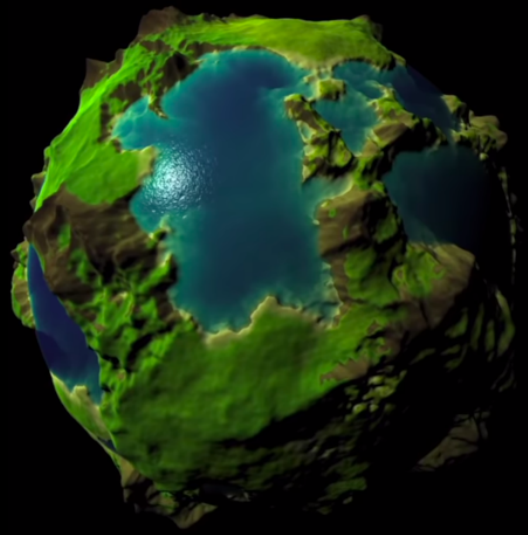
Vizualizarea volumetrică a atmosferei

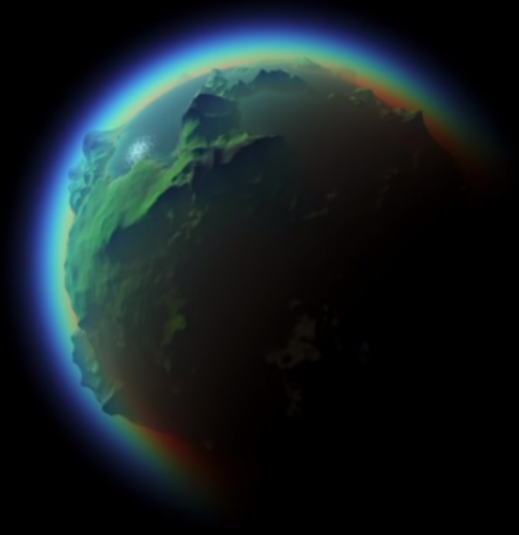
Cuprins



Setarea post-procesării

Generarea atmosferei

Calculul transmiterii luminii



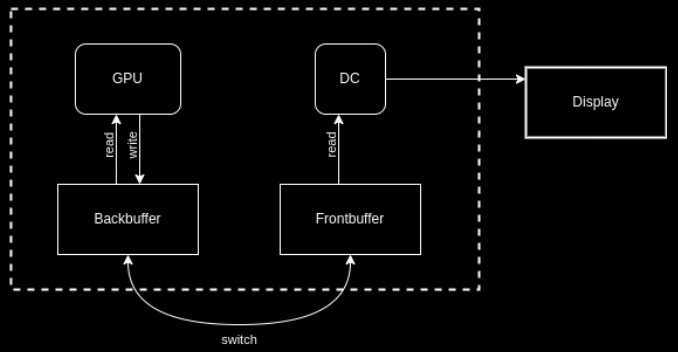
Lungimi de undă: culori

Alte corpuri cerești: s **t** e **l** **e**

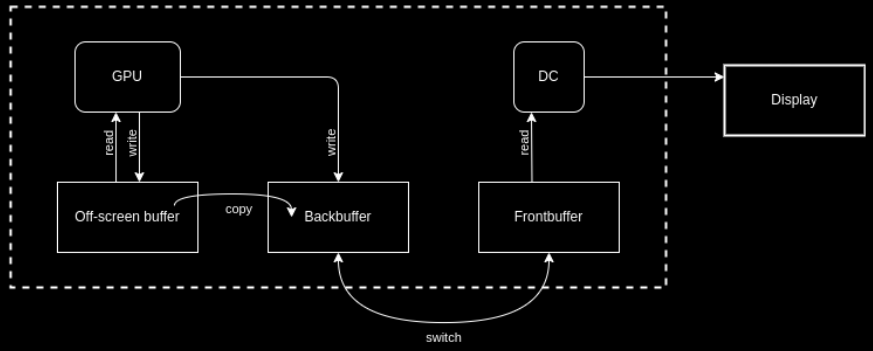
Optimizări

Framebuffers

- double-buffering
  - schimbările se fac pe un „backbuffer”
  - la final, se înlocuiește cu „frontbuffer”-ul
  - permite împrospătarea imaginii înainte de a apărea pe ecran

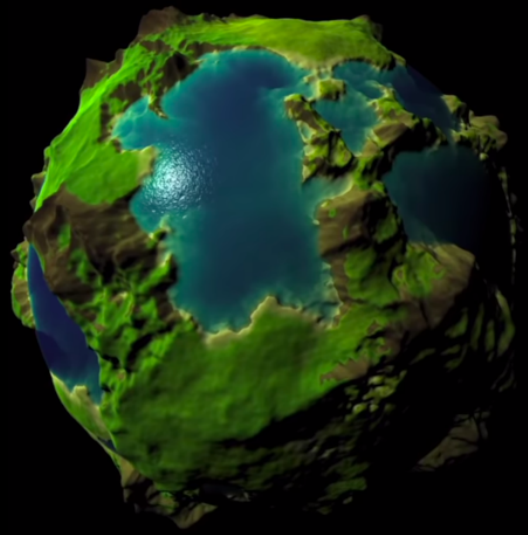


- double-buffering și off-screen buffer
  - schimbările se fac pe un nou buffer ce are ca output o textură
  - textura este aplicată peste un quad pe tot ecranul
  - permite post-procesarea texturii înainte de a fi scrisă în „backbuffer” [1]



- setarea este completă atunci când putem obține același rezultat al scenei, folosind textura generată

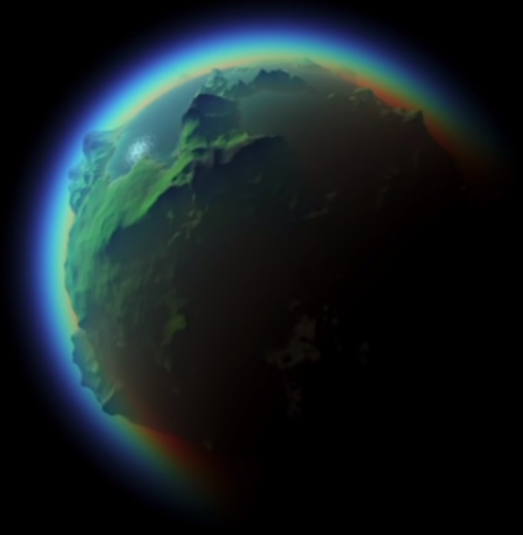
- având control asupra tuturor pixelilor, putem să
  - inversăm culorile
  - transformăm imaginea în grayscale
  - folosim diverse kernele pentru filtrare (ex. blur)
  - calculăm densitatea atmosferei din perspectiva poziției curente a camerei



Setarea post-procesării

Generarea atmosferei

Calculul transmiterii luminii



Lungimi de undă: culori

Alte corpuri cerești: s **t** e **l** **e**

Optimizări

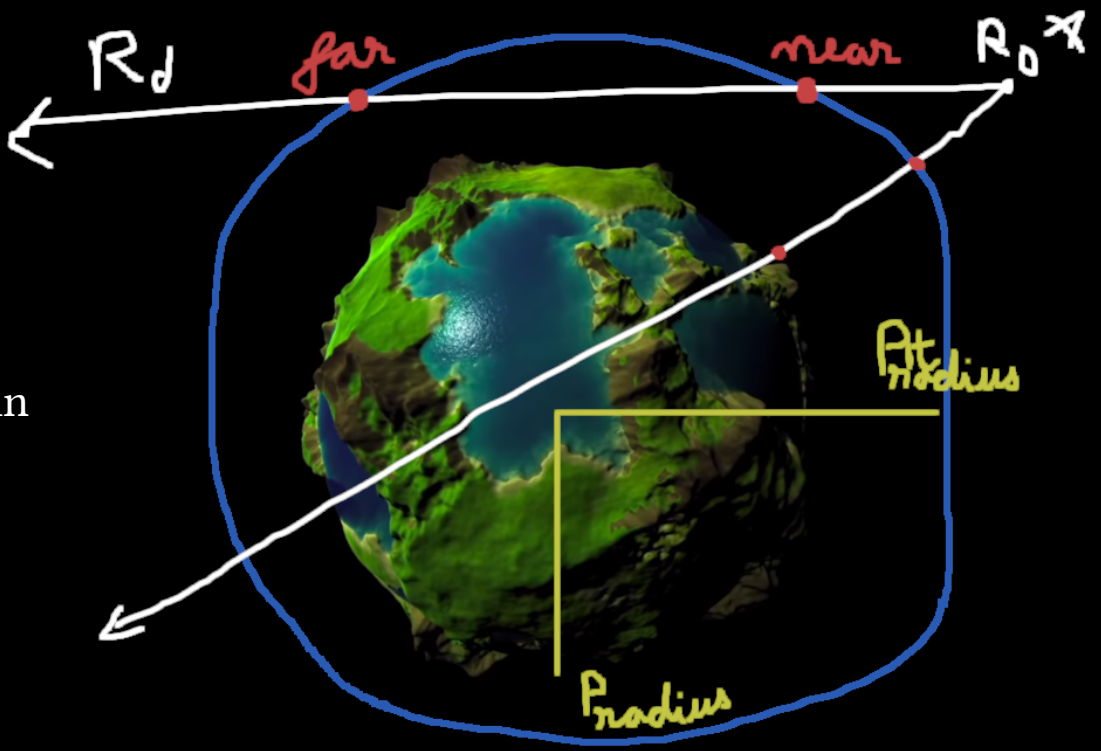
Raycasting

- se calculează punctele de intersecție ale razei $R_o + R_d * t$ cu atmosfera rezolvând ecuația sferei $x^2 + y^2 + z^2 = R^2$ [2]

- densitatea acumulată a atmosferei întoarsă prin pixelul văzut de cameră este calculată ca

$$\frac{\text{far} - \text{near}}{A_{\text{radius}} * 2}$$

adică, proporția distanței parcurse de rază prin atmosferă față de diametrul acesteia



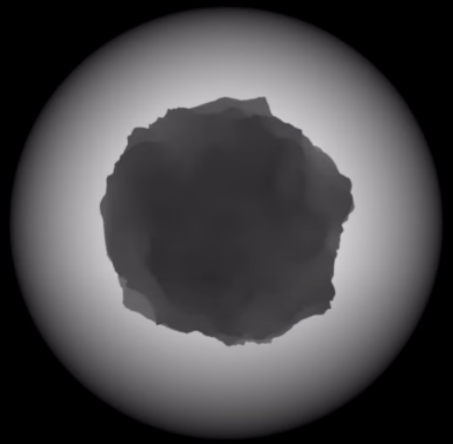
- densitatea acumulată de rază crește cu distanța parcursă prin atmosferă
- atmosferă înconjoară planeta și marimea ei poate fi configurată
  - absolut

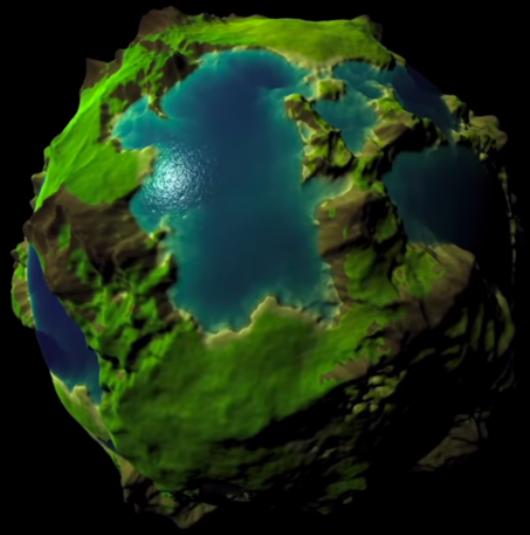
$$A_{\text{radius}} = x$$

  - relativ

$$A_{\text{radius}} = P_{\text{radius}} + \text{offset}$$

- atmosfera pare uniformă deoarece nu sunt modelate
  - aportul soarelui
  - absorpția energiei până ajunge la cameră [3]





Setarea post-procesării

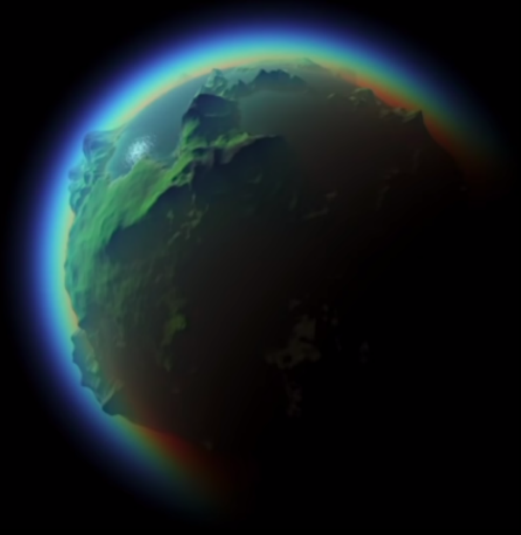
Generarea atmosferei

Calculul transmiterii luminii

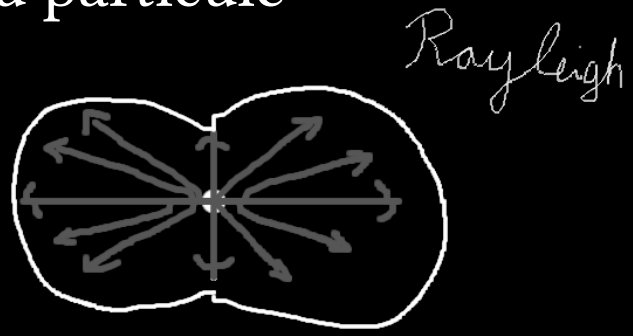
Lungimi de undă: culori

Alte corpuri cerești: s **t** e **l** **e**

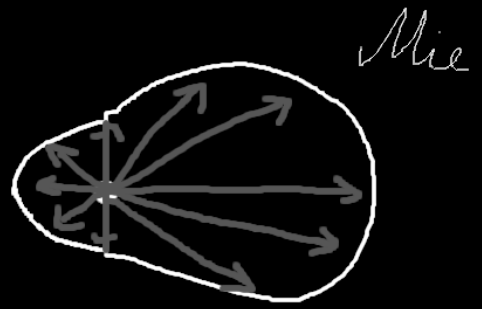
Optimizări



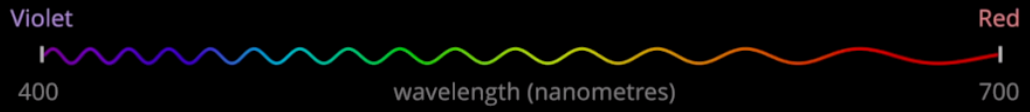
Împrăștierea undelor de lumină la interacțiunea cu particule



- Rayleigh
  - mărimea particulei este mult mai mică ca lungimea de undă
  - razele se împrăștie mai mult spre înainte și înapoi, și mai puțin pe diagonală sau perpendiculară
  - puterea de împrăștiere este invers proporțională cu λ^4
  - ex. atmosfera: lumina violetă este împrăștiată de aproape 10 ori mai mult ca lumina roșie

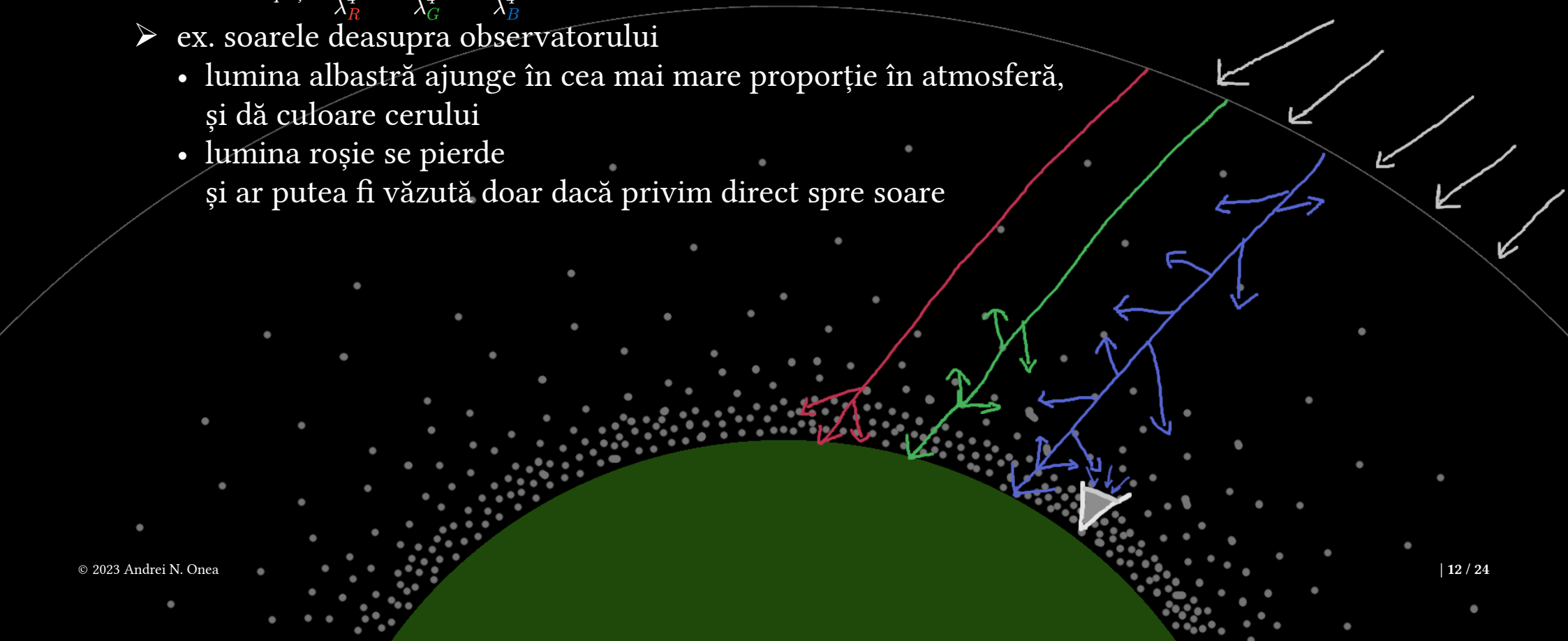


- Mie
  - mărimea particulei este aproximativ la fel ca lungimea de undă
  - razele tind să se împrăștie înainte, unde acumulează mai multă energie
  - puterea de împrăștiere este independentă de lungimea de undă
  - ex. norii: efectul de „silver lining”



- Atmosfera
  - formată din foarte multe particule mici, strâns grupate aproape de suprafață, dar din ce în ce mai răsfirate în depărtare
  - lumina de la soare intră în atmosferă și este împrăștiată în urma interacțiunii cu moleculele

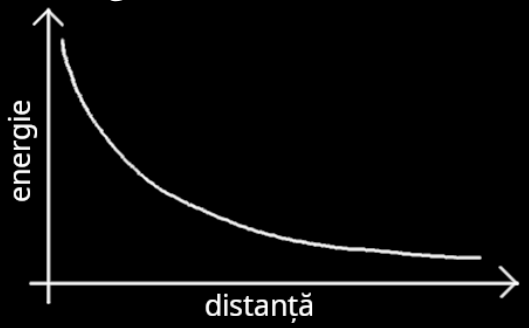
- principalele lungimi de undă detectate de ochiul uman se împrăștie astfel
  - *cel mai puțin* $\frac{1}{\lambda_R^4} < \frac{1}{\lambda_G^4} < \frac{1}{\lambda_B^4}$ *cel mai mult*
- ex. soarele deasupra observatorului
  - lumina albastră ajunge în cea mai mare proporție în atmosferă, și dă culoare cerului
  - lumina roșie se pierde și ar putea fi văzută doar dacă privim direct spre soare

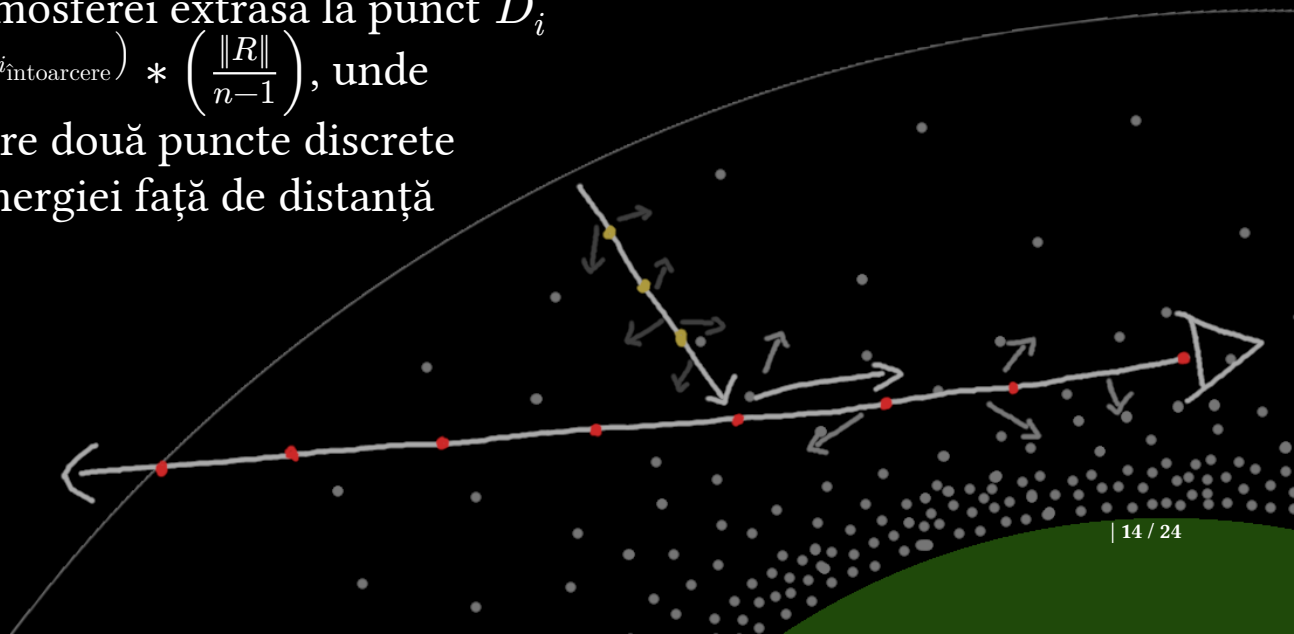


- ex. soarele văzut la orizontul observatorului
  - odată cu distanța crește semnificativ și absorpția de energie
  - lumina albastră este risipită în mare parte până ajunge la ochi
  - lumina verde este și ea risipită
  - predomină lumina roșie, care se va împrăștia mai aproape de observator și ne va dărui un frumos apus sau răsărit

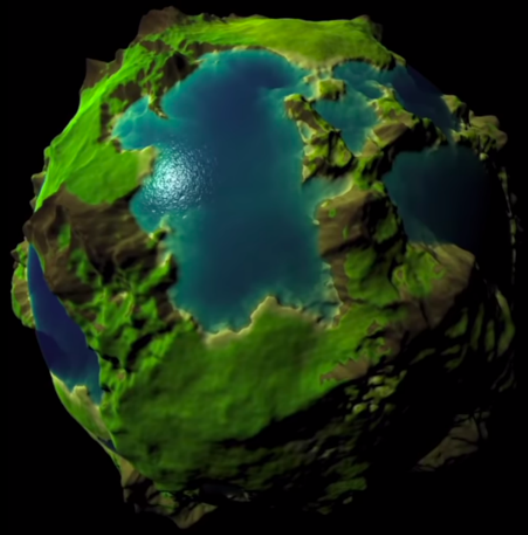
Algoritmul de calcul

- se discretizează parcursul razei R prin atmosferă, apoi pentru fiecare punct i
- se calculează energia primită de la soare de la limita atmosferei până la punct, tot printr-o discretizare a segmentului $E_{i_{\text{soare}}}$
- se calculează energia pe drumul de întoarcere de la punct până la observator $E_{i_{\text{întoarcere}}}$
- se calculează densitatea locală a atmosferei extrasă la punct D_i
- în final, $L = \sum_{i=0}^{n} D_i * e^{-\left(E_{i_{\text{soare}}} + E_{i_{\text{întoarcere}}}\right)} * \left(\frac{\|R\|}{n-1}\right)$, unde
  - ultimul termen este distanța dintre două puncte discrete
  - exponențiala descrie absorpția energiei față de distanță conform legii lui Beer:





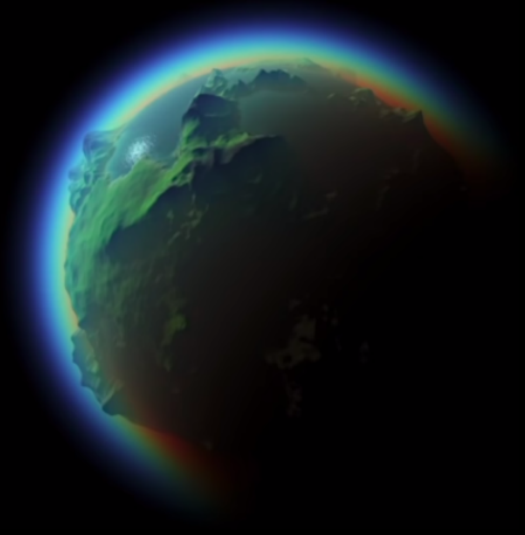
➢ Rezultat



Setarea post-procesării

Generarea atmosferei

Calculul transmiterii luminii



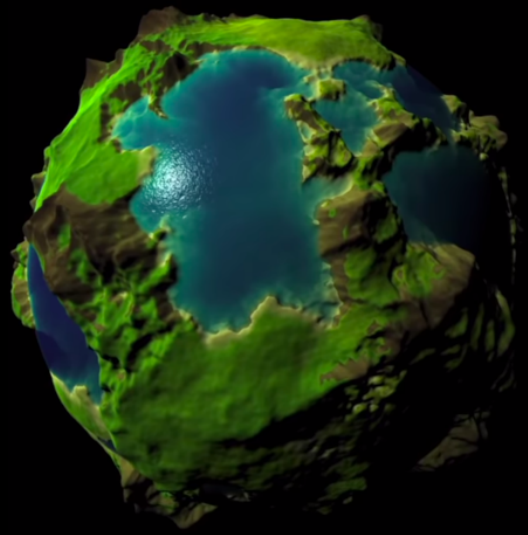
Lungimi de undă: culori

Alte corpuri cerești: s **t** e **l** **e**

Optimizări

Coeficientul de împrăștiere

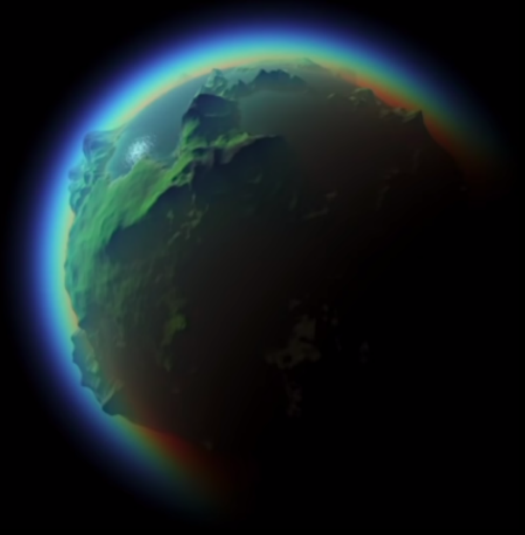
- $C_R = \left(\frac{S_{\text{factor}}}{\lambda_R}\right)^4 * S_{\text{putere}}$
- $C_G = \left(\frac{S_{\text{factor}}}{\lambda_G}\right)^4 * S_{\text{putere}}$
- $C_B = \left(\frac{S_{\text{factor}}}{\lambda_B}\right)^4 * S_{\text{putere}}$
- $S_{\text{factor}} = 1$ în mod normal, dar se recomandă o valoare mai apropiată de lungimile de undă alese (ex. 400)
- S_{putere} amplifică contribuția culorilor
- se înmulțește rezultatul de la algoritmul pentru lumină cu fiecare coeficient în parte



Setarea post-procesării

Generarea atmosferei

Calculul transmiterii luminii



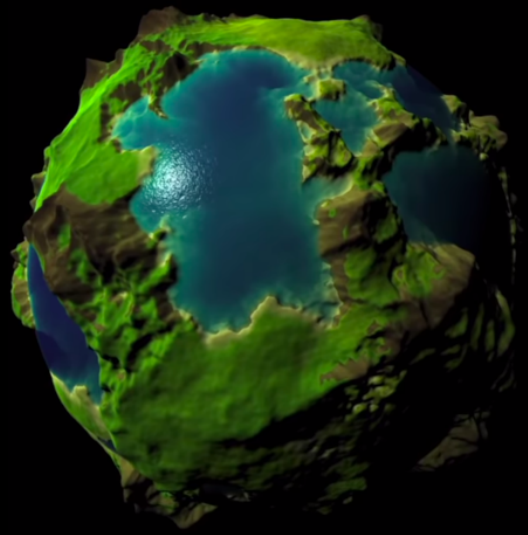
Lungimi de undă: culori

Alte corpuri cerești: s **t** e **l e**

Optimizări

Instanțierea sporită a unui mesh de cerc

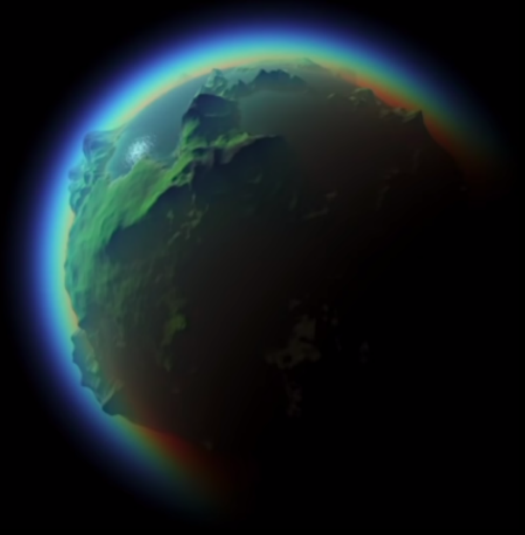
- se creează un număr ridicat de cercuri în jurul planetei
- în shader, se citește luminozitatea pixelului unde s-ar suprapune steaua, și se estompează stele acolo unde este destulă lumină
- în esență, acest lucru permită apariția stelor doar noaptea, când e întuneric, și în plus la hotarul dintre zi și noapte



Setarea post-procesării

Generarea atmosferei

Calculul transmiterii luminii



Lungimi de undă: culori

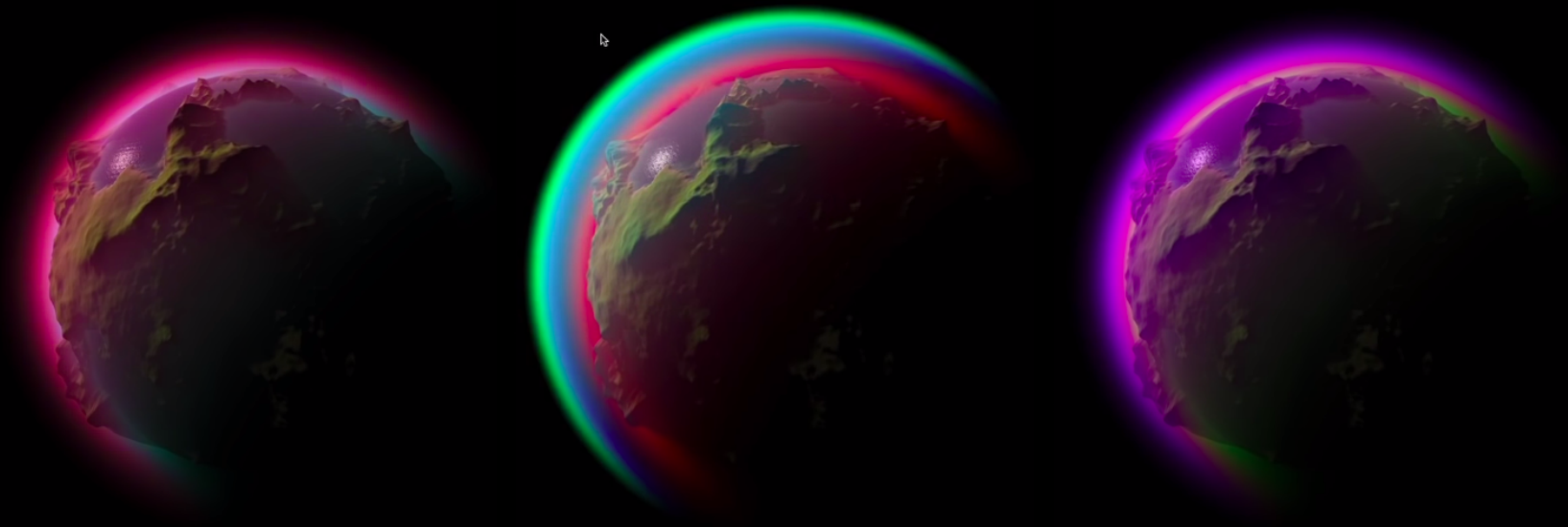
Alte corpuri cerești: s **t** e **l** **e**

Optimizări

Pre-calcularea densităților din atmosferă

- se rulează un număr mare de calcule din toate zonele din atmosferă
- se salvează rezultatul într-o textură 2D
- în algoritm, în loc să se calculeze densitatea pentru fiecare punct, se ia o mostră din textură

Rezultate



Bibliografie

[1] LearnOpenGL, “Framebuffers.” https://learnopengl.com/Advanced-OpenGL/Framebuffers

[2] Scratchapixel, “A minimal ray-tracer: rendering simple shapes (sphere, cube, disk, plane, etc.).” https://www.scratchapixel.com/lessons/3d-basic-rendering/minimal-ray-tracer-rendering-simple-shapes/ray-sphere-intersection.html

[3] S. Lague, “Coding adventure: atmosphere.” https://www.youtube.com/watch?v=DxfEbulyFcY

Mulțumesc pentru atenție!