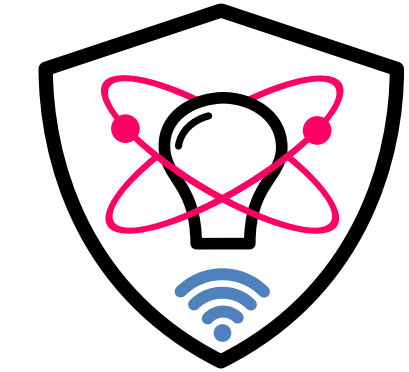


ИКТ в НОС

Текстури

Тема №21

Увод в текстурите

Текстура

От английски *texture*

- Структура (най-често повърхностна)
- Грапавината, усещана при докосване

В компютърната графика

- Изображение, наложено върху обект
- Приема формата на обекта
- Пиксел от текстура се нарича *тексел* (*texel*)

Използване

Структура на материали

- Дървен паркет с чепове, надраскана метална повърхност, кожа с рани, грайфери, …

Повторяеми елементи

- Тухлена стена, плочки в баня, прозорци на сграда, …

Фонови картинки

- Небе с облаци, звезди, карти, …

Структура и координати

Форма

- Текстурата е винаги правоъгълна картинка

Координати

- Собствена координатна система
- Текстура в своята координатна система е от (0,0) до (1,1)

Singapore river skyline building
Автор: cegoh, лиценз: CCo Public Domain
http://pixabay.com/en/singapore-river-skyline-building-243669

POT/NPOT текстури

- За пълна функционалност размерите в пиксели са 2^n x 2^m
- Ако не са, ще има само базова функционалност
- POT-текстура (*power-of-two*) NPOT-текстура (*non-POT*)

POT 512 x 256

NPOT 640 x 359

POT 256 x 256

Зареждане на текстура

Опит с текстура

Задача

- Да се създаде куб
- Кубът да е с текстура
- За текстура ползваме същата картинка, но смачкана до размери 128 x 128 пиксела

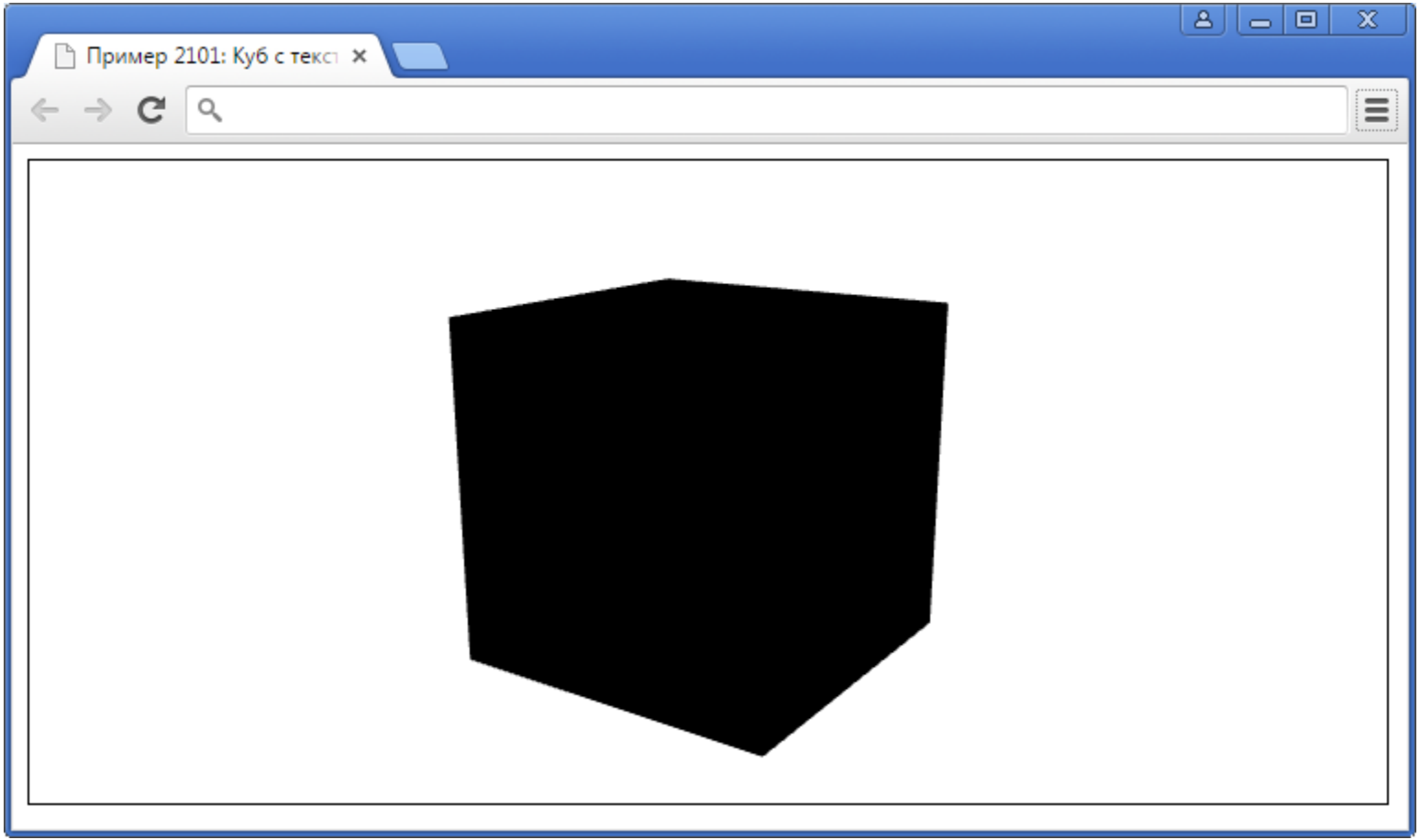
Решение

- 2D и 3D графичните обекти в СУИКА имат свойство image
- В него се записва обект, който съдържа картинка за текстура и допълнителни данни
- Създаване на текстурен обект
- Използва се класът Image с параметър на конструктора път до файла с картинката

Пробата

- Ще я направим първо с Chrome

```
a = cube([0,0,0],30);
a.image = new Suica.Image('texture.jpg');
```



ПРОБА

Проблем

Проблем с текстурата

- Вместо с картинка, кубът е изцяло черен
- Това е защита, с която се контролира каква картинка може да се ползва за текстура и коя не

CORS

- Съкращение на Cross-origin Resource Sharing
- Това е механизъм, с който се контролира кой ресурс от един домейн може да се използва от друг домейн

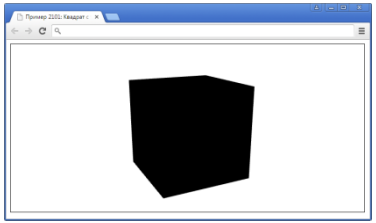
Пояснение

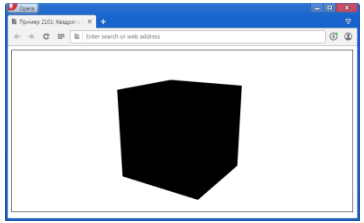
- В Chrome четенето на локален файл се счита за нарушение на CORS и е забранено
- При опит за четене получаваме грешка в JS конзолата:

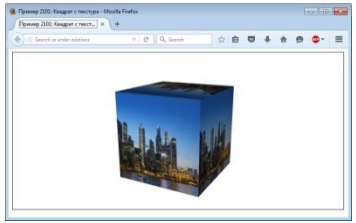
  Uncaught SecurityError: The cross-origin image at file:///SUICA-21 Textures/Example-2101 Textured square/texture.jpg may not be loaded.

Други проби

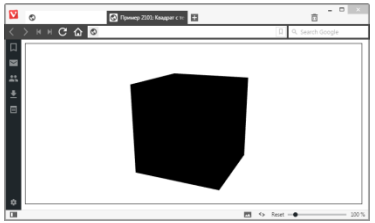
- Проби с различни браузъри
- Ефектът е непредсказуем – т.е. не може да се разчита на кой браузър се ползва


Chrome


Opera


Firefox


IE


Vivaldi

Възможни решения за CORS

Друг браузър (не се препоръчва)

- Избира се браузър, на който тръгва
- Но: не е сигурно дали на друг компютър браузърите ще реагират по същия начин

Други настройки (не се препоръчва)

- Някои браузъри позволяват с настройки да се премахне CORS защитата
- Но: снижава се нивото на сигурност, особено ако забравите да възстановите настройките

Препоръчвани решения

Решение №1 – онлайн съхранение

- Файлът с програмата и картинката са на един и същ домейн, онлайн, някъде из мрежата
- Преимущества

  При готово приложение това е нормалното му състояние (да е някъде качено онлайн)
- Недостатък

  За разработването на приложението ще се изисква достъп до качване на файлове в домейн

Решение №2 – вградени картинки

- Използва се технологията Data URI
- В уеб адрес вместо адрес към ресурс се поставя самият ресурс (в текстов вид)
- Преимущества
  - Не се нарушава CORS дори и при локално ползване
  - Картинките не са в отделни файлове, а за вградени в HTML файла
- Недостатъци
  - Браузърите имат максимална поддържана дължина на URL адрес
  - Картинките не се буферират

Решение

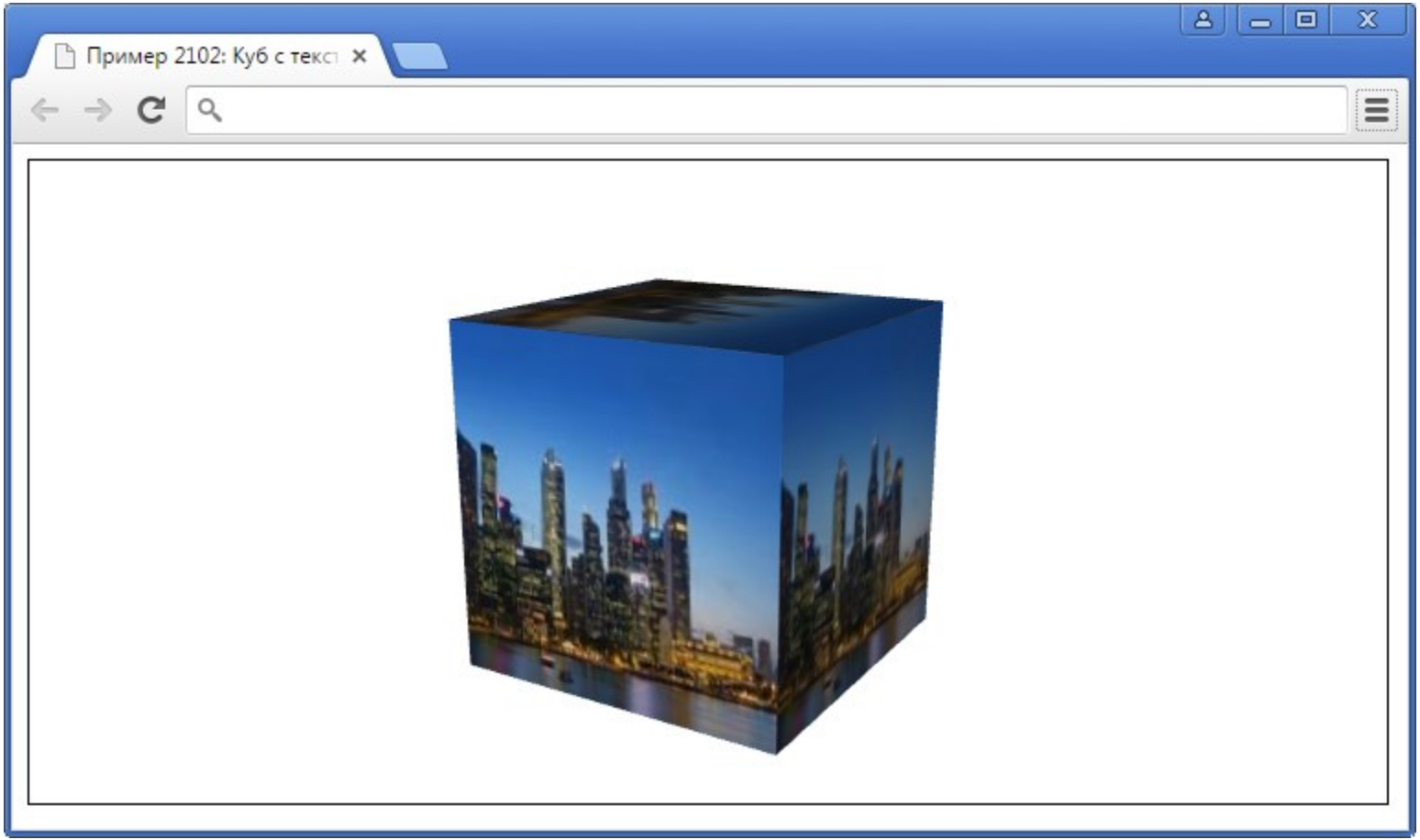
- Използваме онлайн конвертор до Data URI

  Примерно този, но има и много други:

  websemantics.co.uk/online_tools/image_to_data_uri_convertor

- Полученият адрес използваме вместо адрес на картинка

```
a.image = new Suica.Image('
/9j/4AAQSkZJRgABAgEASABIAAD/4RIjRXhpZgAATU0AKgAAAA
gABwESAAMAAAABAAEAAAEaAAUAAAABAAAAYgEbAAUAAAABAAAA
agEoAAMAAAABAAIAAAExAAIAAAAbAAAAcgEyAAIAAAAUAAAAjY
dpAAQAAAABAAAApAAAANAAAABIAAAAAQAAAEgAAAABQWRvYmUg
UGhvdG9zaG9wIENTIFdpbmRvd3MAMjAxNToxMDoyOCAxNzozNz
:
CuV6UWsb+6hw8PvpuHbS24cq4lv3Gu81NqPD7BXPt91f/9k=');
```

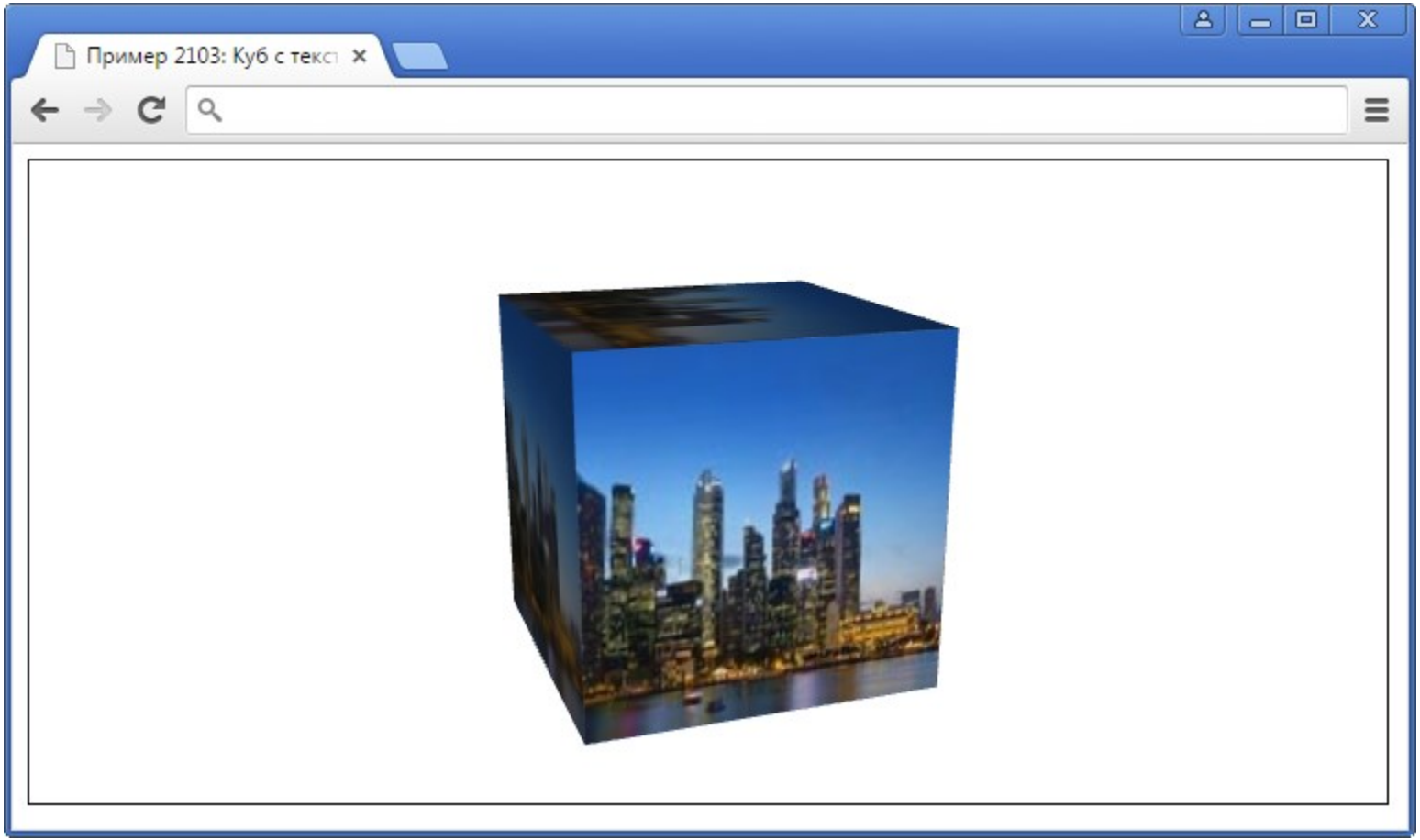


ПРОБА

Решение №3 – локален уеб сървър

- Симулираме онлайн достъп (http или https), но локално
- Има редица подходящи малки сървърчета, примерно: QuickPHP, Mongoose и други
- Вместо адрес на картинка file://texture.jpg се ползва http://localhost/texture.jpg или просто texture.jpg

```
a.image = new Suica.Image('http://localhost/Examp
        le-2103%20Textured%20cube%203/texture.jpg');
```



ПРОБА

Указание за работа

Използване на QuickPHP Web Server

- Стартира се QuickPHP.exe
- В опциите се избира binding address да е 127.0.0.1, port да е 80, allow directory listing да е включено, а root folder да е работната директория
- Натиска се [Start]
- Директорията е достъпна на адрес http://localhost
- След приключване на работата сървърът се спира

Използване на Mongoose Web Server

- Стартира се mongoose-free-5.6.exe
- В опциите се избира document_root да е работната директория, а listening_port да е 80
- Директорията е достъпна на адрес http://localhost
- След приключване на работата сървърът се спира

Използване на друг уеб сървър

- Действа се според указанията му

ИКТ в НОС

Безшевни текстури

Безшевност

Безшевна текстура

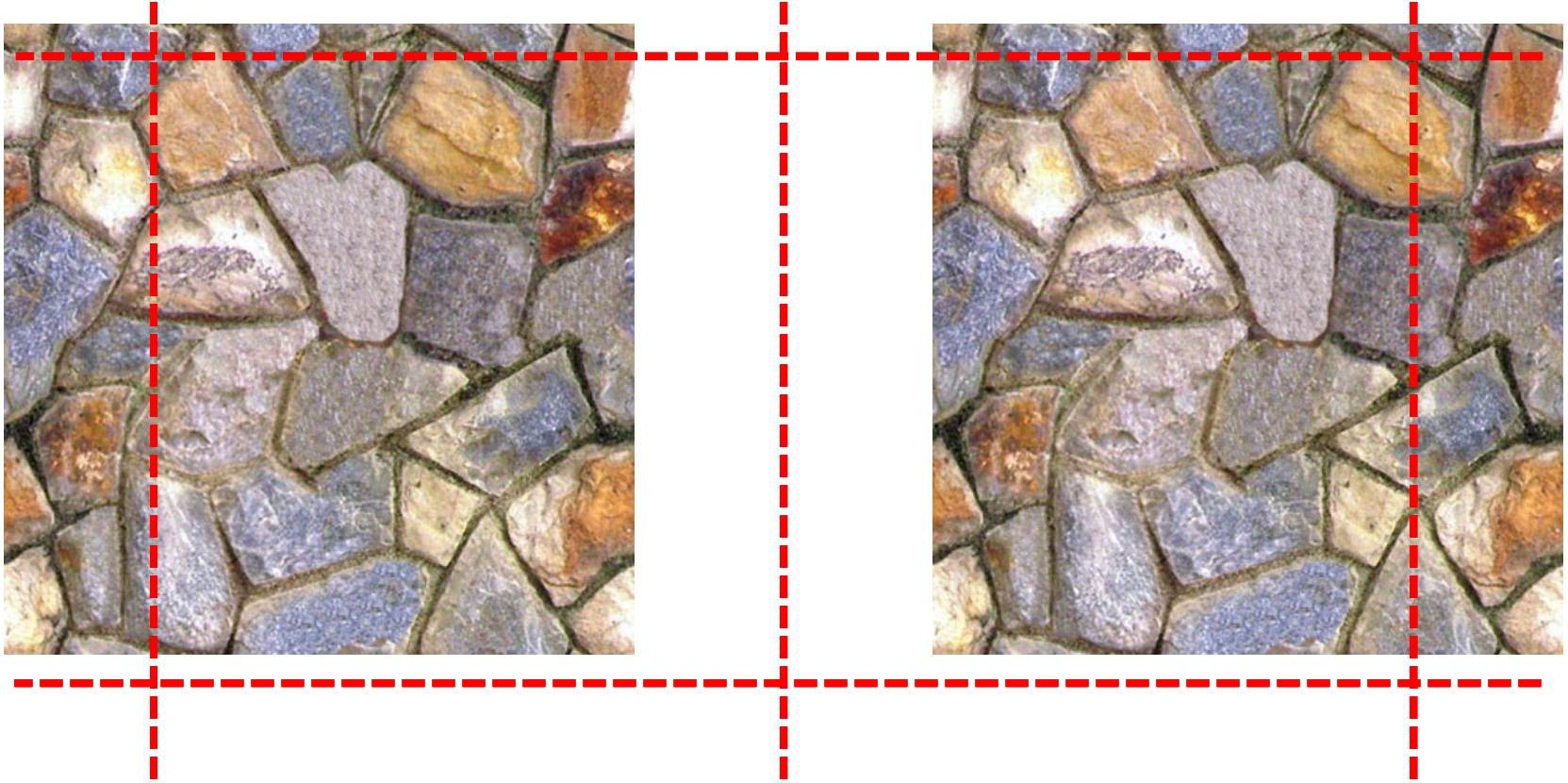
- На английски: *seamless*
- Текстура се долепя до себе си без видим ръб
- Получава се непрекъсната шарка

Употреба

- За големи еднотипни повърхности
- Стена, тротоар, пръст, паркет, ливада …

Как действа

- С измама – текстурна непрекъснатост



Използване на текстури

Примери

Пример №1: Тухлена стена

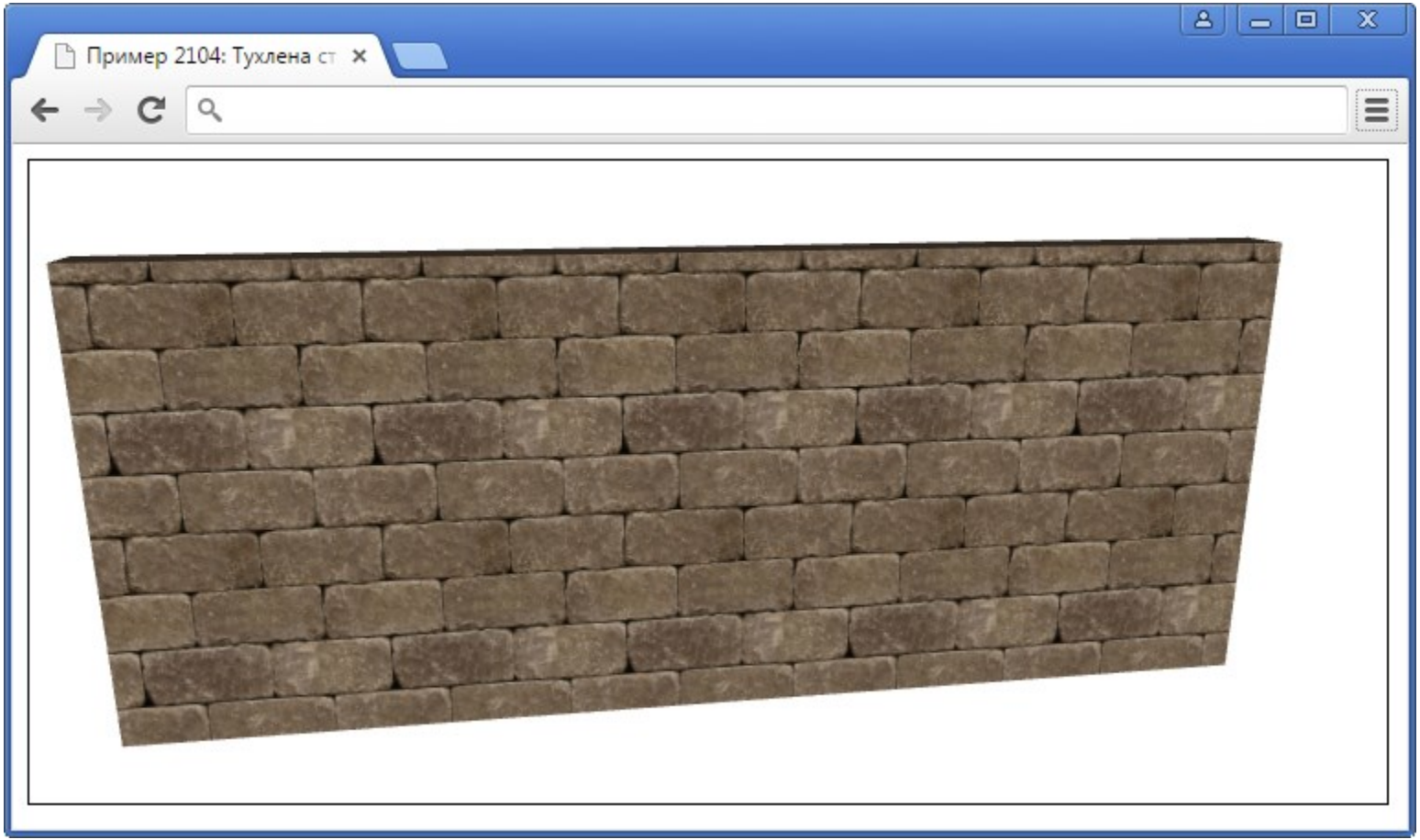
- Имаме текстура на фрагмент от тухлена стена
- Текстурата е безшевна

Seamless Brick 09-28-1001
Автор: texturemate, лиценз: Public Domain
http://texturemate.com/content/free-texture-seamless-brick-09-28-1001

Решение

- Вместо да създаваме много кубчета с тази текстура, създаваме един паралелепипед с размери 100 x 40 x 5
- Свойството image има подсвойство scale, което показва колко копия на текстурата да се сместят; в случая искаме хоризонтално да са 5 копия, вертикално да са 2
- И още: няма нужда да пишем localhost, картинката е в същата уеб директория като HTML файла, няма CORS и можем да я цитираме като че ли е онлайн

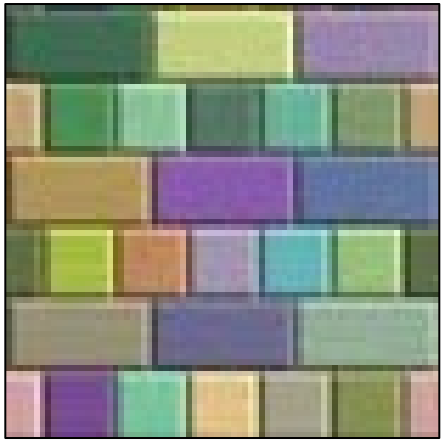
```
a = cuboid([0,0,0],[100,5,40]);
a.image = new Suica.Image('bricks.jpg');
a.image.scale = [5,2];
```



ПРОБА

Пример №2: Под с плочки

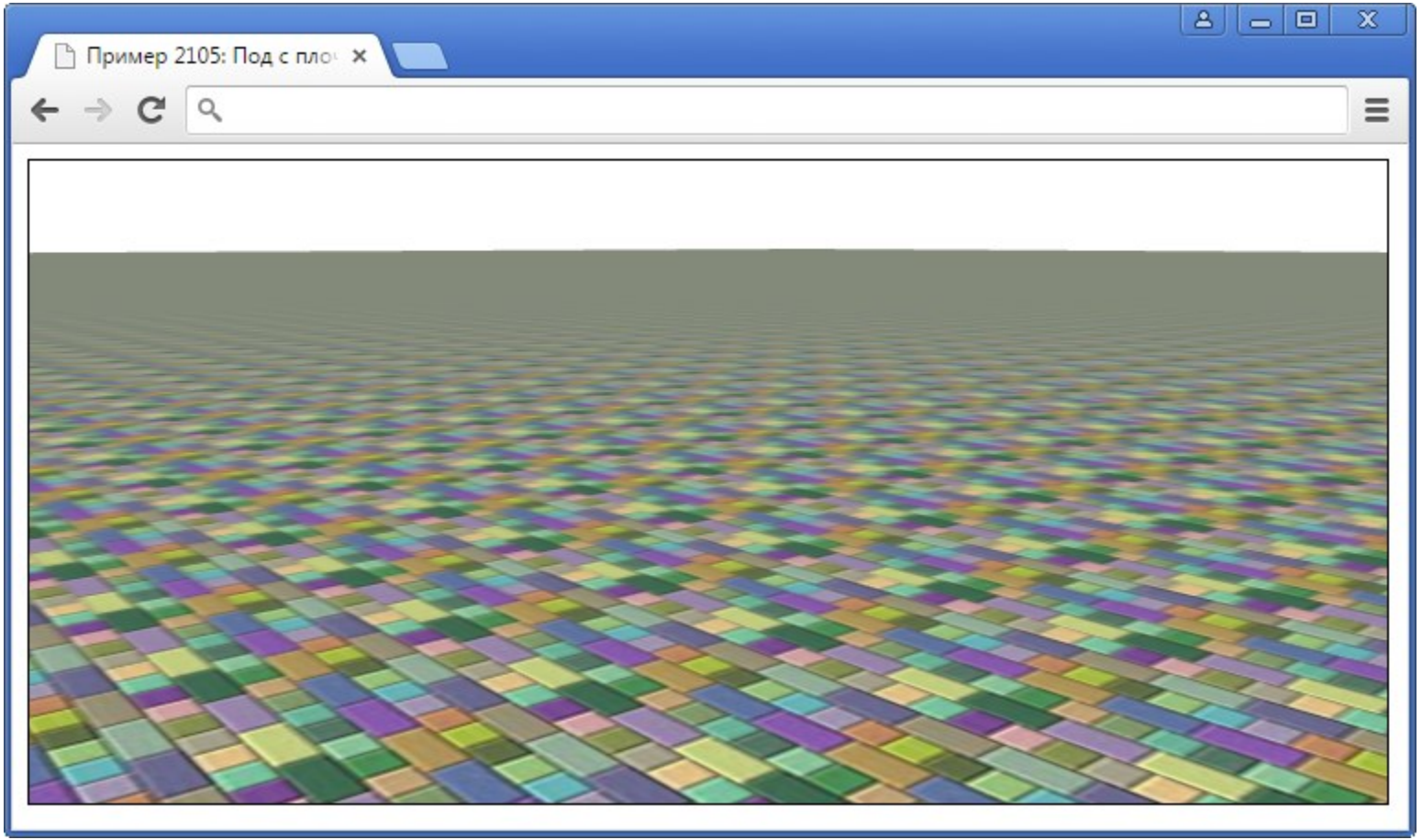
- Имаме текстура на плочки на под
- Подът ще е до хоризонта
- Поради малкият размер на шарките, позволяваме си текстурата да не е кристално ясна



Решение

- Създаваме огромен под 5000 x 5000
- Текстурата се повтаря по 400 пъти
- Поради ниската гледна точка подът е силно скосен и осветяването му е слабо – вижда се силно потъмнен
- За да се възстанови оригиналният текстурен цвят се изключва светлината

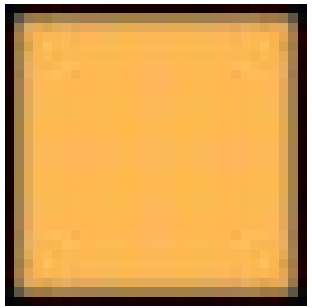
```
a = cuboid([0,0,0],[5000,5000,1]);
a.light = false;
a.image = new Suica.Image('tiles.jpg');
a.image.scale = [400,400];
```



ПРОБА

Пример №3: Куб от кубчета

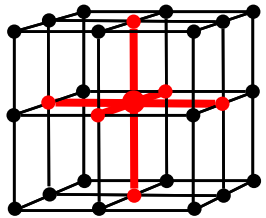
- Да направим пробит куб с текстура по стените му
- Кубът да е сглобен от стандартни кубчета
- На тази конструкция да можем да променяме броя на кубовете, без да променяме броя на обектите



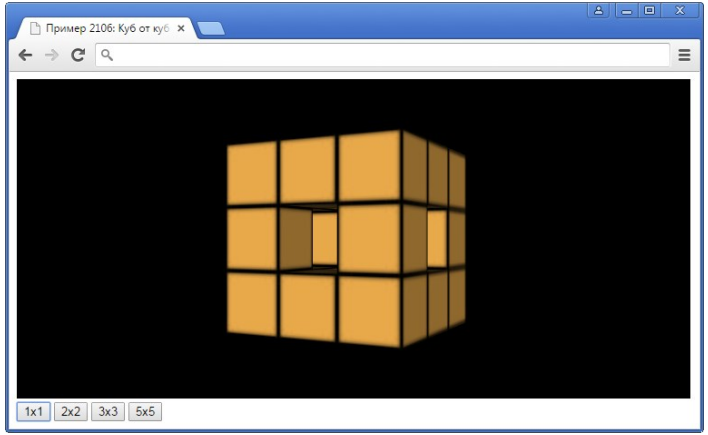
Текстура за всяка стена на куба

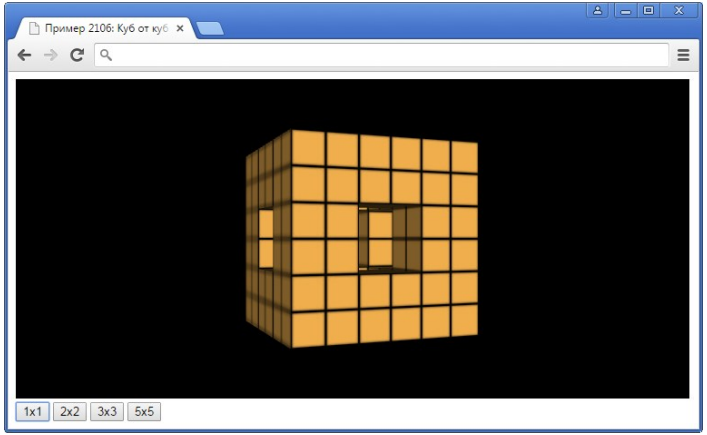
Решение

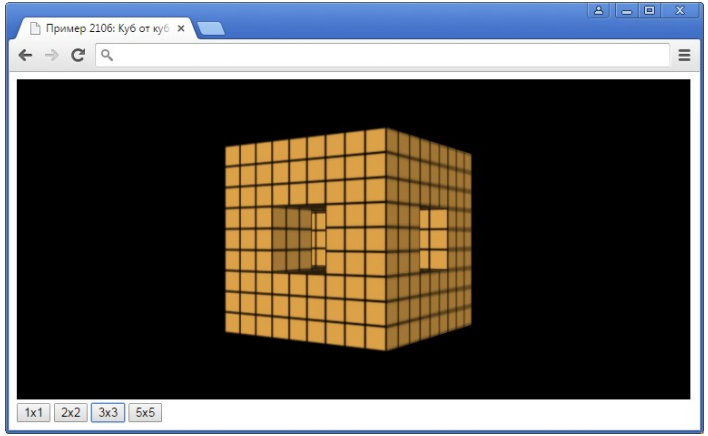
- Правим кубчета в мислен 3x3x3 куб, без централните кубчета
- Затова прескачаме 7-те кубчета на разстояние 1 до центъра
- Текстурата зареждаме еднократно

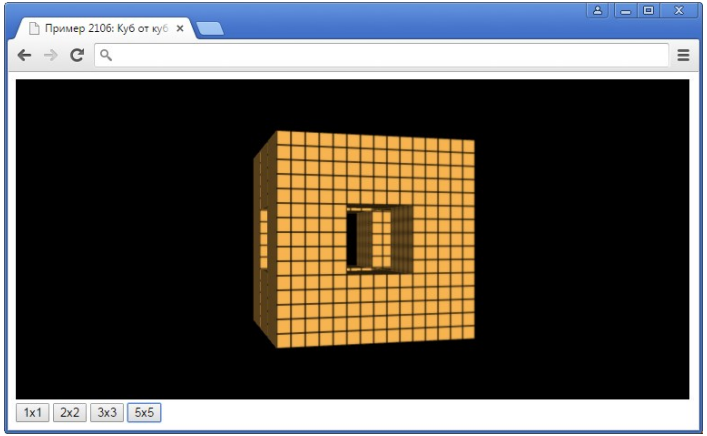


```
style = {image: new Suica.Image('block.jpg')};
for (var x=-1; x<2; x++)
for (var y=-1; y<2; y++)
for (var z=-1; z<2; z++)
   if (x*x+y*y+z*z>1.1)
      cube([x,y,z],1).custom(style);
```


Пример 210б: Куб от куб
1x1
2x2
3x3
5x5


Пример 210б: Куб от куб
1x1
2x2
3x3
5x5


Пример 210б: Куб от куб
1x1
2x2
3x3
5x5


Пример 210б: Куб от куб
1x1
2x2
3x3
5x5

ПРОБА

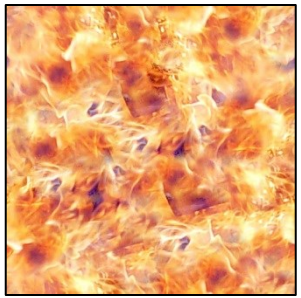
Множество текстури

Множество текстури

Задача

- Да се рисуват няколко куба
- Всеки да има своя текстура



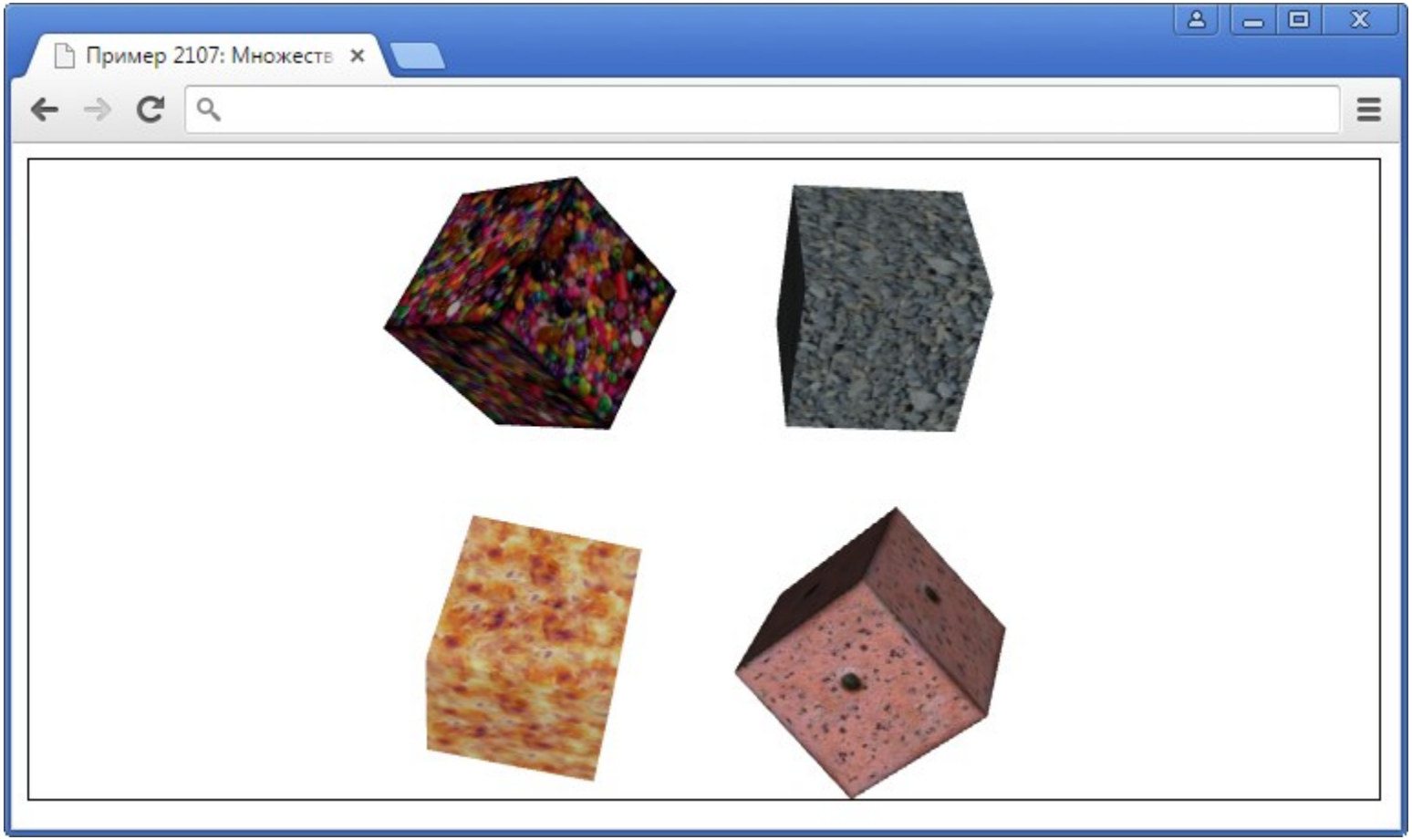
Текстурите са базирани на Rock Textures. Rusty Hinge, Colorful Beads, Fire Pit Flames
Автор: TurboPhoto, лиценз: Public Domain, http://www.turbophoto.com/Free-Stock-Images

Решение

- Всеки куб има различна инстанция на Image

```
a = [];

a[0] = cube([+20,+20,0],20).custom(
        {image: new Suica.Image('texture1.jpg')});
a[1] = cube([+20,-20,0],20).custom(
        {image: new Suica.Image('texture2.jpg')});
a[2] = cube([-20,+20,0],20).custom(
        {image: new Suica.Image('texture3.jpg')});
a[3] = cube([-20,-20,0],20).custom(
        {image: new Suica.Image('texture4.jpg')});
```



ПРОБА

Смяна на текстура

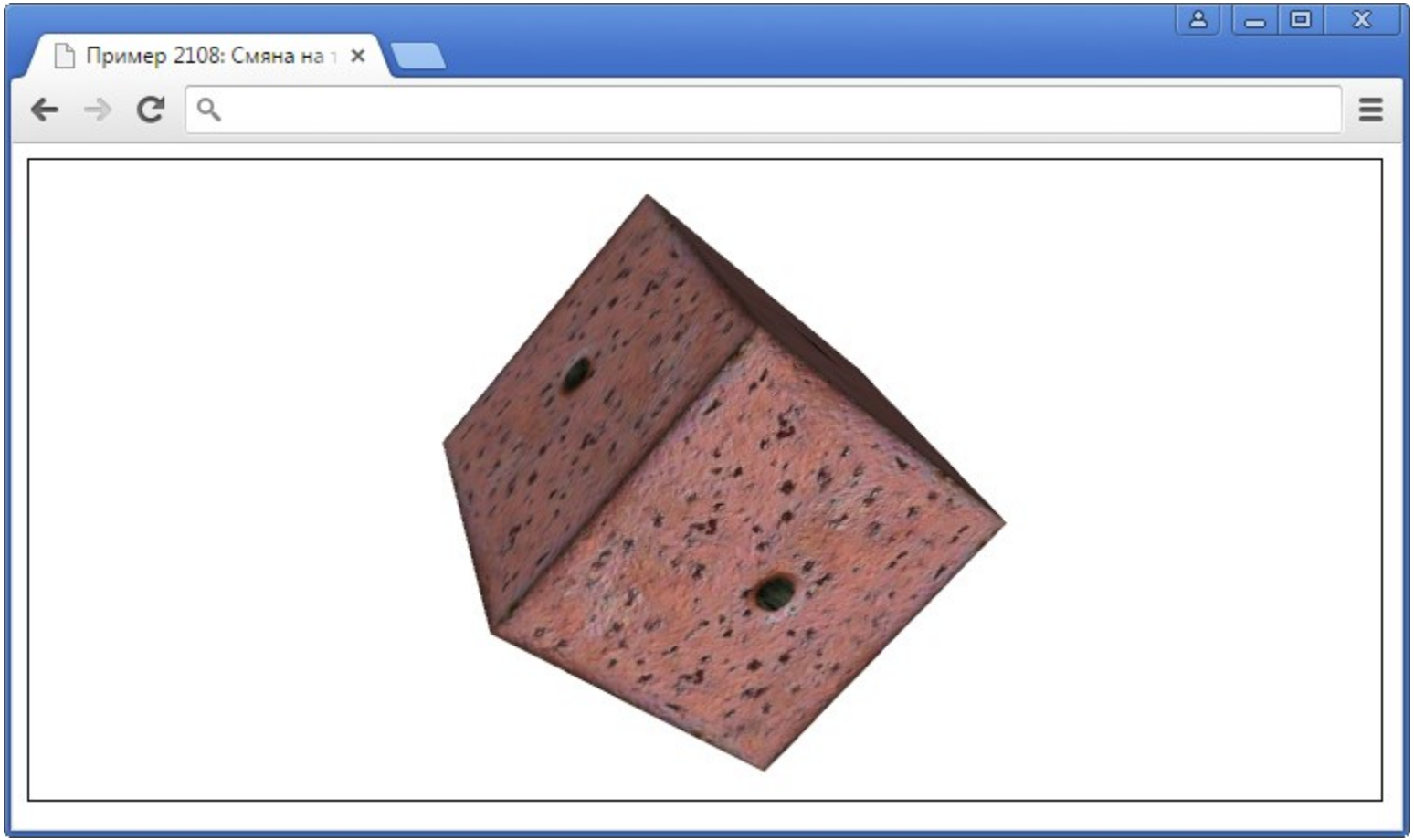
Нова задача

- Пак имаме няколко текстури
- Но само един куб
- Искаме кубът периодично да си сменя текстурата

Решение

- Зареждаме 4 текстури в масив
- В анимационния цикъл сменяме свойството image на куба

```
image = [];
for (var i=0; i<4; i++)
   image[i]=new Suica.Image('texture'+(i+1)+'.jpg');
:
function rotate()
{
   a.spin = Suica.time;
   a.focus = [Math.sin(Suica.time),...];
   a.image = image[Math.floor(Suica.time)%4];
}
```



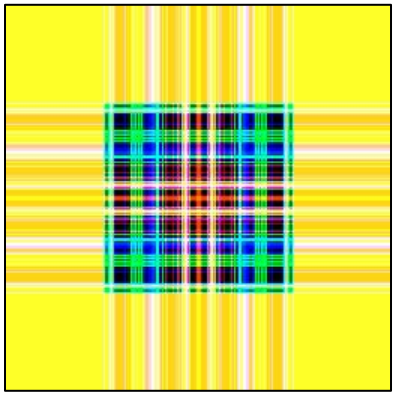
ПРОБА

Други свойства

Отместване на текстура

Свойство offset

- Определя отместване на текстурата
- В комбинация със scale променя коя част от текстурата се показва върху обекта

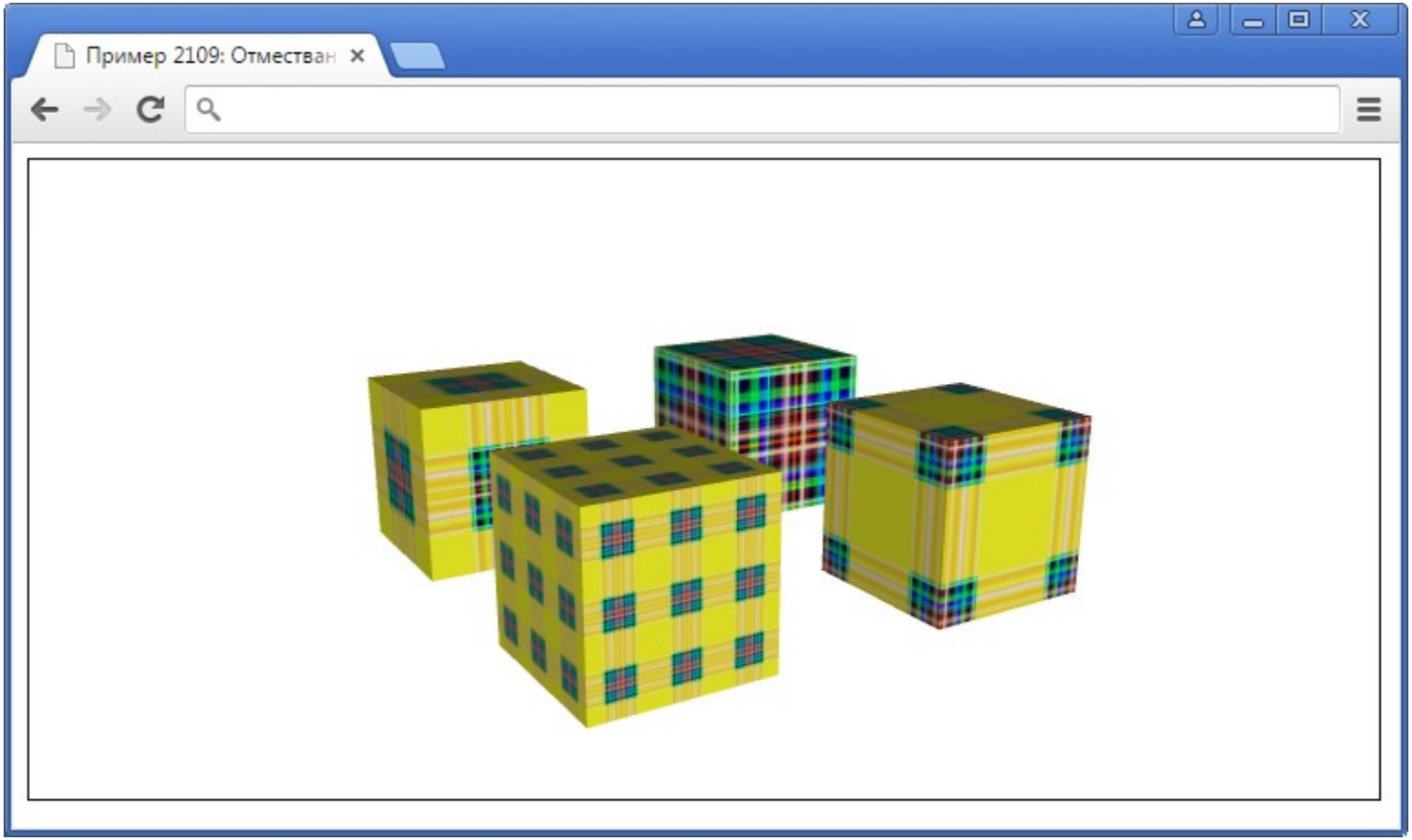


Примери

- Четири комбинации на scale и offset (показани са две)
- За всяка комбинация създаваме нова инстанция на текстурата

```
a = cube([-15,15,0],15);
a.image = new Suica.Image('crosshatch.jpg');
a.image.scale = [3,3];
a.image.offset = [0,0];

a = cube([15,-15,0],15);
a.image = new Suica.Image('crosshatch.jpg');
a.image.scale = [1/2,1/2];
a.image.offset = [0.75,0.75];
```

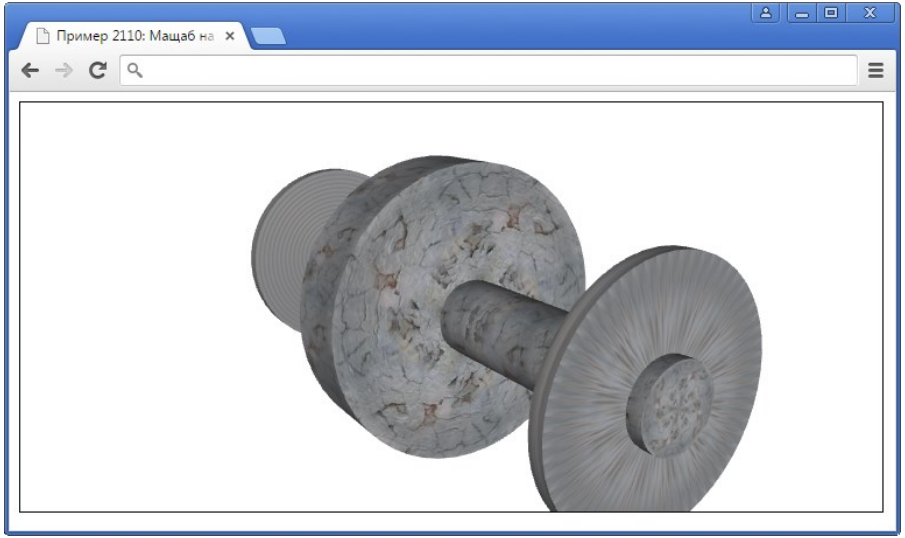


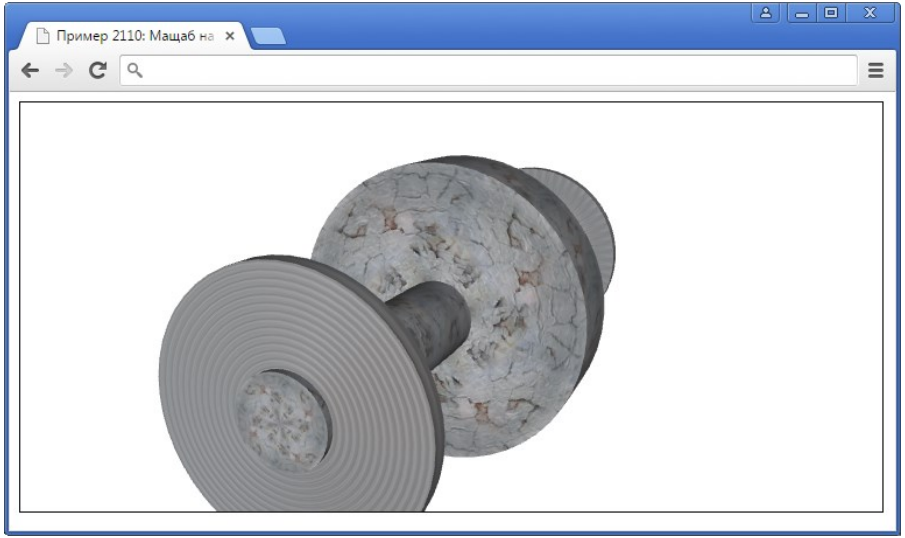
ПРОБА

Мащаб на основите

Свойство baseScale

- Обектите с основи имат отделен мащаб на текстурата за тях
- Така се постигат различни ефекти
  - Различен мащаб на основите и на околните стени
  - Концентрична шарка на основите
  - Радиална шарка на основите



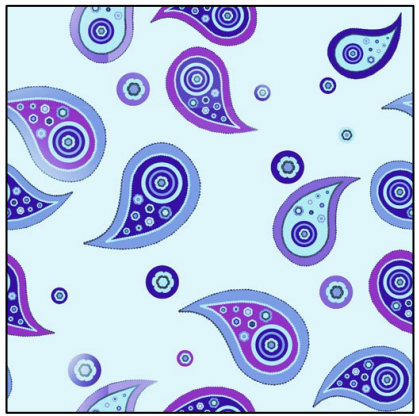


ПРОБА

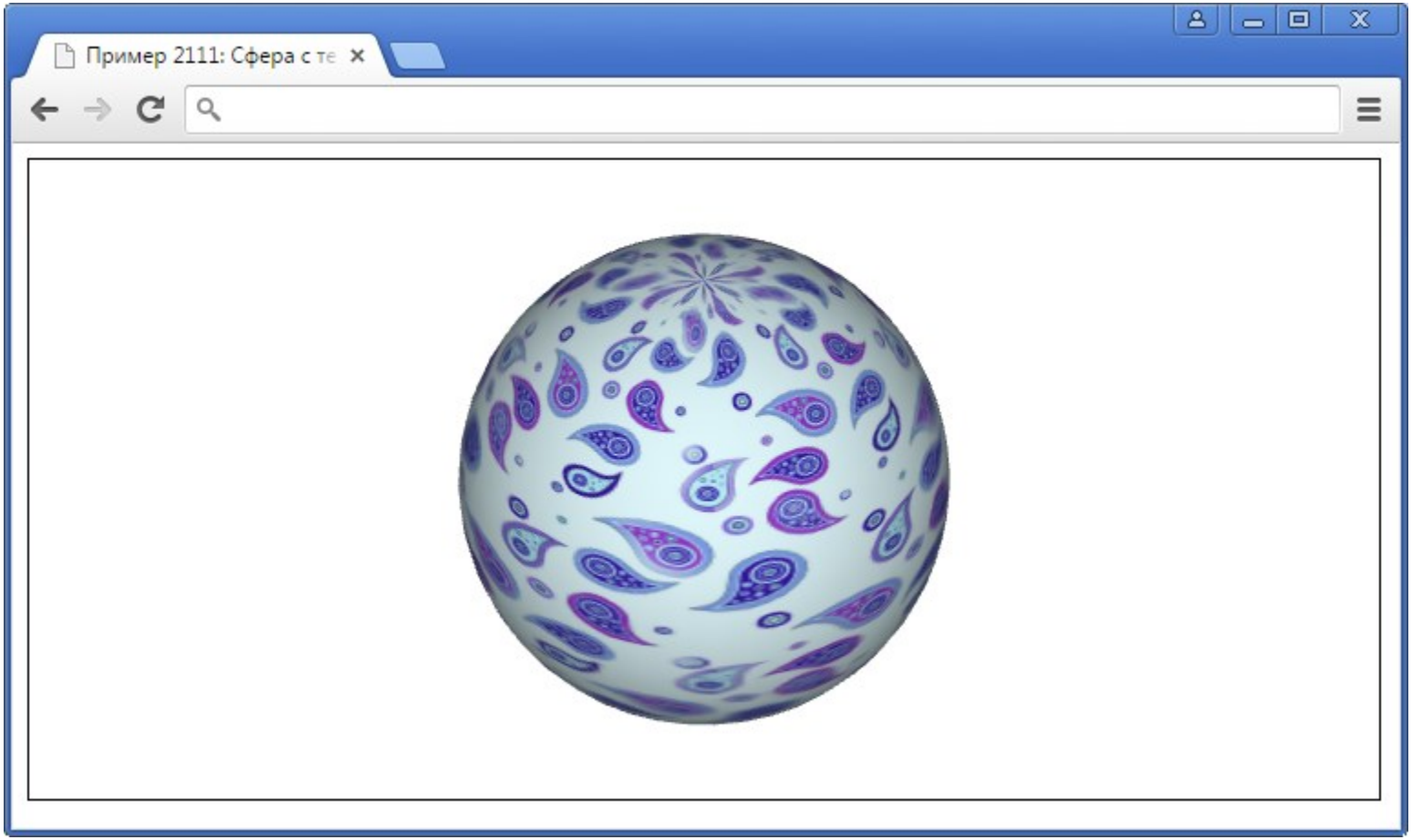
Текстура върху сфера

Основен проблем

- Топологично няма как да се насложи правоъгълна текстура върху сфера
- Получава се „свиване“ на текстурата около полюсите



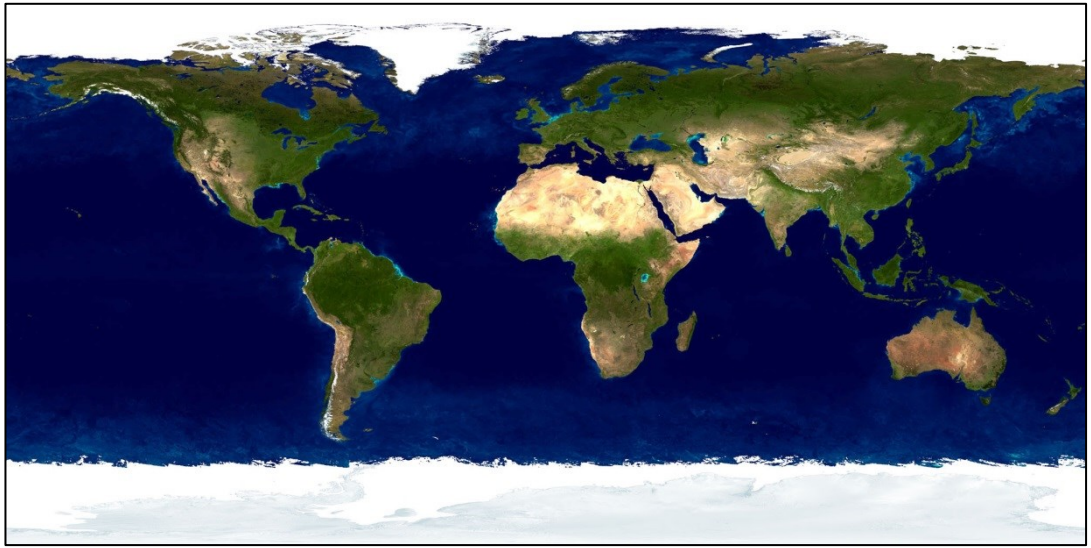
Paisley pattern blue
Автор: Пол Шермън, лиценз: Public Domain
http://www.wpclipart.com/textures/paisley/paisley_pattern_blue.png.html



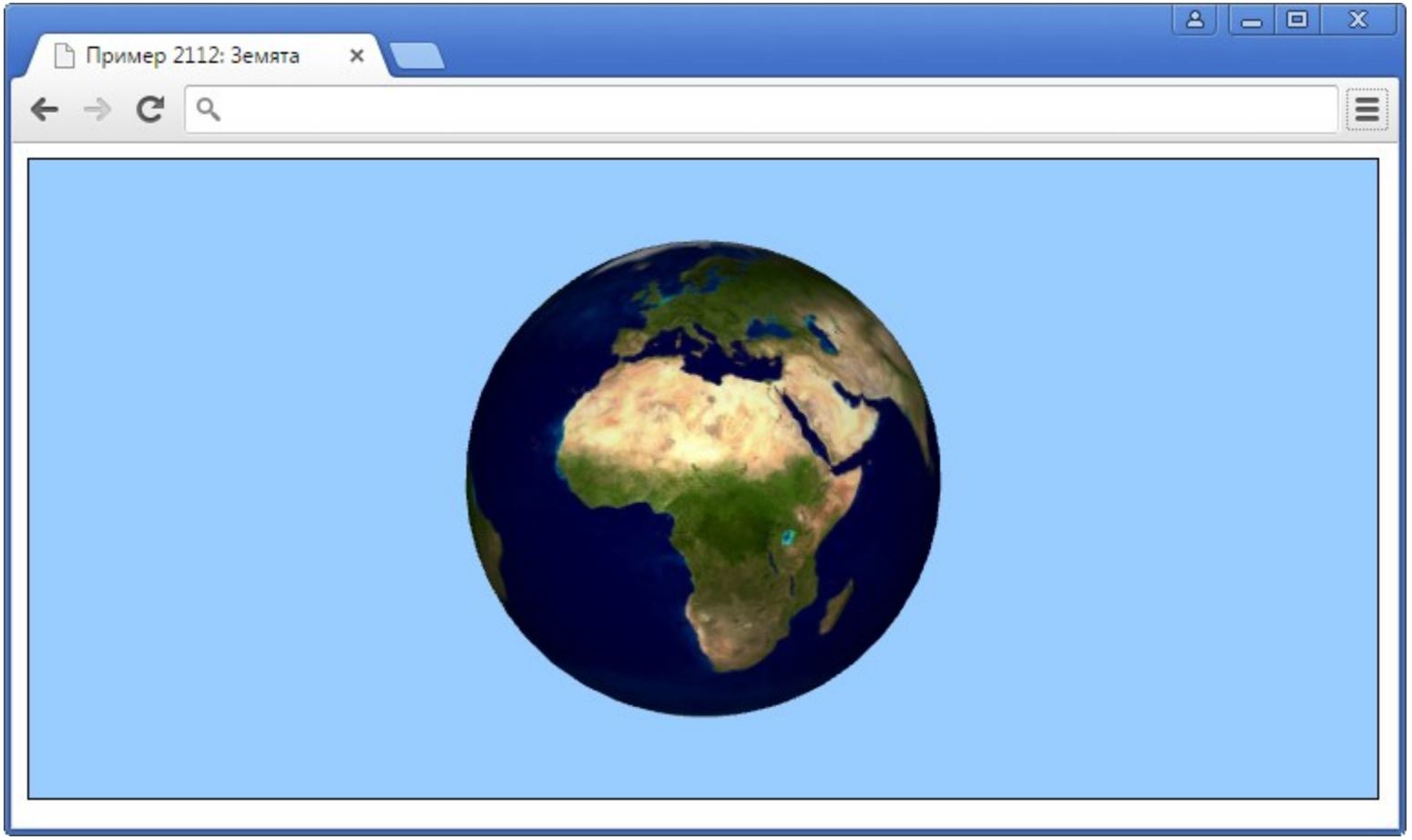
ПРОБА

Модел на Земята

- Използва се специална текстура
- Около полюсите образът е разтегнат
- Компенсира напълно свиването



The Blue Marble: Land surface, ocean color and sea ice
Автори: Reto Stöckli, Robert Simmon, NASA Goddard Space Flight Center, лиценз: Public Domain
http://visibleearth.nasa.gov/view.php?id=57730



ПРОБА

Обобщение

Текстури

Текстури в КГ

- Правоъгълни картинки, по възможност с размери $2^n x 2^m$
- Разполагат се върху графични обекти и приемат формата им
- Безшевните текстури могат да се долепят едни до други

Източник

- Четат се от файл от разбираем за браузъра формат
- Ако файлът е на друг домейн, възниква проблем с CORS
- Може данните за текстура да се закодират като Data URI
- За локални тестове може да се ползва мини уеб сървър

Създаване на текстура

- Чрез инстанция на класа Image
- Записва се като свойство image на графичен обект

Свойства на текстурата

- Мащаб scale
- Мащаб на основата baseScale
- Отместване offset
- Адрес на картинката url

ИКТ в НОС

Край

Коментари, въпроси