

# QuoVadis Vektor

Ab QV 7.3 / QVM 3 gibt es zwei neue, eigene Karten-Formate:

- QMV - QuoVadis Map Vektor, Vektorkarten
- QMR - QuoVadis Map Routing, Routingdateien für Routenberechnung und Navi
- QMP - Ein Paket (zip-archiv) zum einfachen Download und Installieren, welches QMV und/oder QMRs enthalten kann

Die derzeit wichtigste Anwendung dafür sind OpenStreetMap-Daten.

Mit einen ebenfalls neuen, in QV integrierten Compiler, können jetzt Vektorkarten und Routingdateien aus OSM-PBF Dateien erstellt werden. Inhalt und Aussehen der Karten können auf vielfältige Weise angepasst werden, um Karten für die unterschiedlichsten Anwendungen zu erstellen, z.B. für Radfahrer, Wanderer und natürlich auch für Auto-/ Motorradfahrer.

Die Karten und Routingdateien sind kompatibel mit QV und QVM. Die Anzeige- und Routingperformance ist dabei selbst auf mobilen Geräten mit QVM sehr hoch. Das Routing gehört zum schnellsten, was derzeit auf Smartphones möglich ist.

QMV-Vektorkarten können mit der QV-Standard, -PU und easyROUTES 4 angezeigt werden. Der Compiler zum Erstellen von QMVs und QMR-Routingdateien ist in QV-Standard und -PU verfügbar.

## Import in QuoVadis

QMV-Karten werden wie jede andere Karte auch nach QV importiert. Am schnellsten geht es, wenn man die \*.qmv einfach in die gewünschte Kartentabelle im QV-Xplorer zieht.

Anschließend kann man im Stil der Karte verschiedene Anzeige-Parameter anpassen. Z.B. Schriftgröße, Linienbreite, Zoom-Offset und Rendertheme. Siehe Kapitel [Stil von Karten](#).

QMR-Routingdateien werden einfach in die Tabelle der Routingprofile in die Datenbank **Meine Daten** gezogen und stehen dann als Routingprofil beim Routing zur Verfügung.

Ein QMP-Paket wird mit **Install Package** aus dem **Neu**-Menü installiert.

## Import in QuoVadis Mobile

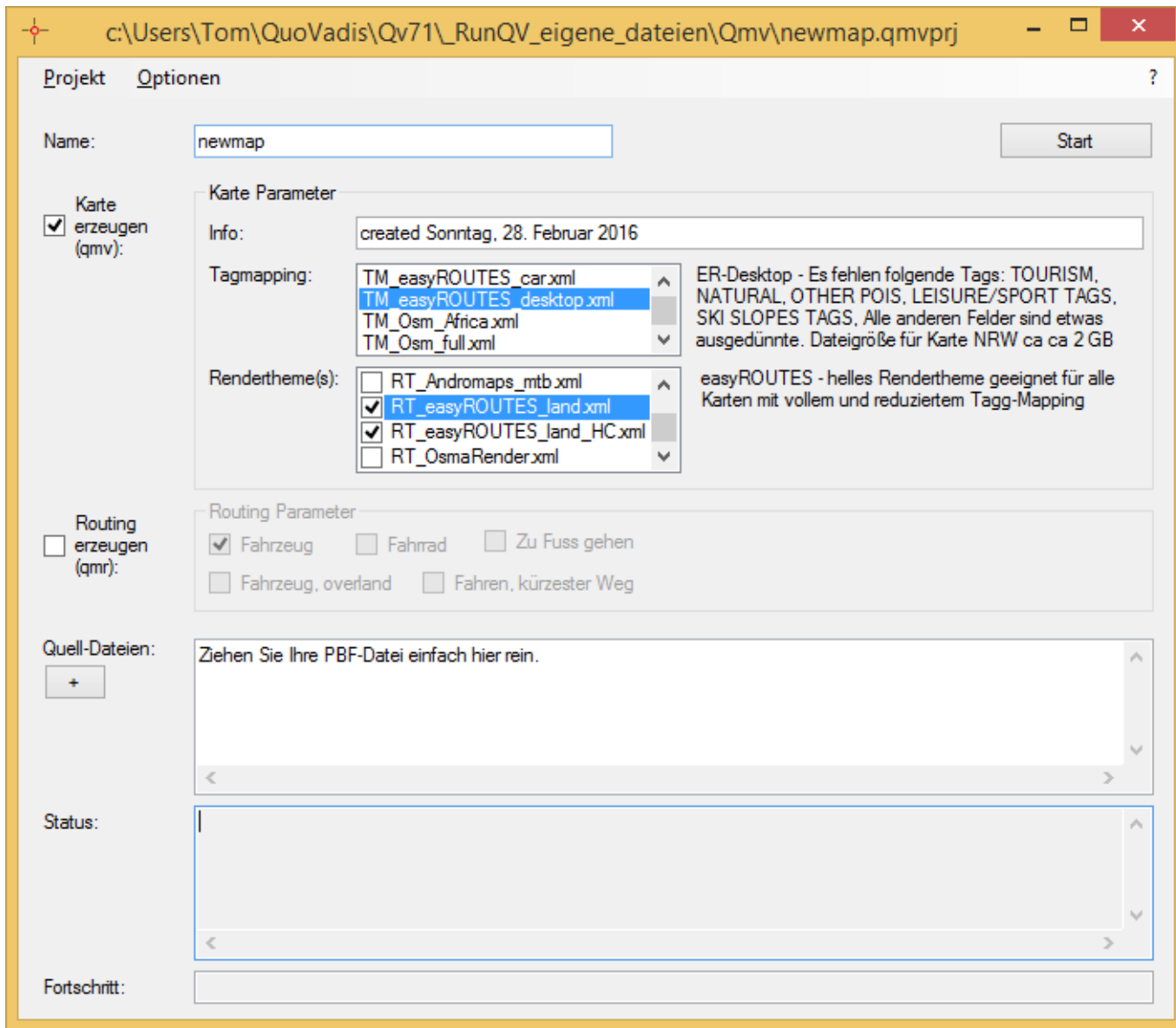
Siehe hierzu die Kapitel im QVM-A3, [QMV Vektorkarten aus QV7 importieren](#)

Und das QVM-I3 Wiki [QMV Vektorkarten aus QV7 importieren](#)

Das Wiki zu QVM-W3 folgt zu gegebener Zeit!

# Der Compiler

Die OpenStreetMap-Daten müssen zuerst in ein für QV und QVM geeignetes Format kompiliert werden. Dazu dient der Compiler. Er ist in QV im Neu-Menü integriert und läuft als 64-bit Anwendung, d.h. er kann nur mit 64-bit-Windows verwendet werden.



Es beginnt mit der Auswahl der gewünschten Region und dem Download der OpenStreetMap-Daten. Hierzu gibt es verschiedene Webseiten:

- [Geofabrik](#): hier können Sie die Daten länderweise downloaden
- [Planet.Osm](#): hier können Sie einen Bereich, auch länderübergreifend, per Polygon bestimmen

In jedem Fall müssen Sie eine \*.PBF-Datei wählen, die Sie dann dem Compiler im Feld **Quell-Dateien** übergeben.

Dann legen Sie einen Namen für Ihre neue Karte fest. Mit diesem Namen erscheint die Karte später im QV-Xplorer und auch die erzeugten Dateien heißen so.

Wenn Sie eine Karte(QMV) erstellen möchten, wählen Sie bitte **Karte erzeugen (qmv)** aus.

Als nächstes bestimmen Sie den Inhalt der Karte. Die PBF-Daten enthalten eine Vielzahl an Daten, viel mehr als oft benötigt wird. Damit die Dateien nicht zu groß werden und auch die Performance gut bleibt, sollte man hier eine sinnvolle Auswahl treffen. Dies geschieht mit dem **Tagmapping**. Verschiedene Tagmappings sind bereits vordefiniert. Wählen Sie einfach ein passendes aus der Liste aus. Hier empfiehlt es sich erst einige Versuche mit einem kleineren Kartenausschnitt durchzuführen!

Dann können Sie das Aussehen bestimmen. Also in welchen Farben und Breiten sollen z.B. bestimmte Straßen erscheinen? Die geschieht über die **Renderthemes**. Auch hier sind bereits einige fertige mit dabei, nur können Sie hier mehrere auswählen zwischen denen dann in der App beliebig umgeschaltet werden kann.

Wenn Sie eine Routingdatei (QMR) erstellen möchten, wählen Sie bitte **Routing erzeugen (qmr)** aus.

Dann müssen Sie bestimmen, welche Art Routing Sie brauchen. Bitte beachten Sie, dass jede zusätzlich getätigte Auswahl die Dateigröße \*.qmr erhöht.

Haben Sie mehrere PBF bei **Quell-Dateien** eingefügt, können Sie keine Routingdatei (QMR), Vektorkarte (QMV) oder auch Paktete (QMP) erzeugen! Sie haben aber die Möglichkeit mit Hilfe von Polygonen bei [Planet.Osm](#) eine Gesamt PBF für die Generierung der QMR, QMV oder QMP zu erstellen.

Das Routing basiert auf OSM-Daten und kann nicht besser sein als die Qualität der OSM-Rohdaten. Speziell für Straßen-Navigation kann daher nicht die Qualität erwartet werden, die ein professioneller Anbieter wie HERE©, Garmin© oder TomTom© bieten.

Als letztes können Sie die gesamte Auswahl als Projekt speichern, um es z.B. später noch einmal laufen zu lassen.

Dann klicken Sie auf Start, um den ganzen Vorgang zu starten.

Das kompilieren kann recht lange dauern und braucht viel RAM. In Status und Progress können Sie den Fortschritt verfolgen.

## Compiler anpassen

Sie können eigene Tagmappings und Renderthemes erstellen und nutzen. Das Standard-Verzeichnis ist QV\_Data\Qmv\Xml. Hier liegen die mitgelieferten Tagmappings und Renderthemes, die Sie als Startpunkt für eigene Anpassungen nehmen können.

Das **Tagmapping** bestimmt, welche Elemente aus dem PBF übernommen werden und ab welchem Zoom-Level sie sichtbar werden in der Karte. Es ist eine XML-Datei, die man nach Wunsch ändern kann. QV benutzt hierfür den [Mapsforge-Standard](#).

Ein **Rendertheme** bestimmt, wie die Elemente gezeichnet werden, also Farbe, Strichstärken, Symbole etc. QV benutzt hierzu eine xml-Datei, ebenfalls im [Mapsforge-Standard](#). Die ausgewählten Renderthemes werden dann zusammen mit den POI-Bitmaps in der QMV gespeichert und - wenn es mehrere sind - kann der Anwender dann unter MapInfo bzw. Style auswählen.

Beim Kompilieren werden die Bitmaps im selben Ordner wie die xml gesucht oder in einem Unterordner. Es werden nur png- und jpg-, aber keine svg-Dateien unterstützt.

## Sprachen

Im OSM sind separate Datenfelder für verschiedene Sprachen der Beschriftungen (Orts- und Straßennamen) vorgesehen.

- „name“ beinhaltet den nativen Namen in der lokalen Sprache und Schrift, dieses Feld ist i.d.R. immer vorhanden
- „name:en“ ist die englische Übersetzung, dieses Feld ist optional, aber relativ oft vorhanden
- „name:de“ ist die deutsche Übersetzung, dieses Feld ist optional, eher seltener vorhanden

Speziell bei Ländern mit anderen Schriftzeichen (Russland, China etc.) ist die Original-Beschriftung („name“) für uns oft nicht lesbar. Im Compiler kann man daher unter **Optionen-Sprachen für Beschriftung** auswählen, welche Sprachen man in der Karte sehen möchte.

Beispiel Karte China: Wenn Sie unter **Optionen-Sprachen für Beschriftung** nur **Default (name)** auswählen, haben sie ausschließlich chinesische Schriftzeichen in der Karte. Wählen Sie zusätzlich noch **EN (name:en)**, haben Sie zusätzlich, wenn vorhanden, die englische Version der Namen angezeigt.

Per default Rendertheme (<caption k=„name“ ...) werden die Beschriftungen in allen vorhandenen Sprachen angezeigt. Wenn Sie das nicht möchten, sondern nur eine bestimmte Sprache, ändern Sie das Rendertheme entsprechend ab. Z.B mit <caption k=„name:en“ ... werden nur die englischen Labels angezeigt.

## Land - Ozean

Leider enthalten die OSM-Daten nur sehr beschränkt Daten zur Darstellung von Land und Ozean, was auf der anderen Seite aber wichtig ist für eine ansprechende Kartendarstellung bei kleinen Maßstäben.

Glücklicherweise haben sich die Jungs von [OpenStreetmapData](#) dem Problem angenommen, und einen hervorragenden Datensatz mit Land-Polygonen erzeugt. Diesen [Datensatz bitte herunterladen](#) (ca. 210MB) und auspacken, dann die darin enthaltene land\_polygon.shp und simplified\_land\_polygons.shp zum Projekt hinzufügen und zwar als letztes.

Im tag-mapping hinzufügen und alle bisherigen Einträge wie „coastline“ oder „ocean“ entfernen:

```
<osm-tag key="shp" value="land-polygons" zoom-appear="9" />
```

```
<osm-tag key="shp" value="simplified-land-polygons" zoom-appear="0" zoom-disappear="9" />
```

Im render-theme werden dann die Farben bestimmt. Und zwar im Header unter map-background die Farbe des Ozeans und dann als erster Weg der Eintrag für „land-polygons“ mit der gewünschten Farbe:

```
<rendertheme ... map-background="#8ccbf7">  
<rule e="way" k="shp" v="land-polygons|simplified-land-polygons">  
  <area fill="#F0EDE5" />  
  <line stroke="#F0EDE5" stroke-width="1" />  
</rule>
```

Die mitgelieferten easyROUTES-Renderthemes und -Tagmappings sowie das TM\_Osm\_Africa sind bereits für die Land\_polygons vorbereitet. Hier genügt es, die SHPs zum Projekt hinzuzufügen mit **Land Polygone hinzufügen** aus dem Options-Menü.

## Pakete

Ein Paket hat die Endung \*.QMP und ist ein Zip-Archiv, welches Karten, Routing-Dateien und Datenbanken in beliebiger Kombination und Menge enthalten kann. Außerdem ist eine xml-Datei enthalten mit Meta-Informationen zum Paket.

Wenn der Schalter „Paket erzeugen“ gesetzt ist, packt der Compiler die erzeugten Dateien automatisch in eine QMP mit gleichem Namen.

## Batch-Lauf

Sie können sehr einfach mehrere Projekte nacheinander laufen lassen. Z.B. um vorhandene OSM-Karten zu aktualisieren nachdem Sie neue pbf's runter geladen haben.

Erstellen Sie einfach ein neues Projekt und laden Sie die gewünschten Projekte (\*. qmvprj) in die Liste der Quelldateien. Sie können das dann auch als neues Projekt speichern.

Mit Start werden die dann alle nacheinander kompiliert. Am Ende wird im Status eine Zusammenfassung ausgegeben mit der Anzahl der Fehler, falls welche aufgetreten sind.

Projekte können nicht mit anderen Quell-Dateien gemischt werden. Wenn Projekte kompiliert werden, werden die Einstellungen zu Karte, Routing, Name, Info etc. in der Maske ignoriert. Die kompilierten Dateien werden in dem Ordner erzeugt, in welchem die entsprechende \*.qmvprj liegt.

From:

<http://wiki07.quovadis-gps.com/> - **QuoVadis GPS Software Wiki**

Permanent link:

[http://wiki07.quovadis-gps.com/doku.php?id=de:35\\_maps:n\\_qmv](http://wiki07.quovadis-gps.com/doku.php?id=de:35_maps:n_qmv)

Last update: **2017/06/18 16:35**

