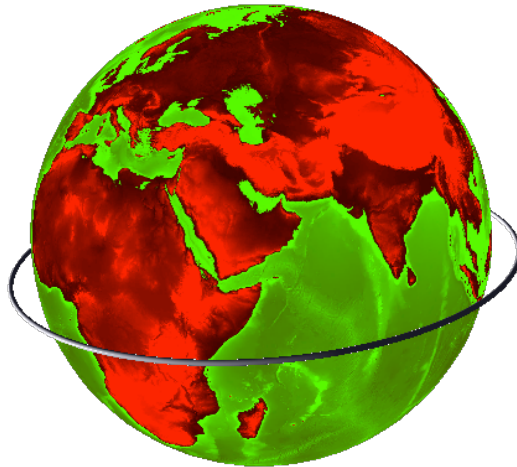


BEUTH HOCHSCHULE FÜR TECHNIK BERLIN
University of Applied Sciences

Computergrafik II

Hinweise zu Aufgabe 5 – Weltanschauung

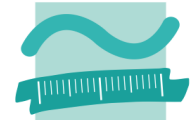


Bachelor Medieninformatik
Wintersemester 2011

Prof. Dr.-Ing. Hartmut Schirmacher
<http://schirmacher.beuth-hochschule.de>
hschirmacher@beuth-hochschule.de



Screenshot



BEUTH HOCHSCHULE
FÜR TECHNIK
BERLIN
University of Applied Sciences



Darstellung & Animation

- ☒ Äquator-Ring anzeigen
- ☒ In der Nacht Lichter darstellen
- ☒ Wolkenschichten
- ☐ Sonne wandert mit Grad/s um den Äquator

Darzustellender Monat:

Kamera / Projektion

☒ Perspektivische Projektion

Öffnungswinkel: Grad

Tiefenbereich (Z) von: bis

☐ Orthographische Projektion

links: rechts:

unten: oben:

vorne: hinten:



NASA-Projekt *Blue Marble*

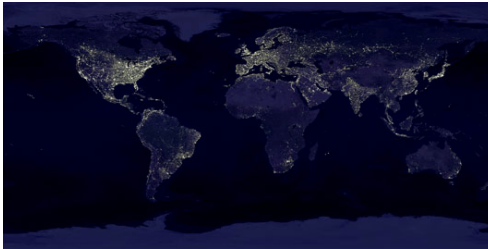
- Original-Bild *Blue Marble*
 - Aus Apollo 17 fotografiert
 - Aufnahme vom 7. Dezember 1972
- NASA-Projekt *Blue Marble: Next Generation*
 - <http://earthobservatory.nasa.gov/Features/BlueMarble/>
 - http://visibleearth.nasa.gov/view_cat.php?categoryID=1484
 - Aus sehr vielen Einzelbilder zusammengesetzte Karten der Erde
 - Verschiedene Daten verfügbar (Farbe, Topographie, Wolken, ...)
 - Frei verwendbar
- Für die Übungsaufgabe
 - ... Wurden einige ausgewählte Texturen kopiert, auf $2^N \times 2^M$ skaliert und zur Deckung gebracht



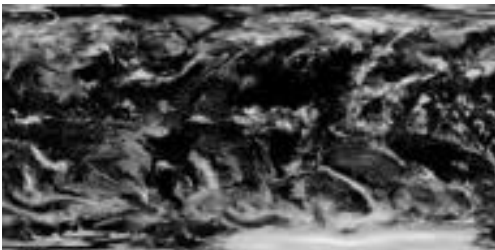
Bilddaten aus dem „Blue Marble“ Projekt (1)



- Topographie, Wasser und Meeres-Eis, Januar ... Dezember
 - <http://visibleearth.nasa.gov/view.php?id=57730>
 - [...] these images are based on surface observations collected from June through September 2001 and combined, or composited, every eight days to compensate for clouds that might block the sensor's view of the surface on any single day.



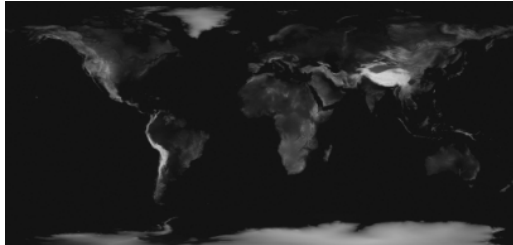
- Lichter bei Nacht
 - <http://visibleearth.nasa.gov/view.php?id=55167>
 - *Global city lights, derived from 9 months of observations from the Defense Meteorological Satellite Program, are superimposed on a darkened land surface map.*



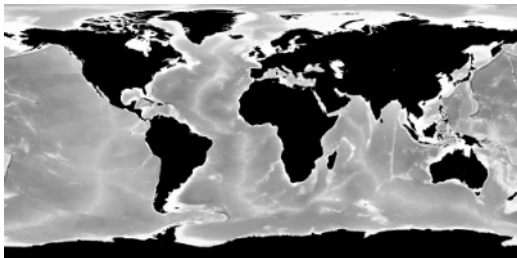
- Wolken
 - <http://visibleearth.nasa.gov/view.php?id=57747>
 - *The cloud image is a composite of two days of imagery collected in visible light wavelengths and a third day of thermal infra-red imagery over the poles.*



Bilddaten aus dem „Blue Marble“ Projekt (2)



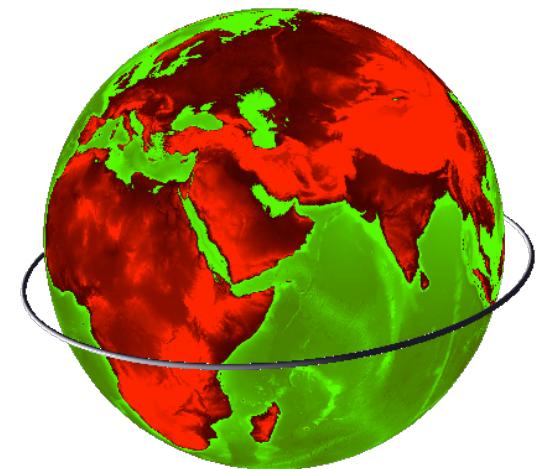
- Topographie
 - Höhe über dem Meeresspiegel, als Graustufen kodiert



- Bathymetrie
 - Tiefe unter dem Meeresspiegel, als Graustufen kodiert

Lizenz zur Verwendung der Bilddaten:

Anyone using or republishing Blue Marble: Next Generation please credit "NASA's Earth Observatory."



rot = Höhe über NN
grün = Tiefe unter NN



- Security einiger Browser verhindert das Laden von Texturen, die im lokalen Dateisystem liegen.
 - Sogenanntes *Cross-Domain-Ressourcen-Problem*
- Lösung 1:
 - Eigenen Webserver aufsetzen (i.d.R. aufwändig)
- Lösung 2:
 - Security im Browser ausschalten
 - Chrome: starten mit `--allow-file-access-from-files`
 - Firefox:
 - Unter Mac OS X scheint einfach so zu funktionieren.
 - Unter Windows tippen Sie „about:config“ in die URL-Leiste, um an die Einstellungen von Firefox zu kommen. Dann stellen Sie den Wert von **security.fileuri.strict_origin_policy** auf **false**.
 - Achtung: dies sollte man nur in einer separaten Instanz des Browsers machen; damit sollte man keine unbekannten Scripte ausführen!



Crashkurs: Texturobjekte im Framework von Aufgabe 5

- In `initScene()` :

```
// Texturobjekt anlegen
this.myTexture1 = new Texture2D(gl, „textures/some_texture.jpg“);
this.myTexture2 = new Texture2D(gl, „textures/some_other_tex.jpg“);
```

- In `drawScene()` :

```
// Texturen gleichzeitig aktivieren, via Texture Units 0 und 1
this.myTexture1.makeActive(program, „mySampler1 “, 0)
this.myTexture2.makeActive(program, „mySampler2“, 1)
```

- Im Fragment Shader (Vertex Shader entsprechend ergänzen...):

```
varying vec2 texCoord;
uniform sampler2D mySampler1;
uniform sampler2D mySampler2;
void myIlluminationFunction(...) {
    vec3 color1 = texture2D(mySampler1, texCoord).rgb;
    float x = texture2D(mySampler2, texCoord).r;
}
```

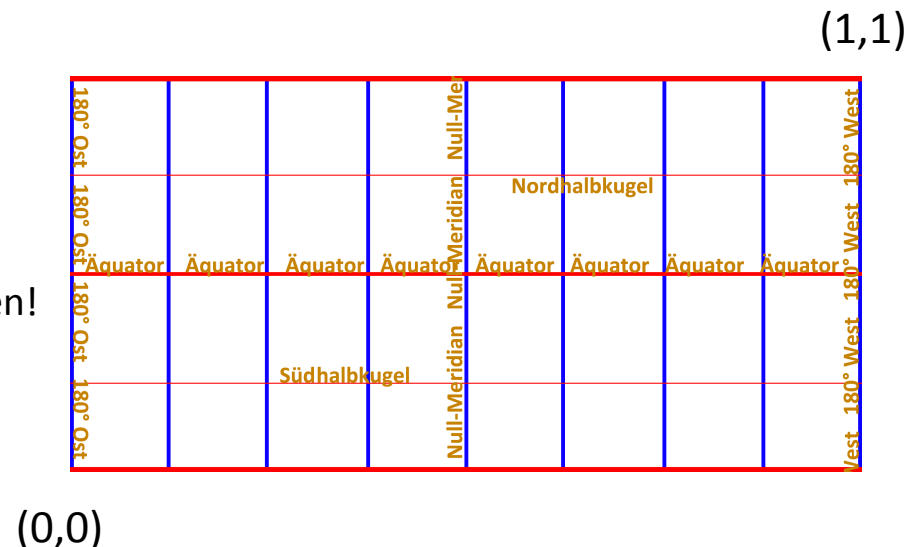


Projektionseigenschaften der Blue-Marble-Karten

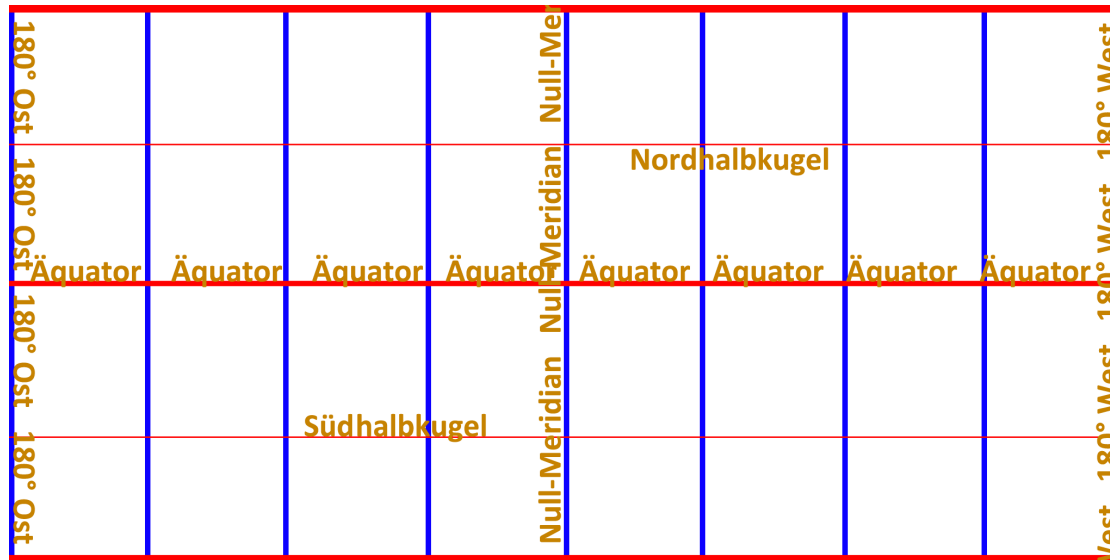
- Quadratische *Plattprojektion* oder *Rektangularprojektion*
 - http://de.wikipedia.org/wiki/Quadratische_Plattkarte
 - Auf der Karte ist der Abstand von jeweils zwei benachbarten Breitengraden und zwei benachbarten Längengraden identisch
 - Bild-Koordinaten (u,v) entsprechen direkt der geographische Länge und Breite (λ , φ)

- Verwendung der Blue-Marble-Karten als Texturen

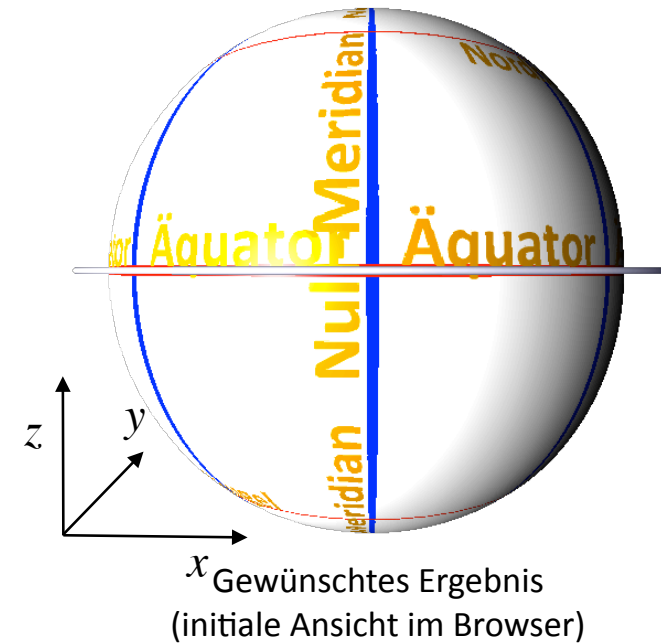
- Jedem Vertex muß die richtige Stelle in der Karte (= Texturkoordinate) zugeordnet werden
- Für jeden Vertex berechne den Längen- und Breitengrad
- Relativ einfach, wenn bei der Erzeugung der Kugel solche Koordinaten (u,v) bereits verwendet werden!
- Normalisiere so, daß die Texturkoordinaten den Bereich (0,0) bis (1,1) abdecken.



Test-Textur test_world_texture.gif



Textur test_world_texture.gif



- Für die Aufgabe sollen die Texturkoordinaten wie folgt ausgerichtet sein:
 - Initial schaut die Kamera in Richtung +Y; Z zeigt nach oben, X nach rechts (siehe lookat()-Aufruf in main.js)
 - Der Äquator verläuft in der Ebene $Z=0$;
 - Der Null-Meridian ist bei $X=0$ auf der Vorderseite der Kugel
- Die Texturkoordinaten müssen ggf. vertauscht, gespiegelt, verschoben werden...

