzustellen. Der Strahlenweg der Radiosignale müsste wegen der Refraktivität der Atmosphäre leicht gekrümmt dargestellt werden. Da eine gekrümmte Linie in Google Earth nicht umgesetzt werden kann, werden die Signale vereinfacht mittels gerader Linie visualisiert. Damit die Sender und Empfänger nicht nur mit einem Punkt dargestellt werden, sollen diese als Objekte visualisiert werden. 3D-Modelle haben den Vorteil, dass das Objekt von jeder Seite betrachtet werden kann und somit eine originalgetreue Abbildung möglich ist. Einfache Grafiken zeigen nicht alle Details eines Objektes und durch die statische Darstellung ist es dem Betrachter nicht möglich, durch Ändern des Blickwinkels einen anderen Teil des Objektes anzuschauen. Aus diesen Gründen wird die Verwendung von 3D-Modellen priorisiert. Modell-Dateien aus CAD-Anwendungen, z.B. das step-Format, können von Google SketchUp nicht geladen und somit auch nicht in Google Earth angezeigt werden. Deswegen werden mit dem 3D-Tool step-Dateien in das 3ds-Format konvertiert. Die darzustellenden Strahlenwege der unterschiedlichen GNSS werden durch verschiedene Farben von einander getrennt.

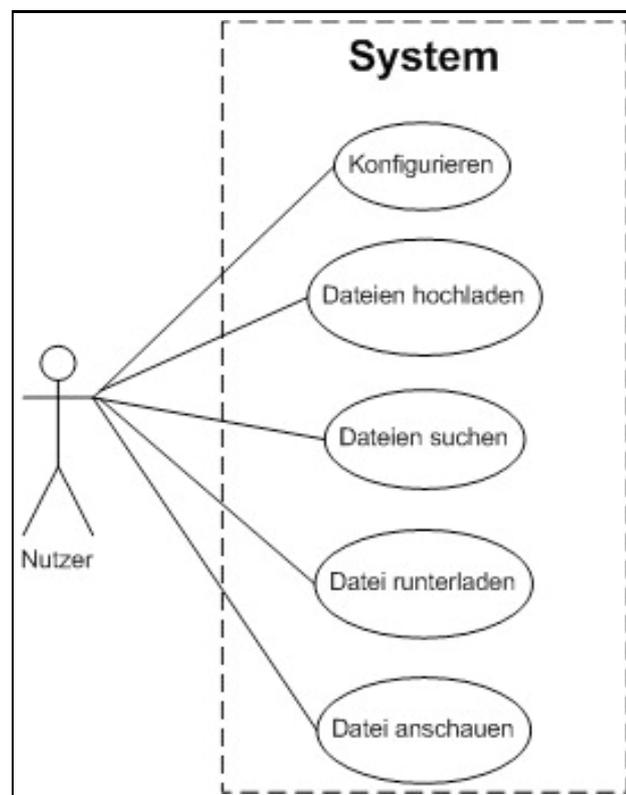


Abb. 3.2: UML-Anwendungsfall-Diagramm - Mögliche Aktionen des Nutzers

Die Messdaten werden eingelesen und der Zeitpunkt und die Positionen zu einen darstellbaren Signalweg verarbeitet. Im Anschluss werden die Informationen in die KML-Datei geschrieben. Dem Nutzer stehen verschiedene Möglichkeiten zur Verfügung das System zu bedienen (Abb. 3.2). Für die Konfiguration des Programms wird eine Parameter-Datei bereit gestellt oder über ein Interface automatisch erzeugt. Die Bedienung soll über die Linux-Konsole oder über ein Webinterface erfolgen. Das Interface bietet die Möglichkeit eigene Daten hochzuladen. Das Ergebnis der Prozessierung wird dem Nutzer direkt angezeigt oder steht zum Download bereit. Eine Übersicht des Gesamtsystems bietet Abb. 3.3. Es zeigt alle Eingangsdaten, die Ausgabedatei und die Applikationen, die für die Bedienung und Anzeige der Ergebnisse verwendet werden. Messdaten sowie einige Grafiken und Modelle werden vom GFZ bereit gestellt. Die Grafiken werden ggf. im Vorfeld bearbeitet und mit einem transparenten Hintergrund versehen. Alle 3D-Modelle werden in Google SketchUp an den drei Drehachsen ausgerichtet und für die Nutzung in Google Earth vorbereitet. Das Programm liest die Konfigurationsdatei ein, verarbeitet die Messdaten und bindet die passenden Grafiken/Modelle ein. Die Ausgabedatei kann dann in Google Earth oder über die API geladen werden.

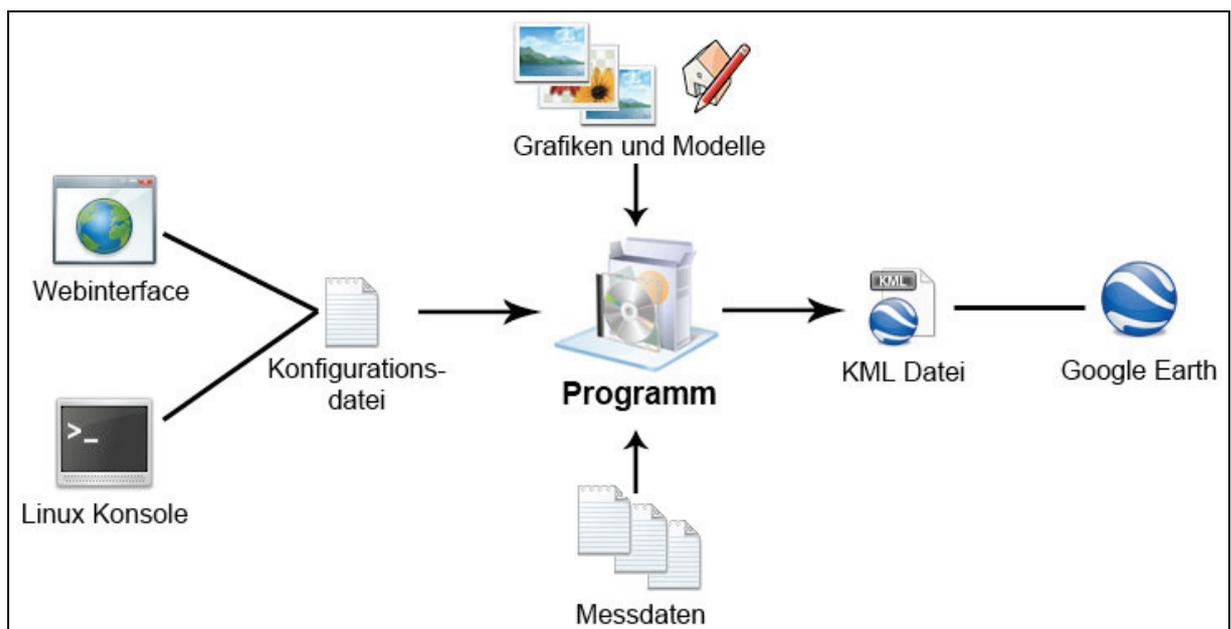


Abb. 3.3: Funktionsweise des Systems mit Input- und Output-Daten

4 Umsetzung

4.1 Arbeit mit WKML

Um alle Anforderungen an das Programm erfüllen zu können, muss der RC2 des WKML-Projektes um einige Funktionen erweitert werden. Alle notwendigen Informationen und XML-Syntaxe wurden der Google KML-Referenz entnommen. Damit überhaupt eine Animation der Datensätze möglich ist, wird ein Zeitintervall benötigt. Zeitpunkte und Zeitabschnitte kann die WKML-Library nicht verarbeiten, daher werden diese zwei Funktionen als Erstes implementiert. Der Zeitstempel repräsentiert einen einzigen Moment und wird durch eine *dateTime* im XML-Format angegeben. Die Genauigkeit ist abhängig von der Eingabe des Nutzers.

Hier zwei Beispiele:

YearMonth (YYYY-MM): Die Genauigkeit der Zeitangabe bezieht sich auf einen Monat

```
<TimeStamp><when>2010-03</when></TimeStamp>
```

dateTime (YYYY-MM-DDThh:mm:sszzzzz): Dieser Zeitstempel ist auf die Sekunde genau (lokale Zeit) und kann in die UTC Zeit konvertiert werden.

```
<TimeStamp><when>2010-03-16T10:43:17+01:00</when></TimeStamp>
```

Die Angabe der Zeitspanne zwischen zwei Ereignissen erfolgt ähnlich und besitzt die gleichen Genauigkeiten. Durch Angabe eines Beginns wird der Startpunkt festgelegt und der Endpunkt wird durch die End-Variable definiert. Wenn bei der Verwendung von Zeitspannen der Anfang oder das Ende nicht gesetzt wird, dann ist ein Objekt von Anfang an oder bis zum Ende der Animation sichtbar. Um die Erstellung einer Tour zu ermöglichen, wird die Library um zwei grundlegende Funktionen erweitert. Als Erstes "FlyTo" und für den richtigen Blickwinkel die Funktion "LookAt". FlyTo spezifiziert einen Punkt im Raum, zu dem geflogen werden soll. Zu den weiteren Angaben gehört die Flugdauer von einem Punkt zu dem anderen und der Flugmodus. Hierbei wird unterschieden zwischen Smooth (ruhig) und Bounce (springend). Die LookAt-Funktion ist das Kernelement des Ganzen. Sie definiert, wohin während des Fluges geblickt wird (Abb. 4.1). Es ist eine Art virtuelle Kamera, die

das Geschehen verfolgt. Hierbei werden die dreidimensionale Position des Objektes, das betrachtet werden soll, die Schrägdistanz und die beiden Blickwinkel übergeben.

Beispiel:

```
<LookAt>
  <longitude>-122.36415</longitude>
  <latitude>37.824553</latitude>
  <altitude>0</altitude>
  <range>150</range>
  <tilt>50</tilt>
  <heading>0</heading>
</LookAt>
```

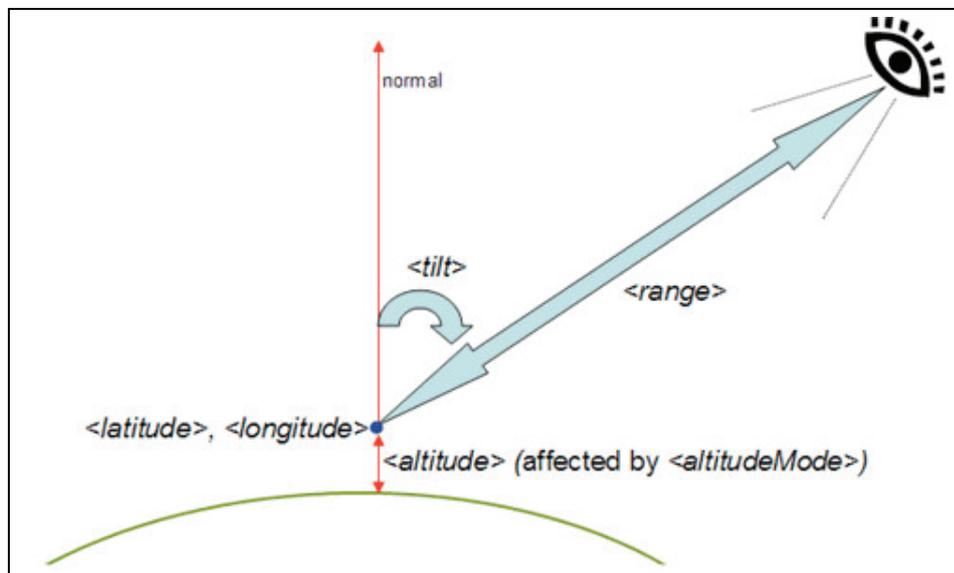


Abb. 4.1: Veranschaulichung der LookAt-Funktion [PicKMLRef]

Damit als Sender oder Empfänger nicht nur Grafiken, sondern auch 3D-Modelle genutzt werden können, hat die WKML-Library zusätzliche Erweiterungen erhalten. Die Model-Funktion beinhaltet die Position des Objektes, drei Drehwinkel, Skalierfaktoren und den Pfad zu dem Modell. In Zusammenarbeit mit Zeitstempel oder Zeitspanne können die LookUp- und Model-Funktion ihre Position und Richtungen jede Sekunde ändern und gewährleisten somit eine sehr dynamische Animation der Daten.

Für das Einbinden von Logos, Legenden oder Karten in Google Earth gibt es zwei Möglichkeiten. Einmal als ScreenOverlay oder als GroundOverlay. Das ScreenOverlay befindet sich statisch an einer

Position auf dem Bildschirm, während dessen das GroundOverlay an einem festen Punkt auf der Erde liegt. Der Aufbau beider Funktionen ähnelt sich etwas. Beide benötigen den Pfad zu einer Grafik und die Position, wo diese Grafik zu sehen sein soll. Unterschiedlich ist die Positionsangabe. Beim GroundOverlay wird eine Box definiert, die die Eckpunkte der Grafik bestimmt. Das ScreenOverlay definiert die Position am Bildschirm und kann, z.B. in Pixel angegeben werden.

Beispiel Position GroundOverlay:

```
<LatLonBox>
  <north>55</north>
  <south>47</south>
  <east>6</east>
  <west>15</west>
  <rotation>0</rotation>
</LatLonBox>
```

An der WKML-Library werden weitere kleine Änderungen vorgenommen um Fehler zu beseitigen oder den KML-Konventionen gerecht zu werden, z.B. definierte Integer-Variablen zu Real geändert. Alle oben erklärten neuen Funktionen enthalten wiederum andere Funktionen (Verschachtelungen), die teilweise vorhanden sind oder neu erstellt werden. Am Ende der Kette befindet sich die "lowlevel" Datei, die alle Grundfunktionen beherbergt. Sie enthält Subroutinen, die den korrekten XML-Syntax in die KML-Datei schreiben. In der lowlevel-Datei werden ca. 30 neue Subroutinen erstellt, die das Erzeugen der KML-Datei vereinfachen sollen.

Beispiel:

```
subroutine kmlCloseObject(xf)
  type(xmlf_t), intent(inout) :: xf
  call xml_EndElement(xf,'Object')
end subroutine kmlCloseObject
```

Diese Anweisung schreibt ein "</Object>" in die KML-Datei.

4.2 Funktionsweise des Programms

Das Programm hat die Hauptaufgabe Reflexionen und Okkultationen darzustellen. Zusätzlich werden Objekte, Grafiken und andere Messungen eingebunden, visualisiert und in einer Animation anschaulich dargestellt. Wegen seiner Funktion und einem Wiedererkennungswert wird das Programm ReflOcc's getauft.

Zu Beginn des Programms werden externe Module geladen, wie z.B. WKML und Transformations-Libraries, im Anschluss werden alle notwendigen Variablen deklariert und einigen Arrays ihre Größe zugewiesen. Die Parameter-Datei, die der Nutzer erstellt hat, wird anschließend eingelesen, alle Einträge ausgewertet und auf Variablen abgespeichert. Zur Auswertung gehört u.a. die De-/Aktivierung von bestimmten Visualisierungen, die Wahl des Empfängers oder welches GNSS genutzt wird. Es folgt das Einlesen der ersten Datei. Die Datei des Empfängers wird analysiert und die Anzahl der Datensätze dazu genutzt, die Größe für nachfolgende Arrays festzulegen. Alle Inhalte der Datei werden in einem Array abgelegt und falls für diesem Empfänger eine Quality-Datei vorhanden ist, wird diese ebenfalls eingelesen und zwischengespeichert.

4.2.1 Ermitteln der Richtungswinkel

Die Richtungswinkel werden im späteren Verlauf für die Ausrichtung von 3D-Modellen und der Kamera benötigt. Die Angaben der Richtung (heading) in Google Earth beziehen sich auf den Nordpol (Abb. 4.2).

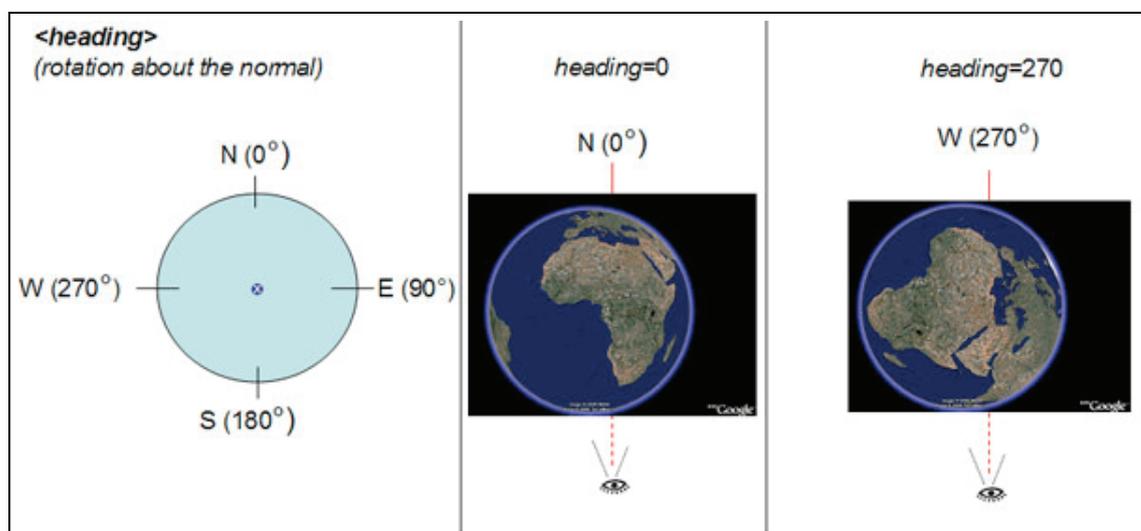


Abb. 4.2: Veranschaulichung der Blickrichtung in Google Earth [PicKMLRef]

Wenn der Empfänger um die Erde kreist, ändert sich der Winkel des Objektes und somit auch der Winkel für die spätere Kameraführung in Bezug auf dem Pol (Abb. 4.3). Es handelt sich hierbei

um den Azimut. In dem Beispiel wird der Winkel der Kamera (K) auf das Objekt (O) von Position 1 nach Position 3 immer kleiner. Der Winkel zu einem Objekt in der Ebene wird vom Norden in Richtung Osten gezählt und muss für jede neue Position des Empfängers ermittelt werden.

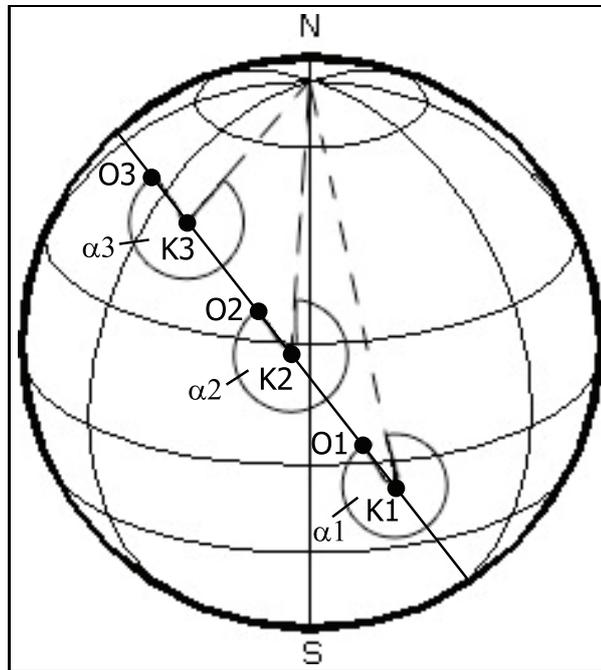


Abb. 4.3: Änderung des Azimuts beim umkreisen der Erde

Der Azimut wird mit Hilfe zweier bekannter Punkte berechnet. Hierzu wird die aktuelle Position des Empfängers und sein vorheriger Standort verwendet. Um bei schnellen Flugobjekten einen möglichst genauen Azimut zu erhalten, wird bei großen Positionsunterschieden der letzte Standort mittels linearer Interpolation dichter an die aktuelle Position heran geführt. Die Berechnung des Azimuts erfolgt über ein bereits vorhandenes Fortran-Modul. Dazu werden die zwei bekannten Koordinaten von WGS84 in das Kartesische Koordinatensystem und von diesem in ein lokales Horizontsystem überführt. Aus den ursprünglichen zwei Positionen des raumfesten Systems werden nun zwei neue Positionen in das erdfeste System transformiert. Der Ursprung wird vom Geozentrum in das lokale System verschoben und die Orientierung zwischen beiden System wird angepasst. In dem lokalen Horizontsystem können nun mittels weiterer Funktion die astronomischen Richtungen, die Zenitdistanz und der Azimut, bestimmt werden. Die Zenitdistanz bzw. als Gegenstück die Elevation spielt im weiteren Verlauf des Programms keine Rolle und ist somit zu vernachlässigen. Um später Sprünge in der Kameraführung, diese können beim Überqueren von West/Ost Längengraden auftreten, zu vermeiden, werden Winkelkorrekturen eingebaut.

4.2.2 Verarbeiten und Anlegen der KML Datei

Hier beginnt die eigentliche Verarbeitung der Messdaten und Positionsdaten. Mittels Befehl aus der WKML-Library wird eine KML-Datei angelegt und ein Header geschrieben. Dieser enthält eine Übersicht der verwendeten XML- und KML-Versionen. Anschließend werden alle benötigten Styles angelegt und mit einer ID zur späteren Identifizierung versehen. Die Styles enthalten Pfade zu Grafiken, Farben und Strichstärke für Reflexionen, Okkultationen, TEC, VLBI, SLR und Downlink. Zusätzlich werden 32 verschiedene Farben definiert, die den Satelliten PRN zugeordnet werden kann. Während der Verarbeitung und Erstellung der KML-Datei muss darauf geachtet werden, dass Google Earth die KML-Datei von oben bis unten sequentiell verarbeitet und nachträglich keine Änderungen mehr erfolgen können. Daher werden alle Schritte in Ordner geschachtelt, um die Daten übersichtlich verwalten zu können (Abb. 4.4).

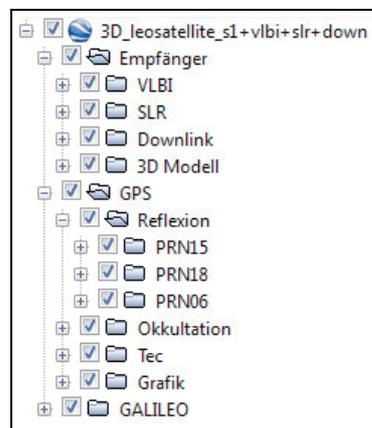


Abb. 4.4: Ordnerstruktur in der Google Earth Seitenleiste

Als Erstes werden alle Messungen des Empfängers verarbeitet. Je nach Einstellung des Nutzers wird VLBI, SLR und Downlink ausgewertet. Das Schema der Datenverarbeitung ähnelt sich und es wird nur einmal für alle drei Visualisierungen erklärt. Alle drei Varianten erhalten eigene Unterordner. Die bereits eingelesene und auf einem Array gespeicherte Empfänger-Datei wird zeilenweise ausgelesen und geprüft, ob eine Information für die VLBI-, SLR- und Downlink-Darstellung (Abb. 4.5) vorhanden ist. Falls eine Information vorhanden ist, wird der Zeitpunkt der Messung/Datenübertragung ausgelesen und als Anfangszeit definiert. Die Endzeit ergibt sich aus der zeitlichen Auflösung der Daten plus der Anfangszeit. Durch die Zeitinformation ist es dann möglich, die Daten über einen längeren Zeitraum zu animieren. Die Position des Empfängers und der Bodenstation, die Zeitspanne und die Style-ID werden einer WKML-Funktion zum Zeichnen von Linien übergeben. Diese verbindet beide Punkte mit der entsprechenden Farbe und Strichstärke.

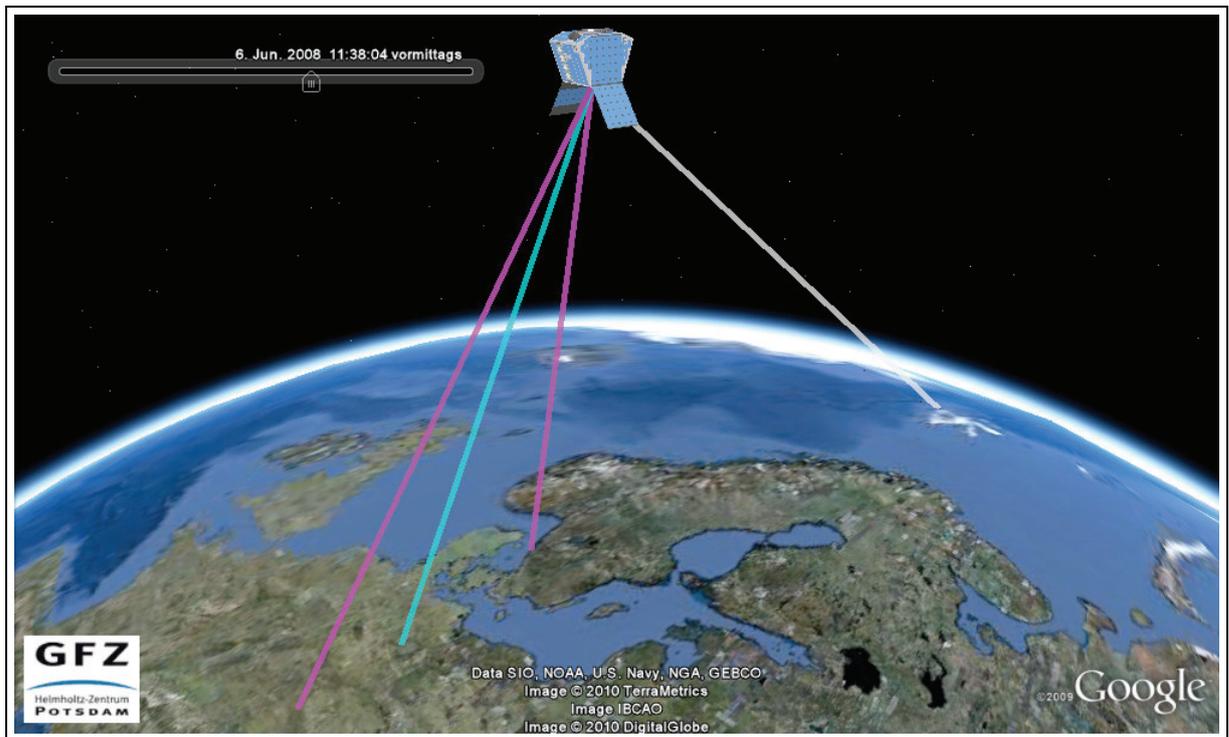


Abb. 4.5: VLBI (magenta), SLR (cyan), Downlink (grau)

Der Empfänger wird als Grafik oder 3D-Modell dargestellt, zusätzlich kann seine Flugbahn angezeigt werden. Für jeden Datensatz, der sich in der Empfänger-Datei befindet, wird ein Bild oder ein Modell eingebunden. Dies geschieht wieder mit einer Position und mit der Zeitspanne, wie lange das Objekt an dem Ort angezeigt werden soll. Um eine Grafik einfügen zu können, wird die Funktion zum Zeichnen von Punkten genutzt und im Style des Empfängers der Pfad zur Bilddatei angegeben. Die Nutzung von 3D-Modellen ist etwas komplizierter. Hierzu wird wie im WKML-Abschnitt beschrieben eine zusätzliche Funktion geschrieben. Es wird eine COLLADA-Datei im dae-Format zusammen mit der Position, Zeitspanne, drei Drehwinkeln und Skalierung der drei Achsen an die Funktion übergeben. Die drei Drehwinkel setzen sich zusammen aus Azimut (heading), dieser wurde bereits berechnet, Elevation (tilt) und Rollwinkel (roll), siehe Abb. 4.6. Die Flugbahn des Empfängers wird mit Hilfe eines Arrays erzeugt. In diesem werden alle Positionen gespeichert und einer Funktion zum Erstellen von Polylines übergeben.

4.2.3 Darstellung von Reflexions- und Okkultationsmessungen

Zur Auswahl steht das Erzeugen von Datensätzen aus den Systemen GPS, GLONASS und Galileo. Grundsätzlich wird jedes System einzeln abgearbeitet und enthält eigene Unterordner für Reflexionen,

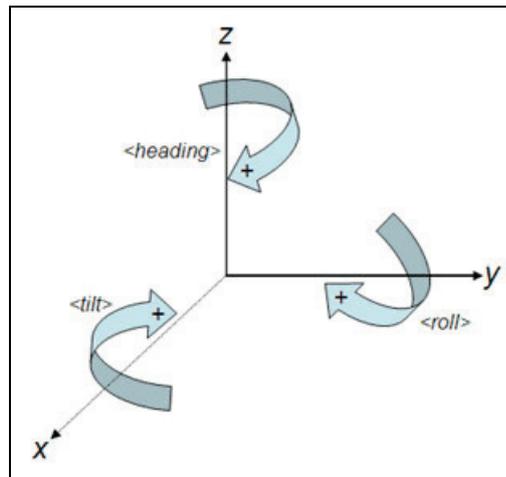


Abb. 4.6: Ausrichtung eines Objektes in Google Earth [PicKMLRef]

Okkultationen, TEC und Grafiken, siehe Abb. 4.4. Die Bearbeitung der Daten ist in allen drei Systemen gleich und wird daher nur einmal erläutert. Da der Schwerpunkt des Programms auf den Reflexionen liegt (Abb. 4.7), werden diese als Erstes verarbeitet. Dazu werden alle Dateien in einem Ordner des jeweiligen Systems gesucht und die Dateinamen zwischen gespeichert.

Die erste Datei wird ausgewertet und mittels Inhaltsinformationen ein Ordner mit der aktuellen PRN erstellt. Um den Strahlenweg von Reflexionen sichtbar zu machen, werden zwei Linien benötigt. Die erste fängt an der Position des GNSS-Satelliten an und endet am Reflexionspunkt auf der Erdoberfläche (z.B. Eis oder Wasser). Die zweite Linie knüpft an der ersten an und geht von der Erdoberfläche bis zur Position des Empfängers. Die Zeitspanne der Sichtbarkeit wird, wie bereits oben erwähnt, ermittelt. Als Farbe für Reflexionen wird für jedes GNSS-System eine unterschiedliche verwendet. Das GPS erhält eine rote Darstellung, Galileo orange und GLONASS gelb. Wenn der Nutzer eine Quality-Datei übermittelt, werden statt dessen nach Verwendbarkeit der Daten die Linien grün (verwendbar) oder rot (nicht verwendbar) dargestellt. Ein weitere Möglichkeit ist die Nutzung von unterschiedlichen Farben für jede PRN. Diese werden bereits am Anfang definiert und sollen der Unterscheidung der PRN dienen.

Ein Reflexionsverlauf soll dargestellt werden, um den groben Überblick zu erhalten, auf welcher Fläche eine Reflexion stattfindet. Der Verlauf wird auf Wunsch des Nutzers auf der Erdoberfläche visualisiert. Dazu wird jedes Mal geprüft, ob vor der aktuellen Reflexion schon eine auftrat und falls dies der Fall ist, werden die Punkte mit einer breiten Linie verbunden. Die Farbe richtet sich dabei nach der des dargestellten Strahlenweges. Die Anzeigedauer des Verlaufs kann variiert werden. Entweder wird er über den ganzen Zeitraum angezeigt oder nur über einen gewissen Zeitraum. Diese Einstellung kann der Nutzer treffen.

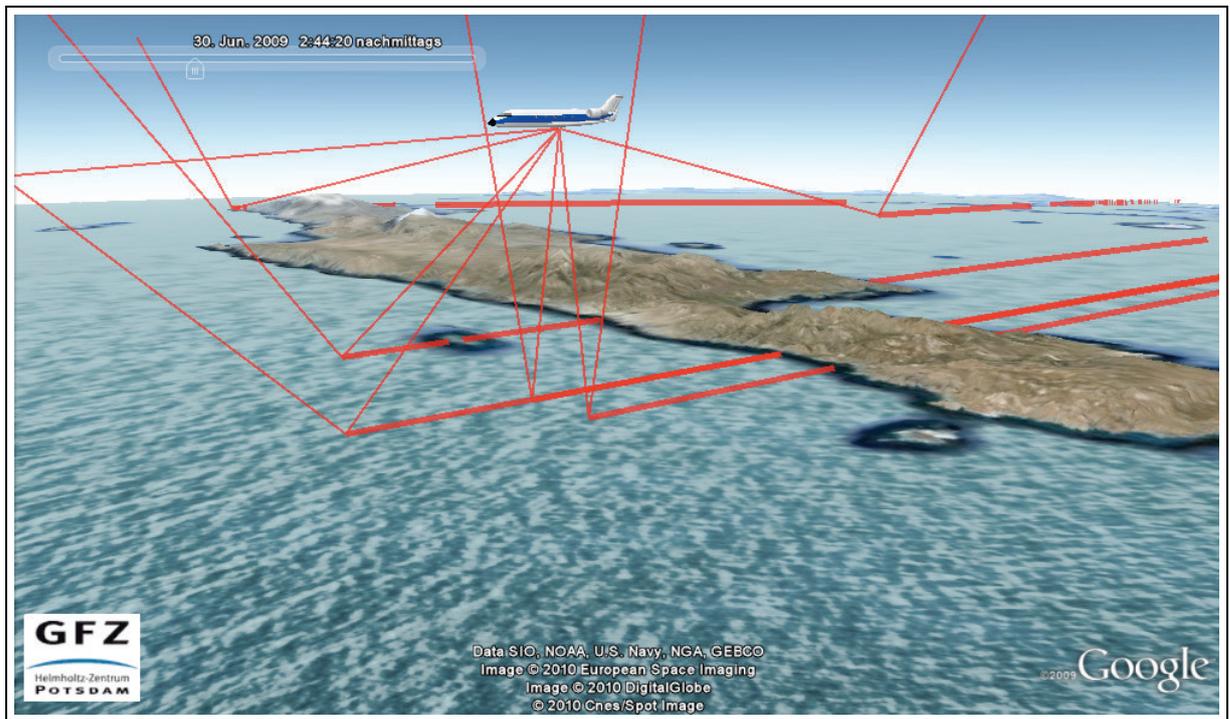


Abb. 4.7: Reflektierte Signale (dünn), Reflexionsverlauf auf Erdoberfläche (dick)

Die Verarbeitung der Okkultationen erfolgt ähnlich denen der Reflexionen. Es werden die notwendigen Dateien gesucht, eingelesen und für jede PRN ein Ordner erstellt. Der Signalweg von Okkultationsmessungen wird grün dargestellt. Da es sich hierbei eigentlich um eine gebeugte Linie handelt, die durch die Atmosphäre läuft (vom Sende- zum Empfängersatelliten), werden zwei einzelne Linien gezeichnet.

Der Punkt, der als Ab- oder Aufstiegspunkt am Ende der vertikalen Linie dargestellt wird, ist der so genannte Tangentenpunkt. Das ist der Punkt, der auf der Sichtlinie zwischen Sender und Empfänger der Erdoberfläche am nächsten ist. Die Beugung durch die Atmosphäre lässt diesen etwas tiefer liegen, als dies ohne Atmosphäre der Fall wäre. Die erste Linie hat ihren Ursprung am Sendesatelliten und geht bis zu dem Tangentenpunkt. Ab diesem Punkt wird das Signal gebrochen dargestellt und von einem zweiten Satelliten empfangen.

Eine Fläche auf der Erdoberfläche soll veranschaulichen, welcher Bereich der Atmosphäre in dieser Messung beobachtet wird. Die Fläche wird als Polygon gezeichnet und im gleichen Intervall aktualisiert wie die Signalwege. Wenn die Okkultation beendet ist, wird das Polygon bis zum Ende der Animation sichtbar gemacht. Verwendet wird die Position der ersten und letzten Okkultation und die Zeitspanne wird so gesetzt, dass die letzte Beobachtung die Anfangszeit bildet. Die Endzeit wird nicht definiert und ist somit automatisch bis zum Ende sichtbar. Der Okkultationsverlauf in der Atmosphäre veranschaulicht ähnlich dem Polygon den Bereich in dem die Beobachtung stattfindet.

Der Verlauf ist eine meist vertikale Linie, die alle Okkultationspunkte verbindet. Abb. 4.8 gibt einen Überblick der drei erwähnten Darstellungen.

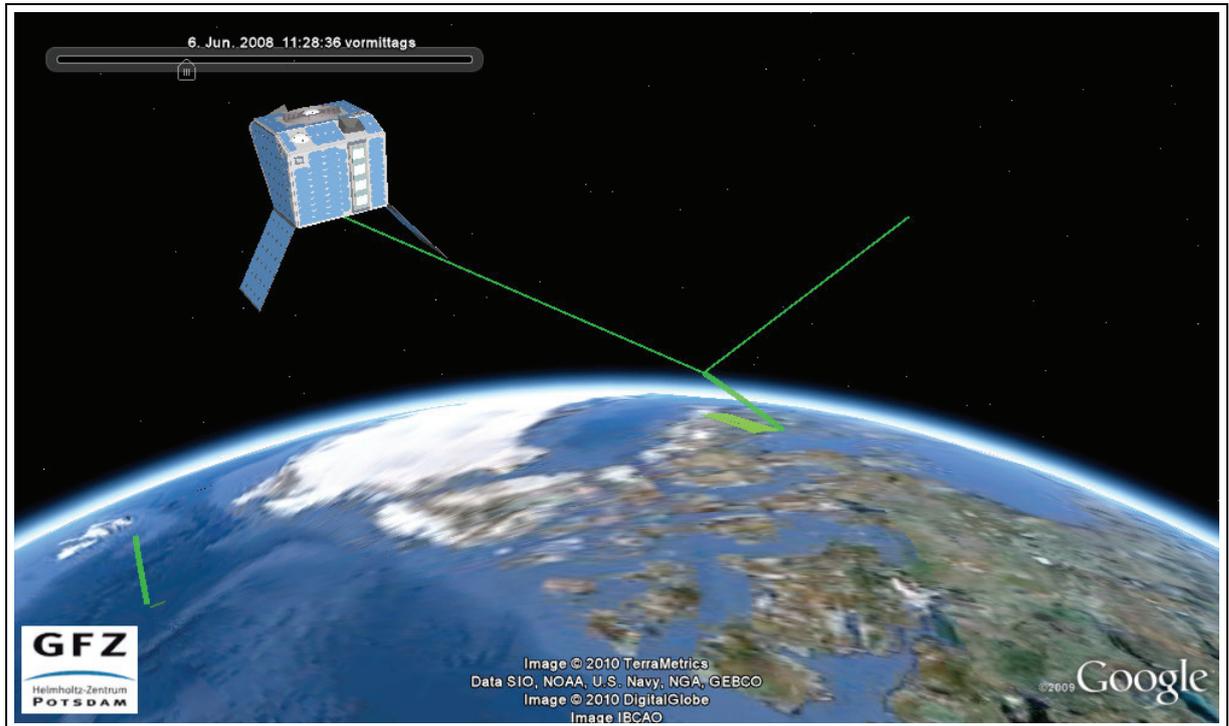


Abb. 4.8: Messung einer Okkultation (gebrochene Linie), Okkultationsverlauf (vertikaler Strich), erfasster Atmosphärenbereich (auf Erdoberfläche projiziert)

Als letzte darstellbare Messung kann der Nutzer TEC aktivieren. Das Einlesen der Daten und Erstellen der Unterordner erfolgt wie bei Reflexion und Okkultation. TEC wird mit Hilfe einer Linie zwischen Sender und Empfänger dargestellt (Abb. 4.9). Sie veranschaulicht den Bereich der in einem gegebenen Zeitintervall eine Gesamtelektronenzahl entlang des Strahlenweges liefern kann.

Für eine bessere Übersicht des Gesamtsystems wird zur Unterscheidung der GNSS-Satelliten jedem Sender eine entsprechende Grafik von einem GPS, Galileo oder GLONASS Satelliten zugewiesen. Dazu werden die Dateien für die angegebenen Systeme gesucht und für jede PRN die Position ausgelesen. Der Nutzer bestimmt im Vorfeld die zeitliche Auflösung der Grafiken. Alle Daten, die nicht im Intervall liegen, z.B. 10 Sekunden, werden nicht verarbeitet. Dies reduziert die Größe der KML-Datei enorm und verringert die Ladezeiten beim Öffnen der Datei in Google Earth. Über die Sekundenangabe lässt sich eine Grafik über einen längeren Zeitraum an der gleichen Position darstellen. Nach Ablauf der Zeit wird die aktuelle Position ausgelesen und die Grafik wieder für die definierte Dauer angezeigt.

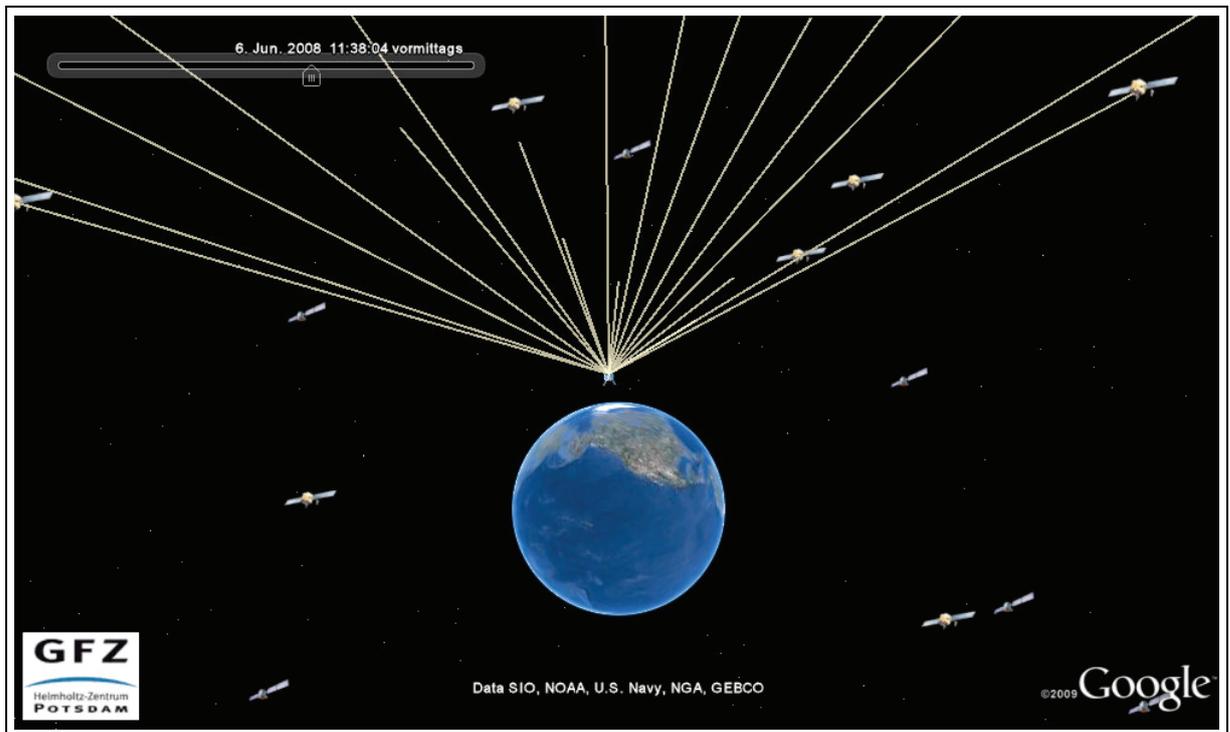


Abb. 4.9: TEC-Messung (beige), im Hintergrund GPS- und Galileo-Satelliten

Während der Auswertung der Reflexionen wird gespeichert, von welcher PRN eine Reflexion ausgeht. Zur Unterscheidung der PRN kann die RandomColor-Funktion aktiviert werden. Dadurch wird jeder Strahlenweg einer Reflexion in einer anderen Farbe dargestellt (Abb. 4.10). Die Farben der PRN werden dann mittels ScreenOverlay-Funktion am Bildschirmrand in einer Legende angezeigt. Die Legende setzt sich dabei aus einzelnen Grafiken zusammen, die übereinander zusammengefügt werden. Die ScreenOverlay-Funktion wird auch dazu genutzt, um das GFZ Logo in jeden Unterordner zu platzieren, damit das Logo nicht verschwindet, wenn eine bestimmte Messung ausgeblendet wird.

4.2.4 Die Tour

Eine Tour ist ein automatischer Flug über den digitalen Globus, der definierte Punkte ansteuert. Die Fluggeschwindigkeit und Verweildauer am Punkt kann definiert werden. Zusätzlich können Texte/Informationen eingeblendet und Audiodateien abgespielt werden. Für die Erstellung einer Tour kommen die in WKML erstellten Funktionen FlyTo und LookAt zum Einsatz. Als Erstes wird die Fluggeschwindigkeit bestimmt. Diese ergibt sich aus der gewünschten Spieldauer der Animation (über Zeitregler in Google Earth einstellbar) und der Anzahl der Positionen, die angefliegen werden

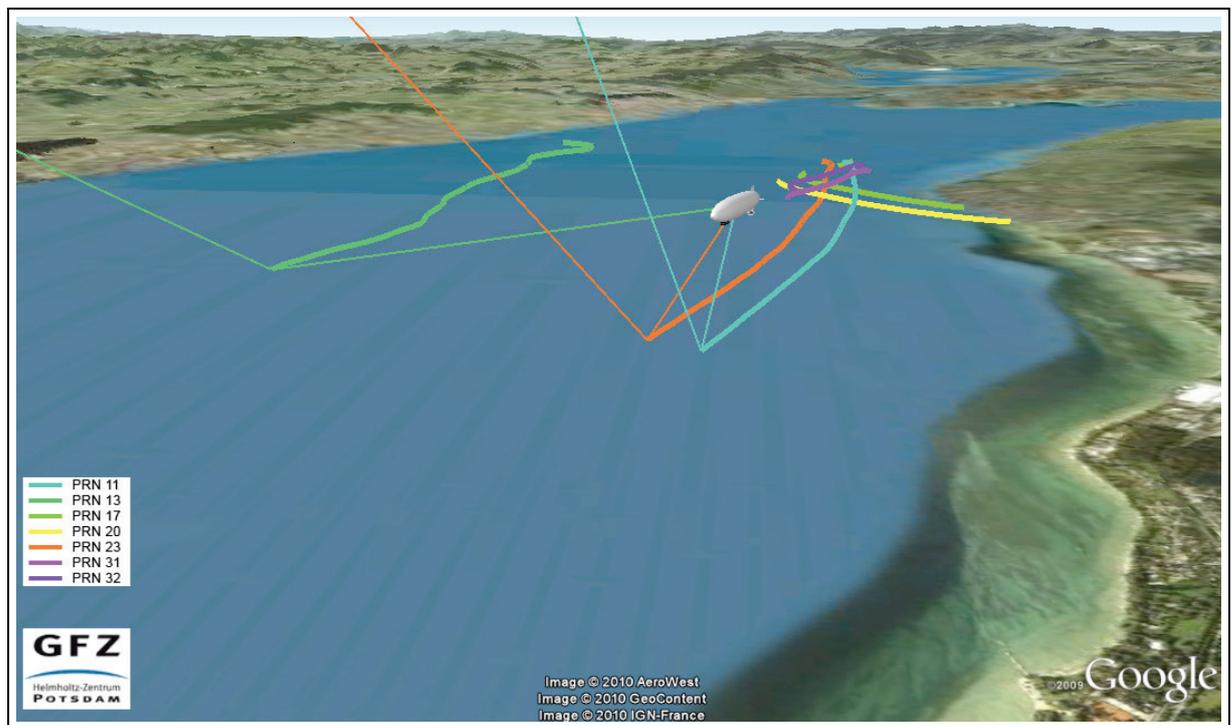


Abb. 4.10: Reflexionen und Verlauf, Unterscheidung der PRN anhand der Farbe

(Anzahl Datensätze). Am Anfang der Tour wird zum Ausgangspunkt gesprungen und die Kamera mittels LookAt auf das Flugobjekt gerichtet, welches während der Animation beobachtet werden soll. Über die ermittelte Fluggeschwindigkeit, die Position der Kamera, dem Zenitwinkel, der Beobachtungsdistanz und dem berechneten Azimut lässt sich das Flugobjekt in einem konstanten Blickwinkel betrachten und verfolgen.

Alle Styles, Visualisierungen und sonstige Informationen werden sequenziell abgearbeitet und in die KML-Datei geschrieben. Zum Ende wird die KML-Datei abgeschlossen und ist für das Öffnen in Google Earth bereit. Die KML-Datei befindet sich im gleichen Ordner in dem auch das Programm ist.

4.3 Parameter-Datei

Um das Programm flexibel wie möglich zu gestalten und variable Einstellungen, wie z.B. Pfade oder Visualisierungswünsche, nicht direkt in den Quellcode zu verankern, muss eine Lösung gefunden werden, dem Programm auf andere Weise diese Informationen zu übermitteln. Die bereits vorhandene Idee wurde erweitert und alle Parameter in einer Datei abgelegt.

Sie bietet eine Übersicht aller Einstellungen und ist schnell und leicht anzupassen. Ein weiterer

Vorteil ist, dass Kommentare hinterlegt und dem Nutzer somit eine Hilfestellung zur korrekten Bedienung gegeben werden kann. Die Parameter-Datei wird beim Starten des Programms übergeben und ausgelesen. Die Auswertung erfolgt nach dem <Key> = <Value> Prinzip. <Key> definiert eine einmalige Variable und Value ein zuweisenden Wert.

Die Datei unterteilt sich in folgende logische Blöcke: zeitliche Auflösung, Pfade und Dateinamen, Auswertung, Tourdaten, Visualisierungen und Sonstiges. Nachfolgend werden alle Parameter aufgelistet und kurz beschrieben.

Tab. 4.1: zeitliche Auflösung, Pfade und Dateinamen

Key	Value
resolution	zeitliche Auflösung der Datensätze (Sekunden)
graphicsres	zeitliche Auflösung der Grafiken der Sendesatelliten (Sekunden)
pathimages	Ordner für Grafiken (Logos, Legende, Satelliten)
nameleo	Name des Beobachtungssatelliten
fileleo	Datei des Beobachtungssatelliten (Messdaten)
docname	Name der Ausgabedatei
date	Datum der Beobachtung
usegraphic	Auswahl Bild oder 3D-Modell
graphicleo	Pfad zum Bild des Beobachtungssatelliten; unterstützt gängige Bildformate
modelleo	Pfad zur dae-Datei des 3D-Modells
filepath1	Pfad zu den Daten der GPS-Satelliten; automatische Zuordnung der Reflexionsfarbe Rot und der Grafik des Senders
filepath2	Pfad zu den Daten der Galileo-Satelliten; automatische Zuordnung der Reflexionsfarbe Orange und der Grafik des Senders
filepath3	Pfad zu den Daten der GLONASS-Satelliten; automatische Zuordnung der Reflexionsfarbe Gelb und der Grafik des Senders

Tab. 4.2: Auswertung

Key	Value
startfolder	Auswahl des Systems welches Ausgewertet werden soll
endfolder	Auswahl des Systems welches Ausgewertet werden soll GPS+Galileo anzeigen (startfolder=1; endfolder=2) Galileo+GLONASS anzeigen (startfolder=2; endfolder=3) nur GPS anzeigen (startfolder=1; endfolder=1) für andere Kombinationen die Filepaths anpassen
checkgpsquality	aktivieren der Qualitätsprüfung für GPS
gpsqualityfile	Pfad zur Qualitätsdatei GPS
checkgalileoquality	aktivieren der Qualitätsprüfung für Galileo
galileoqualityfile	Pfad zur Qualitätsdatei Galileo
checkglonassquality	aktivieren der Qualitätsprüfung für GLONASS
glonassqualityfile	Pfad zur Qualitätsdatei GLONASS

Tab. 4.3: Tourdaten

Key	Value
tour	Tour erstellen oder nicht
duration	Flugdauer (Sekunden)
range	Schrägdistanz zum Objekt (Meter)
tilt	Zenitwinkel 0.0 -> senkrechter 90.0 -> horizontaler - Blick auf das Objekt
heading	Blickwinkel um das Objekt herum, 0 - 180/-180 positiv Blick von West->Ost, negativ Ost->West
historyduration	Anzeigedauer des Reflexionsverlaufs

Tab. 4.4: Visualisierungen

Key	Value
groundstation	Auswahl ob Empfänger eine Bodenstation ist
refl	Anzeigen der Reflexionen
occ	Anzeigen der Okkultationen
tec	Anzeigen der TEC-Messung
vlbi	Anzeigen der VLBI-Messung
slr	Anzeigen der SLR-Messung
downlink	Anzeigen des Datenstromes
UseRandomColor	Nutzen von unterschiedlichen Farben pro Satellit (bei Reflexionen)
trajectory	Anzeigen der Flugbahn des Empfängers

Tab. 4.5: sonstiges

Key	Value
scalex	Vergrößerungsfaktor des 3D-Modells; X-Achse
scaley	Vergrößerungsfaktor des 3D-Modells; Y-Achse
scalez	Vergrößerungsfaktor des 3D-Modells; Z-Achse
offset	Höhenanpassung des Modells, damit Signale in richtiger Antenne enden
screenY	Position der Legende (Pixel), fängt unten an zu zählen

4.4 Webinterface

Die grafische Nutzerschnittstelle, die das ReflOcc's Programm steuert, wird als Webinterface konzipiert (Abb. 4.11). Zur Gestaltung der Seite und Auswertung der einstellbaren Daten kommt HTML, PHP und Javascript zum Einsatz.

The screenshot displays the ReflOcc's web interface, organized into several functional panels:

- Presets:** Includes a 'Load' button and a dropdown menu currently set to 'Zeppelin'.
- Output:** Features a 'Filename' field containing 'Zeppelinnt_s1'.
- Observer:** Contains fields for 'Typ' (with checkboxes for Satellite, Aircraft, and Ground Station), 'Name' (Zeppelin NT), 'File' (presets/zeppelinnt_s1.dat), 'Graphic' (presets/3D_Modelle/Zeppelin.dae), 'Graphic typ' (with checkboxes for 3D Model and Image), 'Offset' (0 meter), 'Scale' (4), 'Resolution' (1 sec), and 'Date' (2008-06-06).
- Data Path:** Lists data sources with 'Upload' and 'Search' buttons: GPS (presets/Zeppelinnt_GPS/), Quality (presets/zeppelinnt_refquality_s1.dat), GALILEO, and GLONASS.
- Visualization:** Offers checkboxes for Reflection, VLBI, Trajectory, Random reflection color, Occultation, SLR, TEC, and Downlink. It also includes 'Show refl / occ' (minutes), 'Signal width' (2 pixel), and 'Signal track width' (5 pixel).
- Tour:** Includes a 'Create tour' checkbox and fields for 'Duration' (69 sec), 'Range' (2000 meter), 'Tilt' (30 degree), and 'Heading' (0.0 degree).

At the bottom of the interface, there are buttons for 'Download KMZ File', 'Start Google Earth in Browser', and 'Re-Download KMZ File (without creating a new one)'.

Abb. 4.11: Übersicht - komplettes Webinterface

Google bietet für Webbrowser eine API an, die Inhalte wie Google Maps oder Google Earth darstellen kann und frei verfügbar ist. Für die Verwendung der Google Earth API muss der Nutzer ein kleines Browser Plug-In installieren. Dieses ist dann in der Lage einige Funktionen von Google Earth bereit zu stellen.

Die Aufgabe des Interfaces ist es, eine Parameter-Datei zu erzeugen und anschließend das fertig kompilierte Fortran-Programm zu starten. Dieses befindet sich ebenfalls auf dem Webserver. Über ein Formular werden alle Einstellungen ermöglicht, die zur Erzeugung der Parameter-Datei wichtig sind. Für den Einstieg in die Bedienung stehen dem Nutzer verschiedene Presets zur Verfügung

und zu jedem Eingabefeld gibt es eine kleine Hilfe. Diese wird sichtbar, wenn die Maus auf das "?" Symbol positioniert wird (Abb. 4.12).



Abb. 4.12: Auszug aus dem Webinterface - Laden von Presets und Hilfe bei MouseOver

Über die Preset-Funktion können verschiedene Beispiele geladen werden. Durch die Auswahl eines Projektes mittels Drop-Down-Menü werden alle notwendigen Eingabefelder automatisch ausgefüllt (Abb. 4.13) und der Nutzer kann sich sofort das Ergebnis ausgeben lassen. Im Hintergrund läuft dabei ein JavaScript, welches bei Änderungen in der Auswahlliste eine Funktion ausführt und überprüft, welches vordefinierte Projekt ausgewählt wurde. Im Anschluss werden Checkboxen aktiviert und die entsprechenden Pfade zu benötigten Dateien in die Eingabefelder geschrieben. Alle Daten wie Grafiken, Sender und Empfänger-Dateien sowie 3D-Modelle befinden sich auf dem Webserver und müssen vom Nutzer nicht selber zur Verfügung gestellt werden. Für die Nutzung eigener Datenbestände kann ein Preset angepasst oder alles manuell eingetragen werden.

Typ ?	<input type="checkbox"/> Satellite <input checked="" type="checkbox"/> Aircraft <input type="checkbox"/> Ground Station
Name ?	<input type="text" value="HALO"/>
File ?	<input type="text" value="presets/HALO_s1.dat"/> <input type="button" value="Upload"/> <input type="button" value="Search"/>
Graphic ?	<input type="text" value="presets/3D_Modelle/HALO.dae"/> <input type="button" value="Upload"/> <input type="button" value="Search"/>
Graphictyp ?	<input checked="" type="checkbox"/> 3D Model <input type="checkbox"/> Image
Offset ?	<input type="text" value="500"/> meter
Scale ?	<input type="text" value="400"/>
Resolution ?	<input type="text" value="1"/> sec
Date ?	<input type="text" value="2009-06-30"/> <input type="button" value="Calendar"/>

Abb. 4.13: Auszug aus dem Webinterface - Einstellungen für den Empfänger

Die Auswahl, an welcher Plattform der Empfänger montiert wird, ist ausschlaggebend dafür, wie die eingelesenen Dateien vom Programm verarbeitet werden und welche Visualisierungsoptionen zur Verfügung stehen. Den unterschiedlichen Empfängern wurde im Vorfeld zugewiesen, welche Messungen überhaupt möglich sind und dem entsprechend werden bei der Auswahl des Empfängers mittels

JavaScript einige Visualisierungen aktiviert bzw. deaktiviert. Um die eigenen Daten benutzen zu können, müssen diese im Netzwerk entweder verfügbar sein oder sie werden dem Webserver temporär übergeben. Für das Hochladen steht eine Upload-Funktion zur Verfügung, die mittels Flash und JavaScript das Hochladen einzelner Dateien oder von kompletten Ordnerinhalten ermöglicht. Im Anschluss bietet das Webinterface eine Suchfunktion an, welche die zuletzt hoch geladenen Dateien sucht und die Eingabefelder für die Pfade füllt.

Dem Nutzer stehen zwei Möglichkeiten zur Verfügung den Datensatz anzuschauen. Er kann sich das Ergebnis direkt im Browser über die Google Earth API ansehen oder für die spätere Betrachtung der Ergebnisse, die Download-Option wählen. Für Linux-Nutzer steht bis jetzt nur die Download-Variante zur Auswahl, da es für Linux noch kein Browser Plug-In gibt, welches die API unterstützt. Unabhängig davon, welche Methode gewählt wird, ist die Erstellung der KML-Datei die gleiche. Durch Auswahl einer der beiden Funktionen werden alle Einstellungen aus dem Formular an ein PHP-Script geschickt und ausgewertet. Bei der Auswertung werden alle Variablen ausgelesen, geprüft und im Anschluss eine Parameter-Datei erzeugt, siehe Kapitel 4.3. Wenn die Parameter-Datei erfolgreich erstellt wurde, wird das Fortran-Programm mit ihr gestartet und der aktuelle Status der Bearbeitung im Browser dargestellt. Dazu wird der Konsolen-Output mit der PHP `fgets`-Funktion abgefangen. Der letzte Schritt ist die Übergabe der erzeugten KML-Datei. Diese wird zusammen mit allen benötigten Grafiken komprimiert und in ein KMZ-Container gespeichert. Die KMZ-Datei ist ein mit dem ZIP-Algorithmus komprimiertes Archiv. Je nach Auswahl wird die KMZ-Datei in einem neuen Browser-Fenster der Google Earth API übergeben oder zum Download angeboten.

5 Installation und Tests

5.1 Installation

Das **Hauptprogramm** wird unter Linux kompiliert und besteht aus einer Datei. Installiert werden kann es unter Linux in einem beliebigen Ordner. Der Nutzer benötigt Schreibrechte in diesem Ordner, damit die KML-Datei erstellt werden kann. Alle Datensätze, die verarbeitet werden sollen, müssen mit Leserechten ausgestattet sein. Es bietet sich an in dem Ordner des Programms für die GNSS, Grafiken und 3D-Modelle Unterordner zu erstellen. Gestartet wird das Programm mit der Übergabe der Parameter-Datei.

```
reflocc preset.para
```

Die erstellte KML-Datei enthält nur die Pfade zu den Grafiken. Beim Öffnen der KML muss darauf geachtet werden, dass die gleichen Ordnerstrukturen vorhanden sind, ansonsten werden keine Grafiken angezeigt. Abhilfe schafft ein KMZ-Container. Dieser beinhaltet die KML-Datei und alle benötigten Grafiken. Die KMZ-Datei kann auf jedem Rechner geöffnet werden, ohne dass Grafiken fehlen. Folgender Linux-Befehl erstellt eine KMZ-Datei.

```
zip NAME.kmz NAME.kml 3D_Modelle/MODELL.dae images/logo.gif images/gps.png
```

Für das **Webinterface** wird ein Apache oder ein anderer Webserver mit PHP-Unterstützung benötigt. Es besteht aus sechs PHP-Dateien und einer Ordnerstruktur, die nicht geändert werden darf (Abb. 5.1). Der komplette Inhalt muss in einem Pfad abgelegt werden auf den der Webserver Zugriff hat und für den er Lese-/Schreibrechte besitzt. Schreibrechte im Wurzelverzeichnis werden für die Erstellung der Parameter und KML-Datei benötigt. Der *tmp*-Ordner und dessen Inhalt benötigen ebenfalls Schreibrechte. Hier werden Datenbestände abgelegt die der Nutzer über das Webinterface hochlädt und im späteren Verlauf die KMZ-Datei gespeichert. Für die restlichen Ordner *css*, *images*, *js*, *presets* und *swf* reichen Leserechte aus. Das Hauptprogramm kann sich auf dem Webserver befinden oder an einem Ort, der über das Netzwerk erreichbar ist. Einige Webserver sind so konfiguriert,

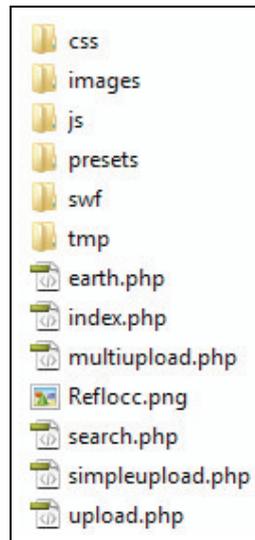


Abb. 5.1: Ordner und Dateien des Webinterfaces

dass sie eine sichere Umgebung besitzen, in der Programme ausgeführt werden können, z.B. der *cgi-bin*-Ordner. Der Pfad zu dem Programm muss in der *index.php* angepasst werden.

5.2 Tests

Getestet wird das Programm in einer Linux-Umgebung (shell) und über das Webinterface. Unter Linux wird die Parameter-Datei manuell angepasst und das Programm gestartet. Es erfolgt eine Konsolenausgabe des aktuellen Bearbeitungsstandes.

Auszug:

```
MicroGEM: bearbeite VLBI ...  
MicroGEM: bearbeite Grafik ...  
GPS: bearbeite Reflexion ...  
Galileo: bearbeite Reflexion ...
```

Bei erfolgreichem Test wird die KML-Datei erstellt und befindet sich im gleichen Verzeichnis wie das Programm. Der Test verläuft erfolgreich, wenn nach dem Start der KML-Datei die Informationen richtig dargestellt werden und alle Grafiken vorhanden sind.

Checkliste:

Parameter-Datei	OK
Programmstart	OK
Bearbeitungsstatus	OK
KML erstellt	OK
KML funktionstüchtig	OK

Für den Test des Webinterfaces wird dieses über eine URL im Browser aufgerufen. Der erste Test wird mit einem Preset durchgeführt, dazu wird das HALO-Projekt geladen. Alle Einstellungen werden automatisch vorgenommen und zusätzlich GPS, die Tour und die Visualisierung von Reflexionen aktiviert. Mit diesen Einstellungen wird die Download-Option und im Anschluss die Option zum Anzeigen der Ergebnisse im Browser ausgewählt (Abb. 5.2).

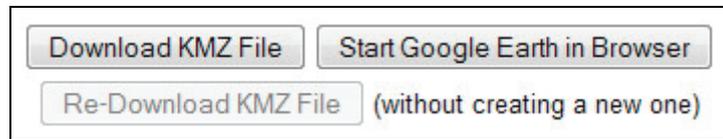


Abb. 5.2: Auswahl der Ausgabe: Download oder direktes Anzeigen der Ergebnisse

Nach Anklicken des Buttons erfolgt wieder die Ausgabe des aktuellen Bearbeitungsstandes. Im Anschluss wird die KML zu einer KMZ-Datei komprimiert und diese Datei zum Download angeboten. Die KMZ-Datei befindet sich nun auf dem PC und wird mit Google Earth geöffnet. Der Test verläuft erfolgreich, wenn alle Informationen korrekt dargestellt werden. Der Button zum Anzeigen der Ergebnisse im Browser durchläuft die gleichen Bearbeitungsschritte, aber anstatt die Datei zum Download anzubieten, öffnet sich ein neues Fenster, in dem das Google Earth Plug-In gestartet wird. Von diesem wird die KMZ-Datei geladen und alle Informationen und Grafiken ordnungsgemäß dargestellt.

Eigene Einstellungen werden getestet, indem Messdaten und 3D-Modelle über das Upload-Formular hochgeladen werden. Als Test dient eine Satellitenmission mit Daten aus den Systemen GPS, GLO-NASS und Galileo. Es werden alle Visualisierungen aktiviert, die zur Verfügung stehen. Getestet wird wieder der Download und die Ausgabe im Browser. Die Schritte sind identisch zum vorherigen Test, siehe oben. Auch hier weist das System keine Fehler auf, wenn alles korrekt dargestellt wird.

Checkliste Preset:

Parameter-Datei	OK
Programmstart	OK
Bearbeitungsstatus	OK
KML erstellt	OK
KMZ erstellt	OK
KMZ funktionstüchtig	OK
Anzeige im Browser	OK
Download-Funktion	OK

Checkliste eigene Einstellungen:

Parameter-Datei	OK
Programmstart	OK
Bearbeitungsstatus	OK
KML erstellt	OK
KMZ erstellt	OK
KMZ funktionstüchtig	OK
Anzeige im Browser	OK
Download-Funktion	OK

6 Schlussbetrachtung

Die Aufgabenstellung dieser Arbeit ist die Entwicklung eines Tools zur Visualisierung und Animation von Messdaten der GNSS-Fernerkundung in der Sektion 1.1 am deutschen GeoForschungsZentrum in Potsdam mittels Google Earth. Das Hauptaugenmerk des Systems liegt in der Darstellung von Reflexionen und Radio-Okkultationen von GNSS-Signalen. Es soll möglich sein GPS, Galileo und GLONASS-Signale in die Visualisierungen einzubeziehen und Signalwege mit unterschiedlichen Farben (rot, gelb und orange) darzustellen. Darüber hinaus können Strahlenwege der einzelnen GNSS-Satelliten über die PRN identifiziert und mit zufälligen Farben belegt werden. Die verschiedenen Farben der PRN werden in einer Legende aufgelistet. Die Unterscheidung zwischen auswertbaren und ungeeigneten Reflexionen erfolgt mit Hilfe einer Qualitätsdatei. Diese Datei markiert Signale, die für die Auswertung ungünstig sind weil, der Einfallswinkel zu gering ist. Geeignete Daten werden grün und die ungeeigneten rot dargestellt.

Für Satellitenmessungen können Radio-Okkultationen aus den Signalen der drei GNSS als Linie dargestellt werden. Zusätzlich wird für eine bessere Anschaulichkeit der Bereich der durchlaufenen Atmosphäre mit einer vertikalen grünen Linie und einem Polygon auf der Erdoberfläche dargestellt. Weitere Messungen wie TEC, VLBI und SLR werden ebenfalls mit Linien visualisiert. Hinzu kommt die Darstellung der Datenübertragung zu einer Bodenstation. Ein aktiver Downlink wird mit einer Verbindungslinie zwischen Sendesatelliten und der Bodenstation dargestellt, die abwechselnd die Farben hellgrau und dunkelgrau annimmt. Bei allen Messungen, die mit Linien visualisiert werden, kann die Strichstärke angepasst werden. Wenn sich der Empfänger auf einem Satelliten oder Flugzeug bzw. Luftschiff befindet, kann zusätzlich eine Flugbahn dargestellt werden.

Alle GNSS-Satelliten werden als Grafiken dargestellt und die Empfängerplattform (Mast, Zeppelin, Flugzeug, Satellit) kann wahlweise mit einem 3D-Modell oder mit einem Bild dargestellt werden. Die Datensätze werden alle mit einem Datum und einer Aufnahmezeit versehen. Somit lassen sie sich in Google Earth animieren und ermöglichen damit nicht nur eine statische sondern auch eine dynamische Betrachtung. Da sich in den meisten Messkampagnen die Empfänger bewegen, wird eine Tour erzeugt, in der die Bewegung des Empfängers nachvollzogen werden kann.

Die Kompatibilität zu anderen Messkampagnen ist gegeben, so dass für weitere Projekte nicht extra ein neues Programm entwickelt werden muss. Dem Nutzer stehen auch verschiedene Voreinstellungen zur Auswahl, um sich einen Überblick über die Funktionalität des Systems zu verschaffen. Durch eine Auswahlmöglichkeit an welcher Plattform der GNSS-R-Empfänger betrieben wird, kann das Programm an die unterschiedlichen Messkampagnen angepasst werden. Eine separate Konfigurationsdatei beinhaltet alle Pfade und Einstellungen, die das Programm benötigt. Diese kann über die Linux-Konsole oder eine grafische Nutzerschnittstelle erstellt werden. Über das Interface lässt sich das ReflOcc's Programm starten und das Ergebnis direkt im Browser anzeigen oder runter laden. Das Programm und die Nutzerschnittstelle sind strukturiert aufgebaut und einzelne Abschnitte sind mit Kommentaren versehen. Somit kann zu einem späteren Zeitpunkt das System leicht erweitert werden. Die Berechtigung zum Ändern des Quellcodes wird durch die Linux-Zugriffsrechte geregelt. Dem GFZ steht mit diesem System ein vollwertiges Tool für die Projektplanung und Veranschaulichung des Aufgabenbereiches der Sektion 1.1 zur Verfügung.

Abkürzungsverzeichnis

API	Application Programming Interface, Programmierschnittstelle
ASCII	American Standard Code for Information Interchange
CAD	Computer Aided Design, rechnerunterstützte Konstruktion
CHAMP	CHAllenging Minisatellite Payload (deutscher Kleinsatellit)
COLLADA	COLLABorative Design Activity (XML-basiertes Austauschformat)
COSMIC	Constellation Observing System for Meteorology, Ionosphere and Climate
dae	digital asset exchange (Dateiformat des COLLADA)
ESA	European Space Agency, Europäische Weltraumorganisation
Galileo	europäisches Sat.-Navigationssystem
GFZ	Deutsches GeoForschungsZentrum
GIS	Geoinformationssystem
GITEWS	German Indonesian Tsunami Early Warning System
GLONASS	Globalnaja Nawigazionnaja Sputnikowaja Sistema, russisches Sat.-Navigationssystem
GNSS	Global Navigation Satellite System, Globales Sat.-Navigationssystem
GPS	Global Positioning System, amerikanisches Sat.-Navigationssystem
GRACE	Gravity Recovery And Climate Experiment
GUI	Graphical User Interface, grafische Benutzeroberfläche
HALO	High Altitude and Long Range Research Aircraft
HTML	Hypertext Markup Language
ID	Identität
ITRF	International Terrestrial Reference Frame
KML	Keyhole Markup Language
KMZ	komprimierte KML-Datei im ZIP-Format
MetOp	Meteorological Operational satellite
MicroGEM	Microsatellites for GNSS Earth Monitoring
NaN	Not a Number
PHP	PHP: Hypertext Preprocessor
PRN	Pseudo Random Noise
RC	Release Candidate
SLR	Satellite Laser Ranging
STEP	Standard for the Exchange of Product model data (Dateiformat)
SVN	Space Vehicle Number
TEC	Total Electron Content
URL	Uniform Resource Locator, einheitlicher Quellenanzeiger
VLBI	Very Long Baseline Interferometry
WGS84	World Geodetic System 1984
WKML	Writing KML, generiert KML-Dateien direkt in Fortran
XML	Extensible Markup Language

Quellenverzeichnis

- [3DTool2010] 3D-Tool Informationen: <http://www.3d-tool-usa.com/overview.html> - 20.05.2010
- [EarthAPI2010] Google Earth API: <http://code.google.com/intl/de-DE/apis/earth/> - 20.05.2010
- [GEMan2010] Google Earth Handbuch: http://earth.google.de/userguide/v4/ug_toc.html - 20.05.2010
- [GFZAtmo2010] GFZ GNSS-Atmosphärensondierung: http://www.gfz-potsdam.de/portal/gfz/Struktur/Departments/Department+1/sec11/M1_Themen/GNSS-Atmosphärensondierung - 23.05.2010
- [GFZRef12010] GFZ GNSS-Reflektometrie: http://www.gfz-potsdam.de/portal/gfz/Struktur/Departments/Department+1/sec11/M1_Themen/GNSS-Reflektometrie - 23.05.2010
- [GoogleEarth2010] Google Earth: <http://earth.google.de> - 20.05.2010
- [GPSConst2009] GPS Konstellation: <http://www.gpsworld.com/gnss-system/gps-modernization/expert-advice-gps-constellation-maxed-out-30-9094> - 20.05.2010
- [Healy2007] Healy, S.B., J. Wickert, G. Michalak, T. Schmidt, & G. Beyerle (2007): Combined Forecast Impact of GRACE-A and CHAMP GPS Radio Occultation Bending Angle Profiles, *Atmospheric Science Letters*, 8: 43-50, DOI: 10.1002/asl.149
- [Helm2008] Helm, A. (2008): Ground-Based GPS Altimetry with the L1 OpenGPS Receiver Using Carrier Phase-Delay Observations of Reflected GPS Signals, Scientific Technical Report Vol. 10, Deutsches GeoForschungsZentrum, Doktorarbeit, TU Berlin
- [Komjathy1996] Komjathy, A., & R. Langley (1996): Improvement of a Global Ionospheric Model to Provide Ionospheric Range Error Corrections for Single-frequency GPS Users, presented at the ION 52nd Annual Meeting, Cambridge MA 19-21 June

- [MicroGEM2009] Brieß, K. Korneman, G. & Wickert, J. (2009): MicroGEM - Microsatellites for GNSS Earth Monitoring, Abschlussbericht Phase 0/A, Deutsches GeoForschungsZentrum und TU Berlin
- [PicGPS] <http://www.kowoma.de/gps/Umlaufbahnen.htm> - 20.05.2010
- [PicKMLRef] <http://code.google.com/intl/de-DE/apis/kml/documentation/kmlreference.html> - 20.05.2010
- [PicSLR] http://www.baader-planetarium.de/kuppeln/refkuppeln_bkg_bremen.htm - 20.05.2010
- [PicVLBI] <http://www.mpifr-bonn.mpg.de/public/holger/2/astrod.html> - 20.05.2010
- [Seeber1989] Seeber, G. (1989): Satellitengeodäsie: Grundlagen, Methoden u. Anwendungen, de Gruyter
- [Sketchup2010] Google Sketchup: <http://sketchup.google.com/intl/de/index.html> - 20.05.2010
- [Stolle2001] C. Stolle, S. Schlüter, N. Jakowski, Ch. Jacobi, S. Heise, A. Raabe (2001): Tomographie in der Ionosphäre unter Einbindung von GPS-Okkultationen, Institut für Meteorologie Uni Leipzig, DLR Institut für Kommunikation und Navigation Neustrelitz
- [Stosius2010] Stosius, R., Beyerle, G., Helm, A., Hoehner, A. & Wickert, J. (2010): Simulation of Space-Borne Tsunami Detection Using GNSS-Reflectometry Applied to Tsunamis in the Indian Ocean, Natural Hazards and Earth System Sciences, (accepted)
- [UNIBW2010] UniBw Okkultation: <http://www.unibw.de/lrt9/forschung/radiosondierung/okkultation> - 20.05.2010
- [Walker1984] Walker, J. G. (1984): Satellite Constellations, Journal of the British Interplanetary Society, 37, 559-571
- [Wickert2002] Wickert, J. (2002): Das Champ-Radiookkultations-Experiment, Algorithmen, Prozessierungssystem und erste Ergebnisse, Doktorarbeit, Institut für Geophysik, Astrophysik und Meteorologie, Karl-Franzens-Universität Graz
- [WKML2010] WKML Projekt: http://web.me.com/dove_family/xml/kml.html - 20.05.2010