

Richtlinie zum „Software-Sprint“

OSM2World – 3D-Modelle der Welt aus OpenStreetMap-Daten

Schlussbericht

Zuwendungsempfänger:

Tobias Knerr

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS24S36 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

OSM2World ist ein Open-Source-Konverter zur Erstellung von 3D-Modellen aus den Daten des OpenStreetMap-Projekts. Damit können dreidimensionale Darstellungen der Welt vollständig auf Basis von offenen Daten und freier Software erstellt werden. Solche Darstellungen werden in alltäglichen Anwendungen wie Online-Karten und Navigations-Apps zunehmend zum Standard und bilden die Grundlage für zahlreiche kreative Projektideen.

Projektziel war die Erweiterung von OSM2World um Schnittstellen, die es Softwareentwicklern erleichtern, die bereits vorhandene Funktionalität zur Erzeugung von 3D-Modellen weiterzuverwenden und die 3D-Modelle in eigenen Anwendungen und Webseiten zu nutzen.

Zu diesen Verbesserungen gehören insbesondere die Erzeugung von gekachelten 3D-Modellen im standardisierten 3D-Tiles-Format, die Verbesserung der Kommandozeilenschnittstelle durch die Einführung von Subkommandos und die Anpassung der Programmarchitektur von OSM2World, so dass die Kernfunktionalität unabhängig vom Rest des Codes mit Maven als Programmbibliothek für Java- und Kotlin-Anwendungen auf Desktop-, Server- und Android-Systemen paketiert werden kann. Auch die Verbesserung der Konfigurationsmöglichkeiten, etwa für das Einlesen eigener 3D-Modelle für bestimmte Objekttypen im glTF-Format, war ein Ziel des Projekts.

Beitrag des Projektes zu den Zielen der Förderinitiative „Software-Sprint“

Digitale Kartenanwendungen sind ein fester Bestandteil unseres Alltags. Ob es darum geht, sich mit dem Smartphone in einer unbekannten Umgebung zu orientieren oder Bürgern ein geplantes

Bauvorhaben zu veranschaulichen – zum Funktionsumfang gehört zunehmend auch die dritte Dimension.

Obwohl diese Anwendungen so eng mit unserer Gesellschaft verwoben sind, basieren sie oft auf den Diensten einiger weniger Anbieter, die innovative Anwendungsfälle durch technische und rechtliche Einschränkungen verhindern. Kreative Nutzungen etwa in der Medienproduktion und der Forschung werden erschwert oder unmöglich. Nutzern, die mit den Geschäftsmodellen oder Datenschutzpraktiken der Anbieter nicht einverstanden sind, fehlt es an Alternativen.

Die freie Weltkarte OpenStreetMap (OSM) setzt dem ein partizipatives Modell auf Grundlage offener Lizenzen und ehrenamtlichen Engagements entgegen. Gängige OSM-basierte Karten bieten allerdings oft nur 2D- oder 2,5D-Darstellungen. Damit der freie Ansatz von OpenStreetMap auch im 3D-Bereich konkurrieren kann, ist eine Software-Infrastruktur nötig, die ähnlich wie die erfolgreichen Produkte aus der 2D-Welt in zahlreichen Endprodukten eingesetzt werden und an die Bedürfnisse der jeweiligen Nutzergruppen angepasst werden kann.

Ausführliche Darstellung der Ergebnisse

Im Rahmen des Projekts wurde die Open-Source-Software OSM2World um Schnittstellen erweitert, die ein einfaches Einbinden der bestehenden Funktionalität zur Erstellung von 3D-Modellen aus OpenStreetMap-Daten in unterschiedliche Anwendungen ermöglichen.

Besonders geeignet für Anwendungen im Java- und Kotlin-Ökosystem ist die direkte Einbindung als **Programmbibliothek**. Um dies zu ermöglichen, wurden die Programmarchitektur und das Build-System von OSM2World stark angepasst und die Codebasis modularisiert. Dies vermeidet unnötige Abhängigkeiten bei der Nutzung als Library. Sämtliche von OSM2World unterstützten Exportformate sind mit der neuen Library-Schnittstelle kompatibel. Dazu zählt der Export von 3D-Modellen in Formaten wie glTF, glb und OBJ ebenso wie das Rendern von Bildern mit OpenGL. An etlichen Komponenten von OSM2World wurden mit Blick auf einfache Verwendbarkeit und Verständlichkeit für Nutzer der Bibliothek erhebliche Verbesserungen vorgenommen, etwa bei der Package-Organisation, beim Erzeugen von Input-Datensätzen, beim Setup von komplexen Datenstrukturen wie virtuellen Kameras in 3D-Szenen und der Handhabung von Konfigurationsparametern. OSM2World wurde zudem so umstrukturiert, dass die neuen Bibliotheksschnittstellen auch intern verwendet werden.

Für die einfache Einbindung in Maven- und Gradle-Buildprozesse wurde die Bibliothek als Maven-Artefakt veröffentlicht. Die notwendigen Schritte zum Einbinden wurden auf der OSM2World-Website dokumentiert. Auch Dokumentation im Javadoc-Format für die diversen Packages von OSM2World wurde angelegt und standardisiert, um neuen Nutzern und Beitragenden die Orientierung zu erleichtern.

Eine Alternative zum Einbinden von OSM2World als Bibliothek stellt der Zugriff auf die erzeugten 3D-Modelle dar. Dieser Ansatz ist prinzipiell weniger mächtig, da lediglich die fertigen 3D-Modelle übertragen werden und der Client keine direkte Kontrolle über die am Server bei der Erzeugung der Modelle verwendeten Parameter hat. Auch eine freie Wahl des Output-Formats ist nicht möglich. Er bietet aber den großen Vorteil, dass der Zugriff über eine standardisierte Webschnittstelle erfolgen kann. Hierbei wurde mit **3D Tiles** ein bestehender, offener Standard gewählt und implementiert.

Dank der Verbreitung von 3D Tiles ist Code zum Laden und Anzeigen solcher Modelle in Client-Anwendungen bereits als marktreife Open-Source-Software verfügbar.

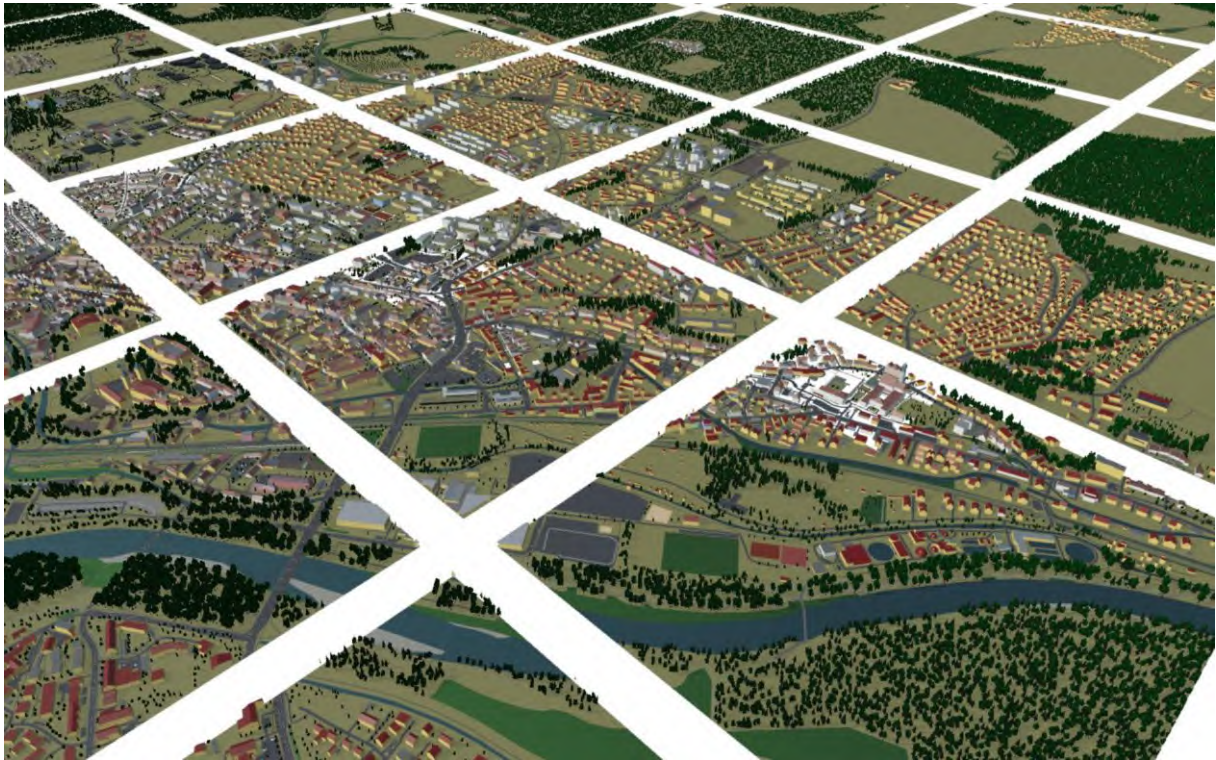


Abbildung 1: Von OSM2World erzeugte 3D Tiles. Kartendaten ©OpenStreetMap (osm.org/copyright).

Um das Ziel einer 3D-Tiles-Unterstützung zu erreichen, mussten viele einzelne Verbesserungen vorgenommen werden: Die Erzeugung von Tileset-JSON-Dateien, Algorithmen zum präzisen Schneiden von Geometrie an Kachelrändern, leistungsfähigere Generierung von Texturen für Physically Based Rendering und verbesserte Server-Performance. Um die Echtzeitdarstellung der 3D Tiles auf schwächerer Hardware wie Mobilgeräten zu ermöglichen, wurden Mechanismen für Level of Detail (LOD) ausgebaut, u.a. durch die automatische Approximation extrudierter runder Formen abhängig vom gewünschten maximalen Fehler und spezialisierte LOD-Ansätze für Gebäude und zahlreiche andere Objekttypen.

Die notwendige Funktionalität für die Erzeugung von 3D Tiles wurde schließlich in einem neu definierten Subkommando gebündelt. Prototypisch konnte auch ein Webserver für das On-Demand-Rendering von 3D Tiles implementiert werden. Die Kompatibilität der Ergebnisse mit Babylon.js, Cesium.js und MapLibre wurde erfolgreich getestet.

Als dritte Schnittstelle neben der Einbindung als Bibliothek und dem Abrufen von 3D Tiles bietet OSM2World eine **Kommandozeilen-Schnittstelle**. Diese wurde durch die Einführung von Subkommandos umstrukturiert. Zudem wurden die bestehenden Parameter systematisch überarbeitet und nach Möglichkeit optional gemacht bzw. mit sinnvollen Defaults versehen, so dass für einen ersten erfolgreichen Aufruf möglichst wenige explizit angegebene Parameter notwendig sind.

Unabhängig von der gewählten Schnittstelle steht nun zudem die Möglichkeit zum **Einbinden von 3D-Modellen** im glTF-Format zur Verfügung, die in der von OSM2World erzeugten 3D-Welt positioniert werden. Neben der Funktionalität zum Einlesen von glTF-Daten wurde hierfür die

Handhabung von relativ zueinander positionierten 3D-Modellen verbessert und die Handhabung von aus Dateien geladenen und zur Laufzeit erzeugten Modellen vereinheitlicht.

Die Umsetzung der neuen Funktionalität wurde durch entsprechende automatisierte Tests begleitet. Bestehende und neue Konfigurationsoptionen wurden an einer zentralen Stelle gebündelt, so dass die Dokumentation der verfügbaren Optionen und Defaultwerte auf einfache Weise aktuell gehalten werden kann.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Als Resultat dieses Projekts haben Softwareentwickler die Möglichkeit, 3D-Darstellungen aus OpenStreetMap zu erzeugen, indem sie OSM2World als Maven-Bibliothek in ihre Anwendungen integrieren oder von OSM2World erzeugte 3D Tiles dort anzeigen. Dies ist in Webanwendungen ebenso möglich wie in Desktop- oder Mobile-Apps und unterliegt dank der offenen Lizenzen der Software und der verwendeten Daten keinerlei Beschränkungen hinsichtlich möglicher Anwendungsfälle.

Aufbauend auf den erzielten Ergebnissen bieten sich zahlreiche Weiterentwicklungsmöglichkeiten an. Dies betrifft insbesondere die Unterstützung weiterer Objekttypen, die Verbesserung der Performance, zusätzliche Ausgabeformate und die Verbesserung bzw. Verbreiterung der Datenbasis:

OSM2World unterstützt bereits eine umfangreiche Liste an Objekttypen. Nichtsdestotrotz finden sich in der OpenStreetMap-Datenbank zahlreiche weitere Typen und Attribute, die bis jetzt noch nicht ausgewertet werden. In den meisten Fällen kann eine Unterstützung mit den bereits implementierten Techniken nachgerüstet werden. Bei einigen Themenbereichen sind jedoch noch umfangreiche Arbeiten an entsprechenden Verfahren notwendig, etwa für eine detaillierte und realitätsnahe Darstellung des Straßenraums in komplexen städtischen Umgebungen.

Eine Verbesserung der Performance von OSM2World ist sowohl bei der Generierung der 3D-Modelle als auch bei der Anzeige wünschenswert. Eine schnellere Generierung von 3D-Kacheln würde es ermöglichen, Kacheln auf einem Webserver erst als Antwort auf die Anfrage eines Clients zu berechnen. Damit würde die Notwendigkeit zum Speichern großer Mengen vorberechneter Kacheln entfallen und Änderungen in den Eingangsdaten würden schneller sichtbar. Eine performantere Anzeige, etwa durch zusätzliche Level-of-Detail-Mechanismen, käme insbesondere mobilen Endgeräten zu Gute. Hier ist die Darstellung aktuell zwar möglich, aber nur unter Abstrichen bei der visuellen Qualität und Reaktionsschnelligkeit der Darstellung.

Klar ist auch, dass das Vorhandensein entsprechender Daten in der OpenStreetMap-Datenbank eine Grundvoraussetzung ist, um eine detaillierte 3D-Welt erzeugen zu können. Hier bietet sich an, in Zukunft durch die Entwicklung und Bereitstellung entsprechender Werkzeuge die Datenerfassung zu erleichtern.

Bei sehr komplexen 3D-Objekten stößt der hier entwickelte Ansatz zudem zwangsläufig an die Grenzen des OpenStreetMap-Datenmodells. Für solche Objekte ist es denkbar, OpenStreetMap als Datenquelle durch externe 3D-Modelle zu ergänzen. Mit der im Rahmen dieses Projekts implementierten Möglichkeit zum Import von glTF-Modellen wurde hierfür bereits eine Grundlage geschaffen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Die ursprünglich geplante MapCSS-basierte Kartenstilsprache wurde im Rahmen des Projekts nicht umgesetzt, da bei Nutzerbefragungen wenig Interesse an einer Ablösung der bereits vorhandenen Konfigurationslösung erkennbar war. Daher wurden die notwendigen Anpassungen, wie z.B. die Möglichkeit zum Einbinden von glTF-Modellen und das Deaktivieren einzelner Module, stattdessen in das bestehende System auf Basis von properties-Dateien integriert.

Der Versuch, neben der Software auch die benötigten Assets des Standard-Kartenstils (Texturen, 3D-Modelle) über Maven auszuliefern, wurde wegen der beträchtlichen Größe der Assets verworfen.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Sämtlicher Code ist in das Git-Repository von OSM2World unter <https://github.com/tordanik/OSM2World/> eingeflossen und steht als Open-Source-Software zur Verfügung. Die Website <https://osm2world.org> dokumentiert die Fähigkeiten und die Nutzungsmöglichkeiten von OSM2World – einschließlich der neuen Features aus diesem Projekt.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Wie zuvor beschrieben zeigten erste Nutzerbefragungen vergleichsweise wenig Interesse an den Plänen, die bisherige Konfigurationslösung durch eine neue Stilbeschreibungssprache zu ersetzen. Ein größerer Schwerpunkt wurde dafür auf häufiger nachgefragte Features gelegt, insbesondere die Unterstützung von 3D Tiles. Mehraufwand entstand hier durch die vergleichsweise komplexe Implementierung der Tileset-Dateien für eine standardkonforme Unterstützung des 3D-Tiles-Standards. In Summe entsprachen die Aufwände für die diversen Teilaufgaben aber der ursprünglichen Planung.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Während des Projekts wurde die kontinuierliche Weiterentwicklung der OpenStreetMap-Tagging-Konventionen durch entsprechende Anpassungen am Quellcode von OSM2World berücksichtigt.