

# Aufbau eines vektorbasierten Kartendienstes

**Betreuer:** Prof. Jörg Roth

**Zielgruppe:** Bachelor Informatik im Rahmen von 'IT-Anwendungen'

## Aufgabenstellung

### Ausgangslage

Das dorenda-Projekt erlaubt die Generierung von so genannten *Kartenkacheln* aus Geo-Datenbanken. Kartenkacheln sind vorberechnete Bitmapgraphiken für Kartendienste (Abb. 1). Diese fügen sich nahtlos zu einem flächendeckenden Teppich zusammen, der für verschiedene Zoomstufen und Orientierungen berechnet wurde.



Abbildung 1: Beispiele für generierte Kartenausschnitte

Das resultierende Bilder-Verzeichnis kann über einen Web-Map-Service abgefragt werden oder in Anteilen auf Datenträger (z.B. SD-Karten) herausgegeben werden. Programme können das Kartenmaterial dann, z.B. als Hintergrund für Geodienste verwenden (Abb. 2).



Abbildung 2: Benutzung der generierten Karten in Android-Apps

Der große Vorteil des Ansatzes: das Endgerät wird nicht mit der Darstellung der Geodaten belastet, sondern muss nur Bilder darstellen können. Bei der großen Menge an Bitmap-Dateien stößt man allerdings auf ein Mengen- und Größenproblem. So benötigt man schon mit der geografischen Region "Mittelfranken" mehr als 4GB an Bilddaten – für ganz Deutschland benötigt man ca. 1 TB (mehrere zigmillionen Kacheln). Die Größe belastet dabei

- den internen Speicher des Endgerätes (z.B. für das Cachen der Bilder),
- den Übertragungsweg, insbesondere wenn drahtlos
- und die Server-Kapazität.

Darüber hinaus lassen sich Bitmap-Dateien nur schwer drehen und skalieren:

- Beim Skalieren auf Bitmap-Ebene, entstehen Bitmap-Artefakte, Schriften sind nicht mehr gut lesbar und Strichstärken verändern sich in ungeeigneter Weise.
- Drehen ist besonders problematisch. Würde man ein Bitmap für die Ausgabe drehen, könnten Schriftzüge auf dem Kopf stehen. Symbol-Ikonen würden sich mitdrehen, obwohl sie immer aufrecht dargestellt werden sollen.

Aus diesem Grund muss die Kartengenerierung verschiedene feste Orientierungswinkel separat generieren und speichern (Abb. 3). Generiert man aber schon alle Orientierungen im 45°-Winkel, verachtfacht sich die Speichermenge auf dem Server. Eine kontinuierliche Drehung ist aber weiterhin nicht möglich.

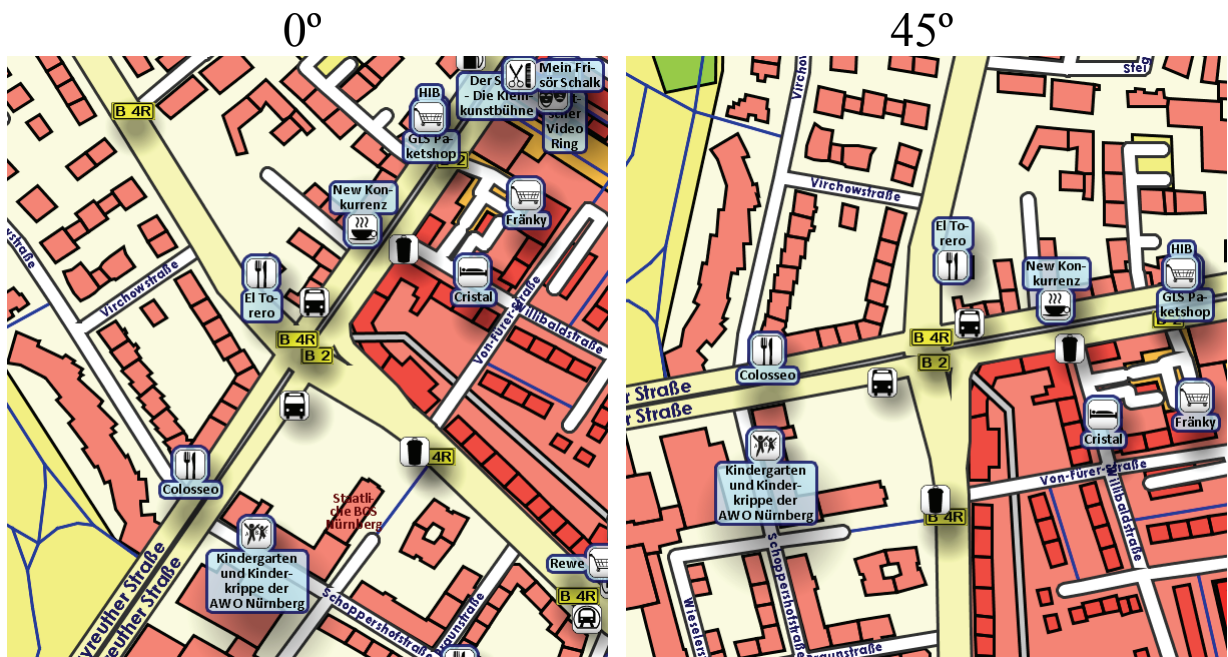


Abbildung 3: Kartenausgaben für verschiedene Orientierungen

Eine Lösung für dieses Problem wäre, Vektor-Grafiken statt Bitmap-Grafiken zu generieren.

### Aufgabe

Es soll untersucht werden, wie man die dorenda-Kartengenerierung auf der Basis von Vektor-Grafiken aufbauen könnte. Folgendes ist zu betrachten:

- Die Vektor-Grafik soll so optimiert werden, dass das Endgerät bei der Darstellung der Karte nicht allzu stark belastet wird. Das Format muss also so optimiert werden, dass die Kartendarstellung über einen einfachen, schnellen und wenig rechenintensiven Mechanismus möglich ist.
- Anteile die nicht gedreht oder skaliert werden sollen, müssen identifizierbar sein. Beispiele: Namen von Punkt-Objekten und Symbol-Ikonen sollen immer aufrecht dargestellt werden, auch wenn die Karte sich dreht. Bei der Beschriftung von Linienzügen darf der "krumme" Schriftzug nicht plötzlich auf dem Kopf stehen.
- Während eine Kartenkachel einen konkreten geografischen Bereich identifiziert, ist die Entität einer "Kachel" im Vektor-Bereich nicht sinnvoll. Es ist also zu klären, welche Portionen auf Serverseite verwaltet werden.
- Beim Zoomen werden bestimmte Objekte ein- oder ausgeblendet. Auch das ist zu berücksichtigen. Vielleicht gibt es weiterhin verschiedene vorgenerierte Zoom-Stufen mit unterschiedlichen Objekten.

Die Aufgabe umfasst:

- die Analyse des Problems, Aufbau eines Konzeptes, insb. Entwickeln eines geeigneten Vektorformates;
- eine Testimplementierung und Optimierung des Konzeptes;
- den Aufbau einer Kartengenerierung auf Serverseite und
- den Aufbau einer Kartendarstellung auf Clientseite.

Das Projekt ist daher insbesondere für Informatiker geeignet. Die das Team sollte Studierende umfassen. Die Realisierung erfolgt in Java (auch der Client). Das gesamte dorenda-Projekt liegt als Source-Code vor. Eine Geodatenbank steht zur Verfügung.