

# Vorschläge zur Erweiterung der OGC Styled Layer Descriptor (SLD) Specification in die dritte Dimension – eine Analyse möglicher Visualisierungsvorschriften für 3D Stadtmodelle

Steffen NEUBAUER und Alexander ZIPF

## 1. Problemstellung

3D Stadtmodelle werden in zunehmenden Maße erstellt. Auch 3D-Visualisierung von Stadt- und Landschaftsmodellen werden über entsprechende OGC Standards bzw. Standardisierungsentwürfe (WTS/WPVS/W3DS) interoperabel ermöglicht. Auch für Basisdaten für 3D Stadtmodelle werden Standards entwickelt – insbesondere CityGML ist hier vielversprechend. Aber in GIS gibt es typischerweise eine klare Trennung von Georohdaten und Visualisierungen. Dies hat große Vorteile, da die gleichen Daten in unterschiedlichen Anwendungsbereichen auf spezifische Weise visualisiert werden können. Dies ist auch für 3D Stadtmodelle notwendig. Bisher war das 3D-Modell quasi schon die Visualisierung, da v.a. Graphikformate wie DXF, DWG, VRML und weitere spezielle CAD-Formate zum Einsatz kamen. Idealer wäre aber eine Situation wie im klassischen 2D-GIS Bereich: D.h. es wäre besser, wenn die Rohdaten nur die Geometrie plus Semantik (Objektklassen) beschreiben an die Sachdaten angehängt werden können, wie z.B. bei GML. Davon unabhängig sollte ein eigenes Format für die Visualisierungsvorschrift definiert sein, die auf die entsprechenden Features angewendet werden sollen. Diese Situation findet man heute bei 2D-Karten im Web – SLD bietet zahlreiche Möglichkeiten (vg. Kap.2). So ist es möglich in einer WMS-Karte Daten aus verschiedenen Datenquellen zu integrieren und einheitlich per SLD zu stylen. Es wäre wünschenswert wenn dies ebenfalls mit 3D-Daten möglich wäre. Der WPVS/W3DS/WTS ist ja das Analogon zu WMS und CityGML ist ja eine Anwendungsschema von GML. Daher wäre eine Erweiterung von SLD in die dritte Dimension die ideale Lösung. Bisher ist hierfür noch nichts vorgesehen oder geplant. Auf der anderen Seite gibt es allerdings Überlegungen CityGML um weitere Visualisierungselemente zu erweitern. Dies würde aber gegen die gewünschten Trennung von Rohdaten und Visualisierungsvorschrift sprechen und zudem neben SLD eine parallele Visualisierungsvorschrift innerhalb des OGC implizieren, da ja im 3D Bereich auch 2D-Elemente abgedeckt werden müssten. Daher erscheint eine Erweiterung von SLD der konzeptionell vielversprechendere Ansatz zu sein. Allerdings ist dies nicht ganz trivial, da 3D-Visualisierung deutlich komplexer ist und umfangreiche Erweiterung nötig werden. In diesem Beitrag möchten wir einige erste Ideen vorschlagen und mögliche Erweiterungen von SLD in Richtung 3D diskutieren. Diese Erweiterungen können dann in einem W3DS implementiert werden. Dies geschieht gerade im Projekt 3D-GDI Heidelberg. Einerseits kann die SLD-Datei schon zur Konfiguration eines W3DS dienen, andererseits kann es dynamisch wie bei einem WMS genutzt werden, um das Aussehen des Ergebnisses vom Client aus zu bestimmen.

## 2. Möglichkeiten und Grenzen von Style Layer Descriptor (SLD)

Mit SLD bietet OGC die Grundlagen, die begrenzten Möglichkeiten eines gewöhnlichen WMS bezüglich der Kartendarstellung wesentlich zu erweitern. Ohne SLD ist es einem

WMS ist es lediglich möglich, einen Datensatz in einer vordefinierten Weise durch eine begrenzte Anzahl von Styles "im Ganzen" darzustellen. Das heißt, mit WMS allein ist z.B. von Clientseite aus keinerlei Klassenbildung der vorhandenen Daten möglich - einem Layer wird ein einziger Style zugeordnet. Will man z.B. die Daten klassifizieren, d.h. attributabhängig in verschiedene Klassen einteilen und diese dann auch unterschiedlich darstellen, müsste man mit einem gewöhnlichen WMS alle möglichen Varianten als eigene Layer vorhalten. Alternativ benötigt man ein weiteres Werkzeug, das es ermöglicht ein derartiges Finetuning auch von Clientseite „auf Request“ hin auszuführen. Unter anderem dies ermöglicht die SLD Spezifikation. So können ähnlich eines Signaturenkatalogs die Visualisierung der Geodaten dynamisch vom Klient, der die Karten anfordert bestimmt werden. Hiermit gewinnt man wesentlich an Flexibilität bei der Darstellung von Karten mit Daten aus heterogenen Quellen, da diese auch "on the Fly" mit gleichartigen Visualisierungsvorschriften versehen und homogen dargestellt werden können. SLD ist nicht nur in der Lage, mit WMS zusammenzuarbeiten, sondern kann auch benutzt werden, um Inhalte von WCS oder WFS darzustellen und entsprechend zu symbolisieren.

### 3 Struktur von SLD

Die SLD-Spezifikation nutzt XML-Schema zur Definition der möglichen Elemente für die Symbolisierung der Karten. Der wichtigste Teil der SLD ist der Abschnitt der Regel (Rule): hier wird festgelegt, in welchem Maßstab, in welcher Füllung oder Strichstärke bzw. Transparenz der Layer auf der Karte erscheint. Das Element, welches im SLD-Dokument weiter "unten" liegt, wird später gezeichnet und befindet sich damit relativ zu dem im Dokument "darüber liegenden" Element im Vordergrund. Somit ist eine Prioritätenverwaltung möglich. Das Element *Rule* ist wichtig, da mit ihm die Definition einer Regel zur Klassifizierung des Layers eingeleitet wird. Alle notwendigen Informationen sowohl zur Klassenbildung einer aktuellen Klasse als auch deren Symbolisierung stehen innerhalb des Elementes *Rule*. Für die Symbolisierung selbst gibt z.Zt. *PointSymbolizer*, *LineSymbolizer*, *PolygonSymbolizer*, *TextSymbolizer*, (*RasterSymbolizer*). Unter den jeweiligen Symboldefinitionen sind alle zur Symbolisierung der jeweiligen Klasse notwendigen Einstellungen und Eigenschaften festgelegt. Attribute wie Farbfüllung, Strich-, Text- und Punktfarbe, Strichstärke, Texthöhe, Punktgröße, Deckkraft der Farbe u.v.m., aber auch die Art von Flächenfüllungen (vollflächig, mit graphischen Füllelementen oder der Referenz auf ein hinterlegtes Bitmap) sowie das Aussehen von Linien (gestrichelt, gepunktet etc.) werden hier definiert (vgl. Müller 2007, Andrae et al 2007).

### 4 Interoperable Visualisierung von 3D Stadtmodellen

3D-Szenen könne mit W3DS über das Internet an einen 3D-Viewer übersandt werden. Bei der Verwendung des W3DS (Web 3D Service) können bislang noch keine Beschreibungen für dreidimensionale Geometrien per SLD verwendet werden, denn die derzeitige SLD bezieht sich nur auf zwei Dimensionen. Im W3DS sind omit bisher nur feste vordefinierte Styles per Namen anzugeben, wie bei WMS ohne SLD. Eine Erweiterung des W3DS-Entwurfs (bzw. WTS/WPVS) zur Unterstützung von 3D-SLD erscheint somit sinnvoll, da dann ähnliche Anwendungen mit 3D-Visualisierungen von Stadtmodellen möglich werden,

wie bisher bei SLD-WMS. Interessante Szenarien sind denkbar: So könnte Stadt oder Region X und Y jeweils einen eigenen W3DS mit dem eignen Stadtmodell betreiben, aber z.B. ein 3D-Navigationsclient bei einer Route von Stadt A nach B beide Städte nach den gleichen Visualisierungsvorschriften darstellen. Oder Stadtplaner können in einer Webanwendung per 3D-SLD Planungsvorhaben unterschiedlich einfärben, etc.

### 3. Erweiterungen von SLD für die 3. Dimension

Im Folgenden wird ein Ansatz für die Erweiterung von SLD in die dritte Dimension vorgestellt. Es wurde untersucht welche Erweiterungen von SLD für die dritte Dimension notwendig sind. Dabei haben wir folgende Liste möglicherweise relevanter Aspekte zusammengestellt:

- Rotation von Elementen für alle drei Achsen
- Verschiebungen und Positionen werden mit Z erweitert
- SolidSymbolizer zur Beschreibung von Volumenkörpern
- SurfaceSymbolizer können Oberflächen mit Dreieckvermaschung definieren
- Extrusion von Geometrien
- Einbinden von externen 3D-Objekten in die Scene
- Materialbeschaffenheit mit Reflexionseigenschaften des Lichts beschreiben
- Billboard's (Definition 1) für 2D Grafik
- Legenden auch in 3D
- Linien als Rohr darstellen (z.B. für Routenplan, Rohre o.ä.)

Die folgende Tabelle fasst die Erweiterungsvorschläge zusammen:

Element	Erweiterung	Funktion
LegendGraphic	Graphic_3D <ul style="list-style-type: none"> <li>• ExternalGraphic_3D               <ul style="list-style-type: none"> <li>o Format_3D</li> <li>o OnlineResource</li> </ul> </li> <li>• Mark_3D               <ul style="list-style-type: none"> <li>o WellKnownName</li> <li>o Fill</li> <li>o Stroke</li> </ul> </li> <li>• Opacity</li> <li>• Size</li> <li>• Rotation3D</li> </ul>	Einbinden von 3D Legenden Als externes 3D Objekt Format der Datei Link zur Datei Als einfaches 3D Objekt Bekanntes Objekt Ausfüllen der Flächen Zeichnen der Kanten Undurchsichtigkeit Größe 3D Rotation
Rule	SolidSymbolizer <ul style="list-style-type: none"> <li>• Geometry</li> <li>• Fill</li> <li>• Stroke</li> </ul>	Beschreiben von 3D Körpern Geometrie zuweisen Ausfüllen der Flächen Umranden der Flächen
	SurfaceSymbolizer <ul style="list-style-type: none"> <li>• SurfaceGeometry               <ul style="list-style-type: none"> <li>o TIN</li> <li>o MassPointRelief</li> <li>o BreaklineRelief</li> <li>o RasterRelief</li> </ul> </li> <li>• Fill</li> <li>• Stroke</li> </ul>	Beschreiben von 2.5D Flächen Art der Geometry Siehe Definition 2 Aus City GML übernommen Bruchkanten DGM Raster DGM Ausfüllen der Oberfläche Verbinden der Flächenpunkte
LineStyle	Extrusion	Linien werden zu Fläche

	<ul style="list-style-type: none"> <li>• Height</li> <li>• Offset</li> <li>• Base_Height</li> <li>• SurfaceGeometry ...</li> </ul>	Höhe der Extrusion Zuschlag zur Grundhöhe Grundhöhe Oberfläche als Grundhöhe
	Pipe <ul style="list-style-type: none"> <li>• Radius</li> <li>• CssParameter</li> </ul>	Linien werden zu Rohren Radius des Rohres Farbe
	Billboardplacement <ul style="list-style-type: none"> <li>• AnchorPoint ...</li> <li>• Displacement ...</li> <li>• Rotation</li> </ul>	Siehe Definition 1 Ankerposition des Billboard Verschiebung vom Ankerpunkt Rotation des Labels auf der Billboard
	Extrusion ...	Polygone werden zu Klötzchen
	BillboardPlacement ...	Siehe Definition 1
PointSymbolizer	Graphic_3D ...	Einbinden von 3D Modellen
	BillboardPlacement ...	Siehe Definition 1
	Extrusion ...	Punkte zu Linien
TextSymbolizer	Depth	Schrifttiefe
RasterSymbolizer	BillboardPlacement ...	Siehe Definition 1
LabelPlacement	BillboardPlacement ...	Siehe Definition 1
Fill	Material <ul style="list-style-type: none"> <li>• DiffuseColor</li> <li>• AmbientIntensity</li> <li>• SpecularColor</li> <li>• Shininess</li> <li>• Transparency</li> </ul>	Farbe & Reflexionseigenschaften der Füllung Farbe der diffusen Reflexion Intensität der diffusen Reflexion Farbe der direkten Reflexion Intensität der direkten Reflexion Transparenz
PointPlacement	Rotation3D <ul style="list-style-type: none"> <li>• Rotation_X</li> <li>• Rotation_Y</li> <li>• Rotation_Z</li> </ul>	Rotieren des Punktes 3D Rotation um X Rotation um Y Rotation um Z
AnchorPoint	AnchorPointZ	Anker in Z-Richtung
Displacement	DisplacementZ	Verschiebung in Z-Richtung

#### 4. Umsetzung der Erweiterungen für SLD

Für all die in der Tabelle vorgeschlagen Erweiterungen wurden entsprechende XML-Schema erstellt. Hierbei wurde bei den Erweiterungen auf eine Abwärtskompatibilität geachtet. D.h. es sind alle alten Elemente erhalten geblieben und nur zum Teil mit optionalen Elementen aufgefüllt wurden. Somit lässt sich die erweiterte SLD für WMS und W3DS verwenden. In der Abb. 1 ist der Aufbau des Elements *Rule* zu sehen, nachdem es auf 3D erweitert wurde. Als zweites Beispiel ist der Aufbau des *PointSymbolizer* mit seinen neuen und alten Unterelementen in Abb. 2 dargestellt. Man sieht, dass die Erweiterungen sehr umfangreich werden und hier aus Platzgründen nicht dargestellt werden können. Nach erfolgreichem Test werden sie online zur Verfügung gestellt. Bis zur AGIT sollte dies geschehen sein, so dass die hier skizzierten Erweiterungen ausführlicher diskutiert werden können. Ausserdem sollte bis dahin wenigstens unsere W3DS-Implementierung (Schilling et al 2007) 3D-SLD zur internen Konfiguration (also als Konfigurationsdatei) weitestgehend nutzen können. Wahrscheinlich wird auch schon die Nutzung per Web-Request mit einem Verweis auf eine entsprechende SLD in der URL möglich sein.

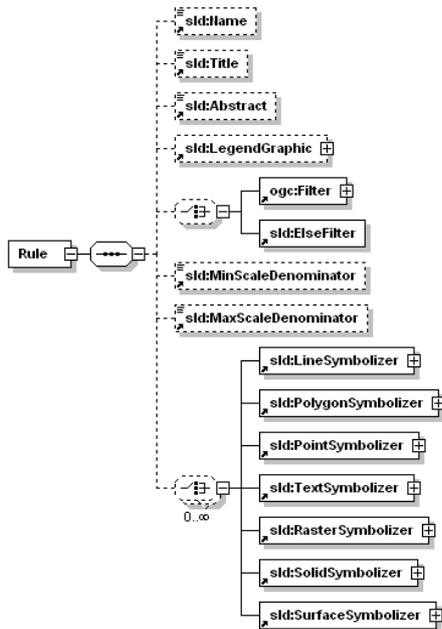


Abb. 1: Schema Diagramm des Elements Rule

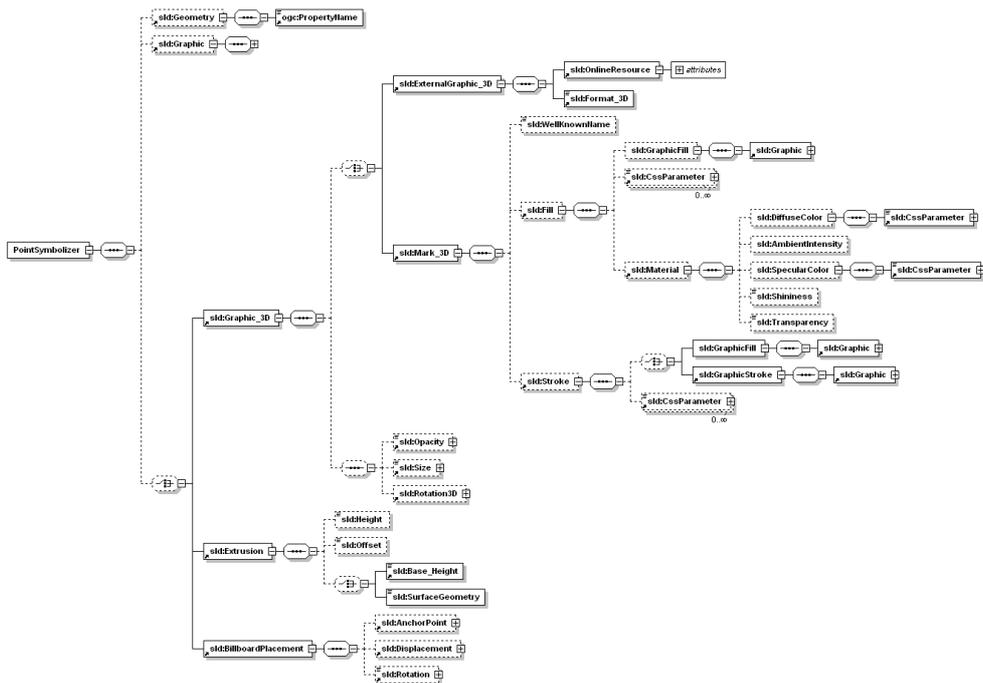


Abb. 2: Aufbau des PointSymbolizer

## 5. Ausblick

Die vorgestellten Ergänzungen und Überlegungen sind nur ein erster Ansatz und müssen als nächster Schritt an einem konkreten 3D Stadtmodell mit einem W3DS getestet werden. Dann wird sich zeigen, ob auch wirklich alle Grafiken und Kombinationen untereinander mit der Visualisierungsvorschrift der SLD ausreichend sind oder welche Änderungen sinnvoll sind. Dies wird in einer aktuellen Diplomarbeit (Neubauer 2007) im Rahmen des Projektes 3D-GDI Heidelberg ([www.heidelberg-3d.de](http://www.heidelberg-3d.de), [www.gdi-3d.de](http://www.gdi-3d.de)) (Schilling et al 2007) umgesetzt.

## 6. Literatur

- Andrae, C., Müller, M., Schäfer, K., Weiser, A., Zipf, A. (2007 in print): OGC Portrayal Standards - WMS und SLD. Reihe: OpenGIS Essentials - Spezifikationen des OGC im Überblick. Hrsg: Andrae, C., Fitzke, J., Zipf, A. Wichmann Hüthig Verlag. Heidelberg. CityGML: [www.citygml.org](http://www.citygml.org)
- Coors, V. und Zipf, A. (Hrsg.)(2004): 3D-Geoinformationssysteme. Grundlagen und Anwendungen. Wichmann - Hüthig Verlag. Heidelberg. 525 Seiten.
- SLD: SLD-Spezifikation (OGC): <http://www.opengeospatial.org/docs/02-070.pdf>
- Schilling, A., Basanow, J., Zipf, A. (2007 accepted): VECTOR BASED MAPPING OF POLYGONS ON IRREGULAR TERRAIN MESHES FOR WEB 3D MAP SERVICES. 3rd International Conference on Web Information Systems and Technologies (WEBIST). Barcelona, Spain. March 2007.
- Neubauer, S. (2007 Arbeitstitel): Visualisierungsvorschriften fuer 3D Stadtmodelle auf Basis von SLD. Diplomarbeit. FH Mainz.
- Nonn, U., Zipf, A. (2007 submitted): Metadaten für 3D-Stadtmodelle - Untersuchung der Eignung von ISO 19115 und Möglichkeiten der Erweiterung.
- Open Source ArcMap2SLD Tool Homepage: <http://arcmmap2sld.geoinform.fh-mainz.de/>
- Müller Markus: Workshop IV [www.mediakomm-transfer.de/.../Veranstaltungen/X-Planung/Workshop-Dokument\\_2010,property=dokument.pdf](http://www.mediakomm-transfer.de/.../Veranstaltungen/X-Planung/Workshop-Dokument_2010,property=dokument.pdf)