# Visualization in Astronomy

Astro Tech Talk

MPIA, 12.05.2017

Thomas Müller
Haus der Astronomie (MPIA)
Email: tmueller@mpia.de

# Data in Astronomy

- Images
  - @ different wavelengths, Hubble, Spitzer, VLT,...
    ($360°$ 20-Gigapixel image of New York, http://360gigapixels.com/nyc-skyline-photo-panorama)
- Spectral data
  - Integral field spectroscopy
- Point data sets
  - Hipparcos, Gaia (DR1: TGAS-sources: $2{,}057{,}050$; DR2: > 2 Billion)
- Volume data sets
  - from numerical simulations: Illustris-Dark1 $(1820^3 = 6{,}028{,}568{,}000)$

# How to bring image data to the dome

**Haus der Astronomie**

Model: ZEISS powerdome VELVET
Dome diameter: 12m
tilted by $30^\circ$

5 projectors: $1920 \times 1200$
Nvidia Geforce GTX 560
($11{,}520{,}000$ pixels)

Image of „Haus der Astronomie“
Image licence is not clear

# How to bring image data to the dome

**Haus der Astronomie**

Model: ZEISS powerdome VELVET
Dome diameter: 12m
tilted by 30$^\circ$

5 projectors: $1920 \times 1200$
Nvidia Geforce GTX 560

Domemaster: $3072 \times 3072$ (max 30 fps)

# How to bring image data to the dome

## THOR image (H. Beuter et al.)

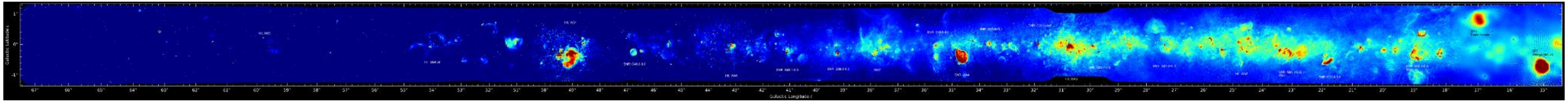
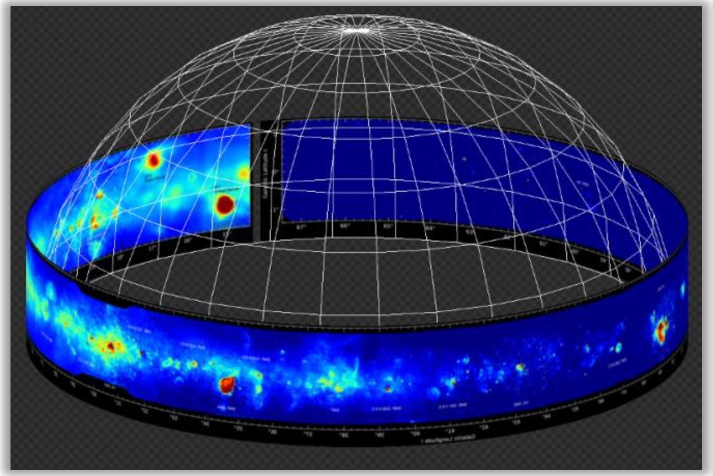
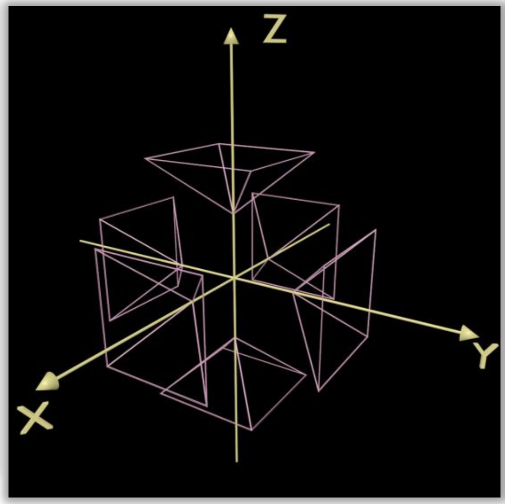
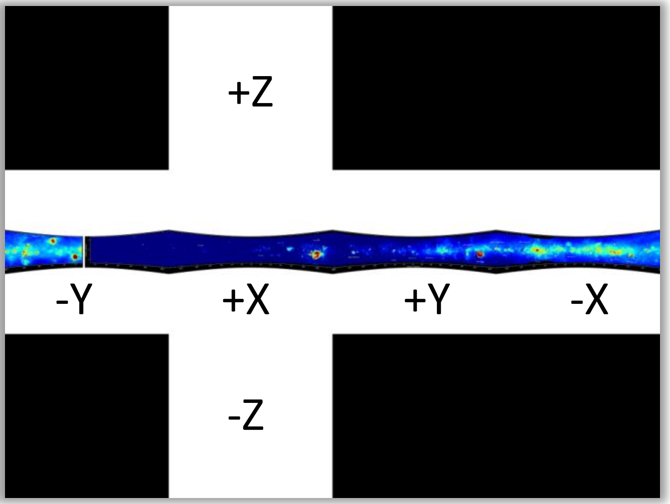


Image resolution 82257 x 4331 (19:1); -1.4 < b < 1.4, 67.4 > l > 14.6

Scale to 360 x 18.9; Cubemap camera: 90° field of view





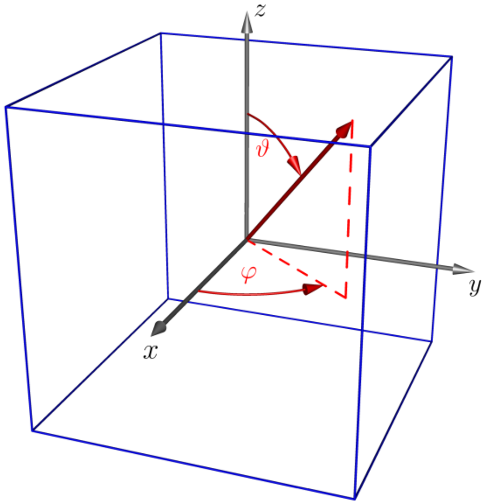



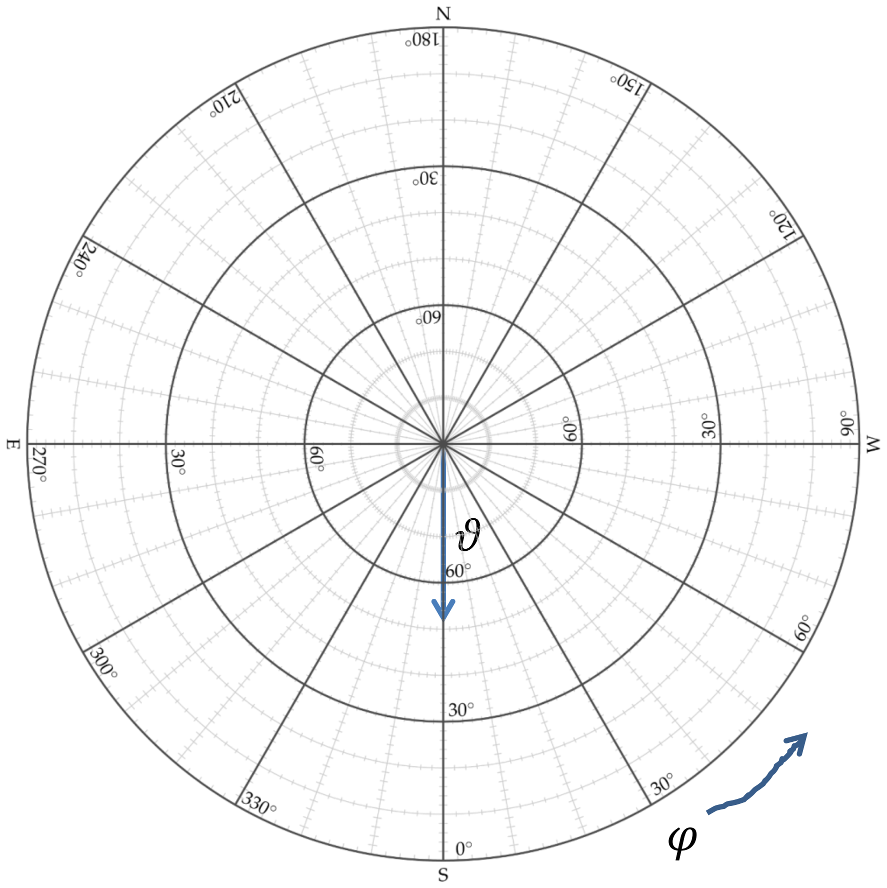
# How to bring image data to the dome

**THOR image (H. Beuter et al.)**

Better: scale to 720 x 37.8

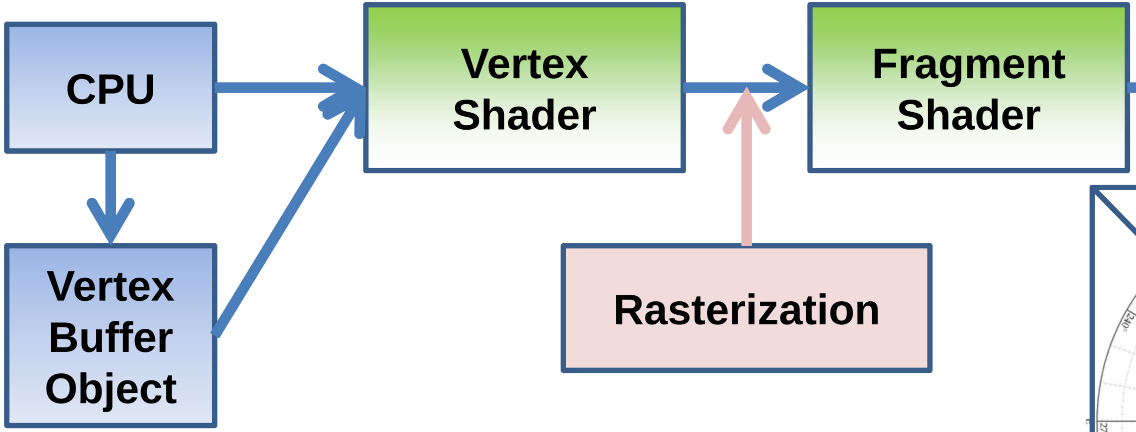
Map Cubemap to DomeMaster

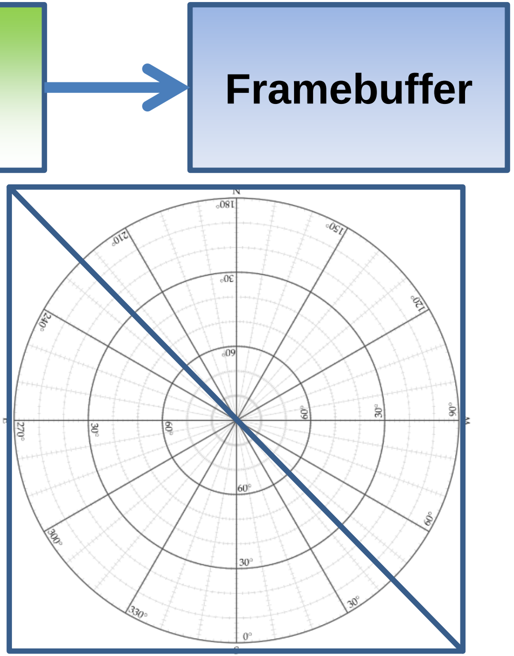


# Insertion: OpenGL graphics pipeline

Interactive rendering using OpenGL vertex and fragment shader



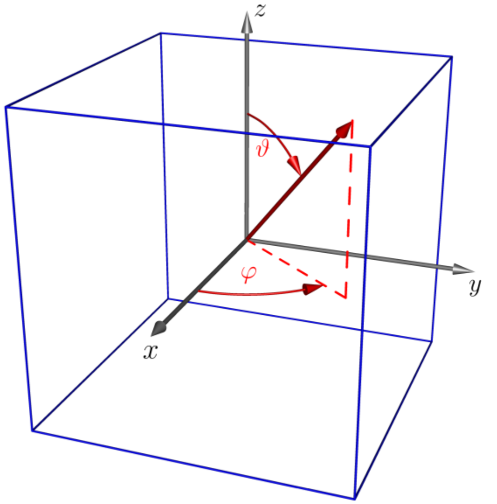

- draw quad using two rectangles
- for each fragment of the quad do…
  - determine $(\vartheta, \varphi) \mapsto \vec{d} = (\sin\vartheta \cos\varphi\,, \sin\vartheta \sin\varphi\,, \cos\vartheta)$

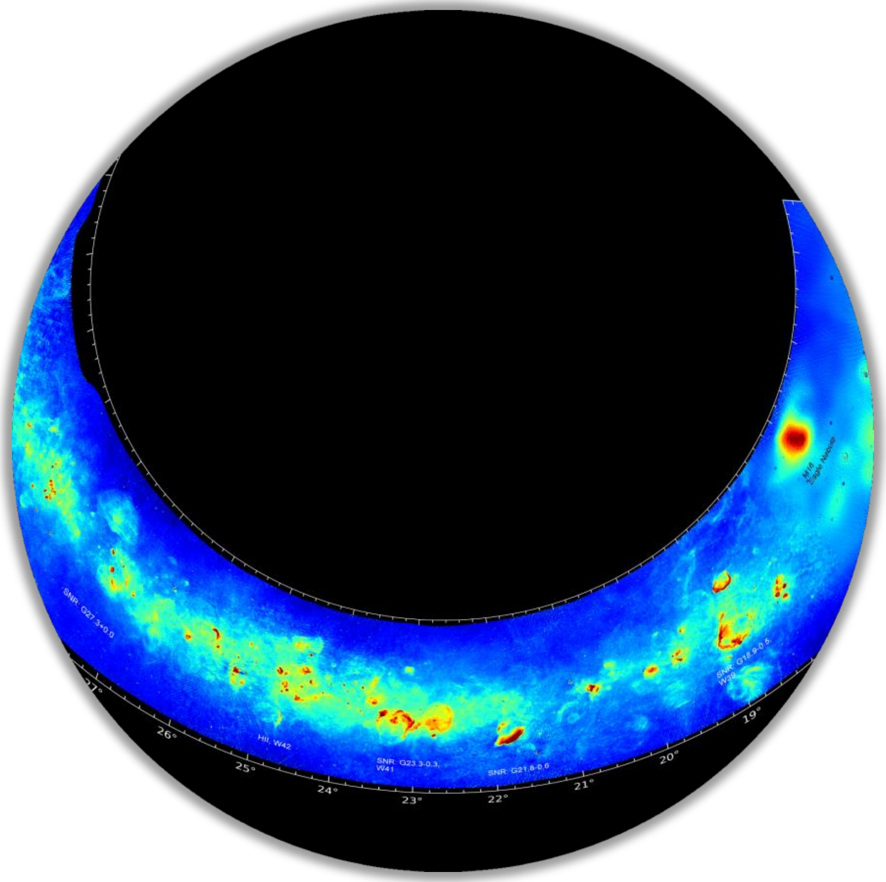
# How to bring image data to the dome

**THOR image (H. Beuter et al.)**

Better:  scale to 720 x 37.8

Map Cubemap to DomeMaster



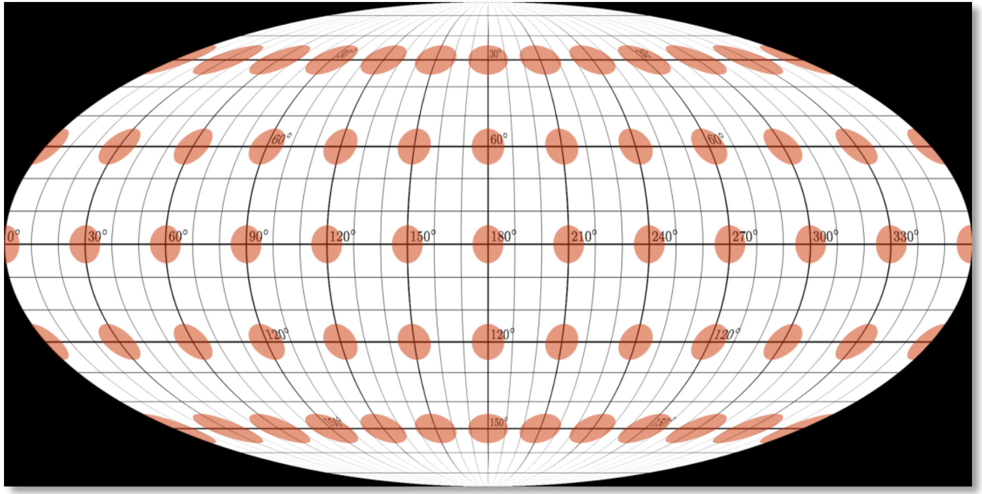



# How to bring image data to the dome

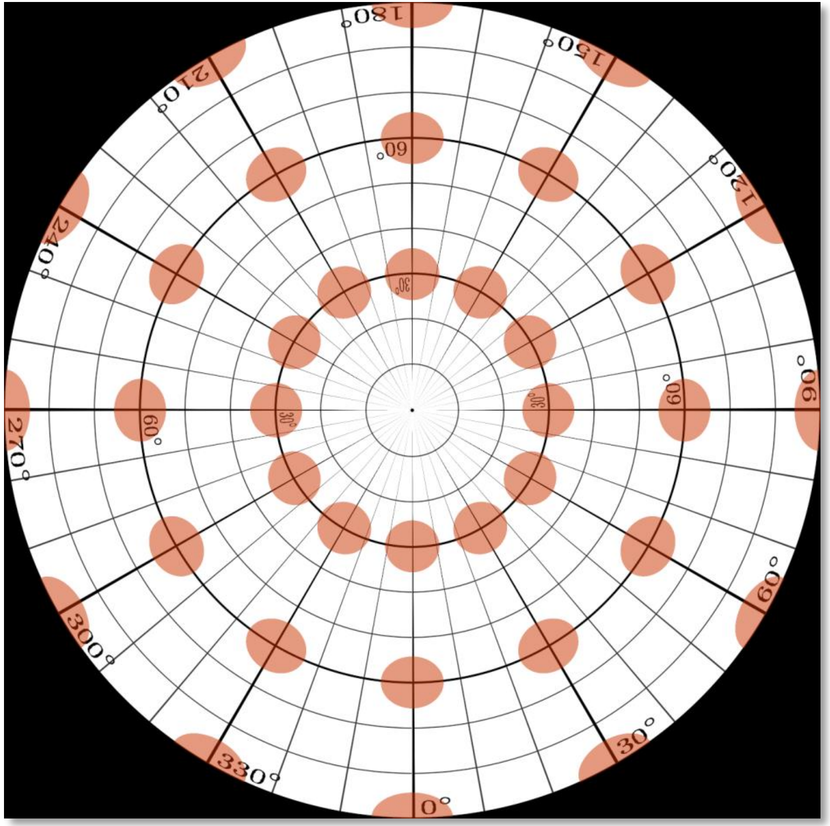
PanSTARRS image covers only $3\pi$

(Panoramic Survey Telescope And Rapid Response System;
1.8m telescope with 1.4 Gpix camera; map sky in 5 broadband filters)

Mollweide to Domemaster




$4\pi$



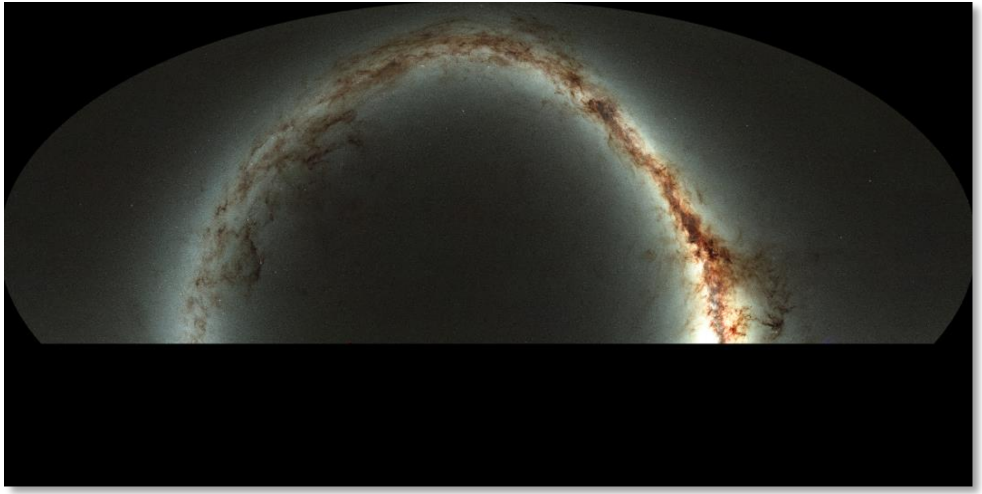

$2\pi$

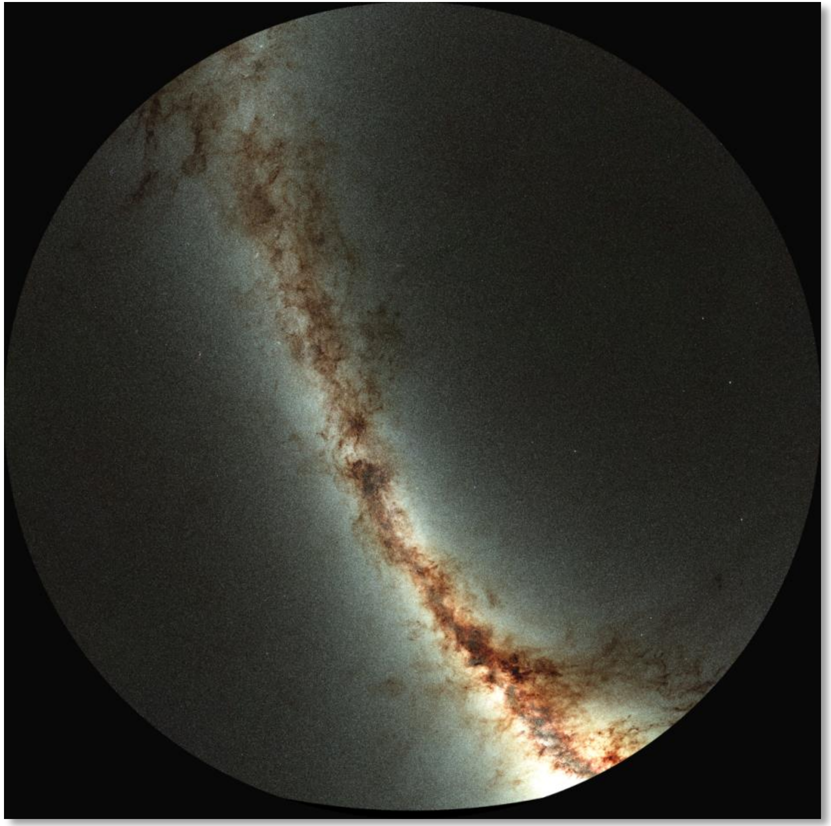
# How to bring image data to the dome

PanSTARRS image covers only $3\pi$

(Panoramic Survey Telescope And Rapid Response System;
1.8m telescope with 1.4 Gpix camera; map sky in 5 broadband filters)

Mollweide to Domemaster





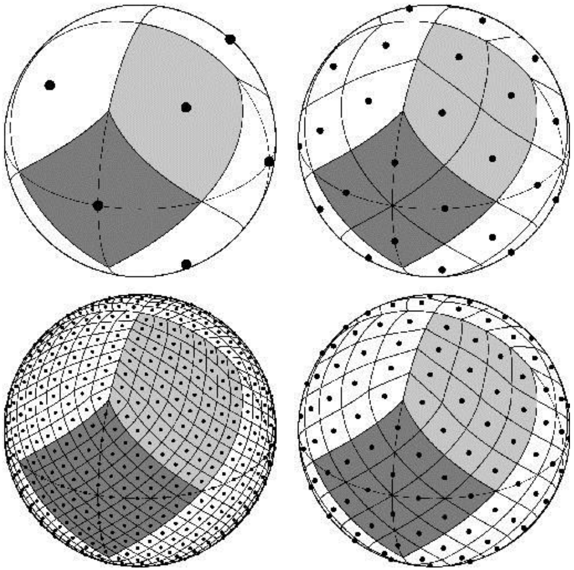
https://panstarrs.stsci.edu/

$2\pi$

# How to bring image data to the dome

## HEALPix to Domemaster

(Hierarchical Equal Area isoLatitude Pixelization)



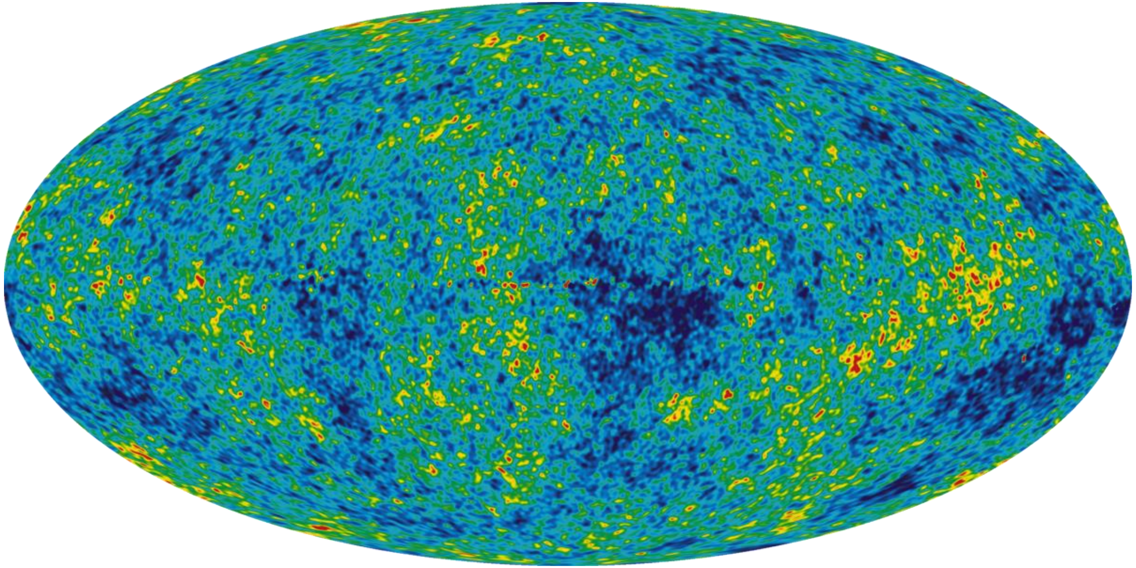
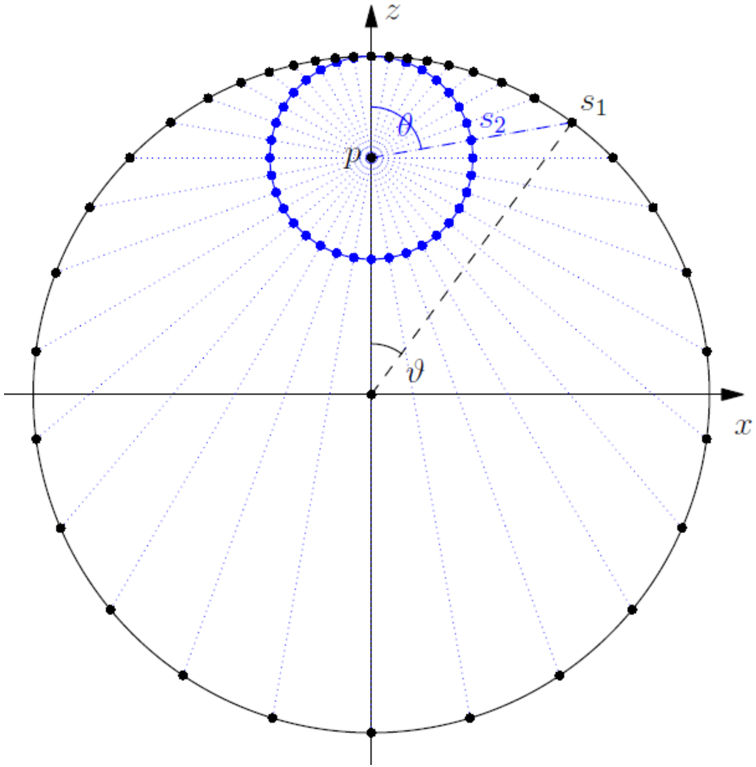


Image source: http://healpix.sourceforge.net/

Image source: NASA / WMAP Science Team

# Zoom into spherical image

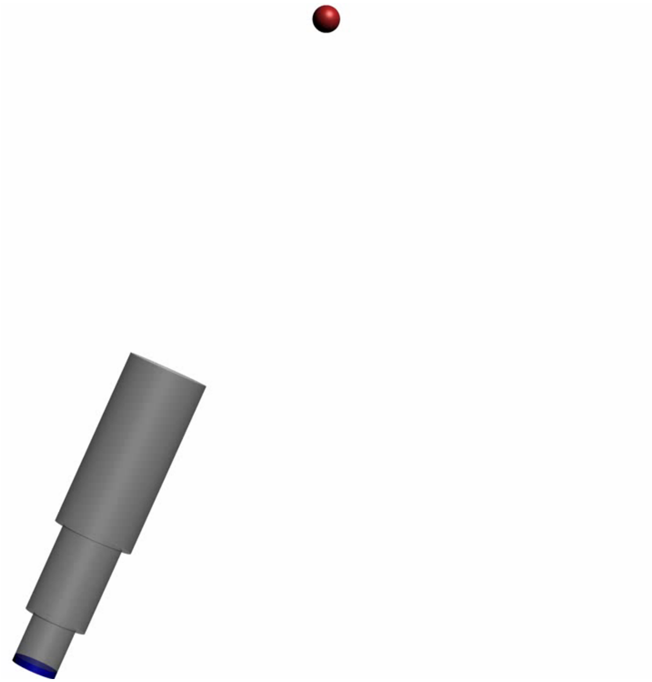
Zoom into image via sphere-to-sphere mapping:



# Zoom into spherical image

Zoom into image via special relativistic aberration

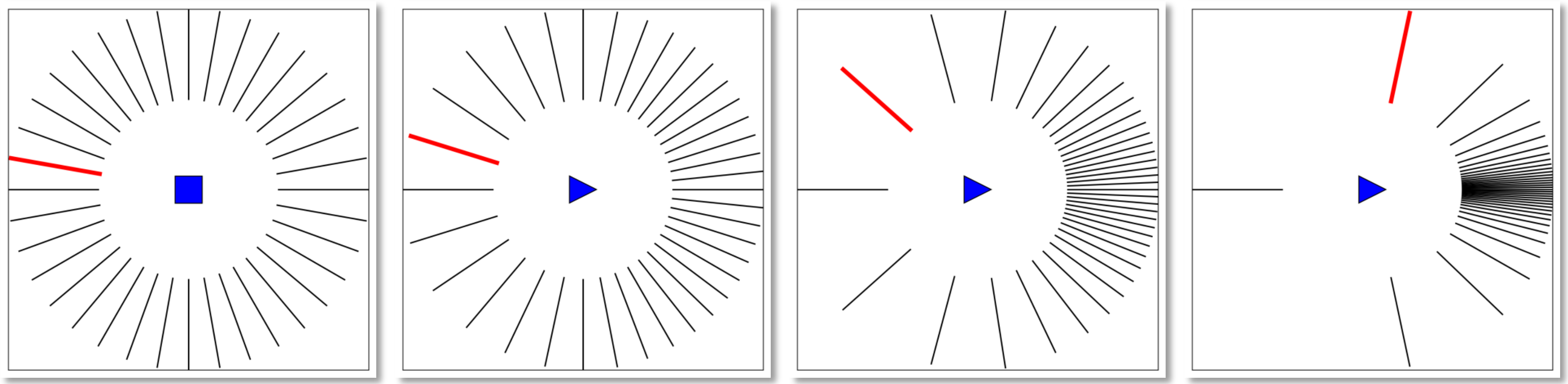
$$\cos\theta = \frac{\cos\vartheta + \beta}{1 + \beta\cos\vartheta}$$



# Zoom into spherical image

Zoom into image via special relativistic aberration

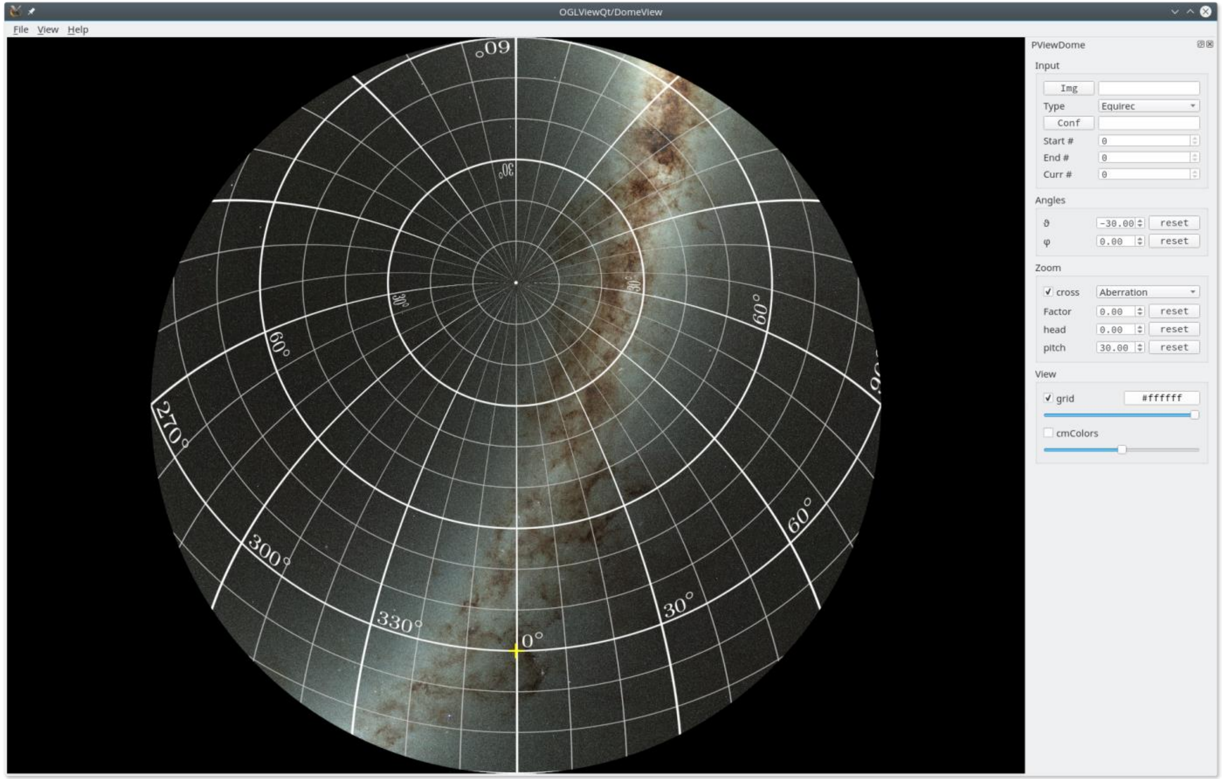
$$\cos\theta = \frac{\cos\vartheta + \beta}{1 + \beta\cos\vartheta}$$



$\beta = 0$ $\beta = 0.5$ $\beta = 0.9$ $\beta = 0.99$

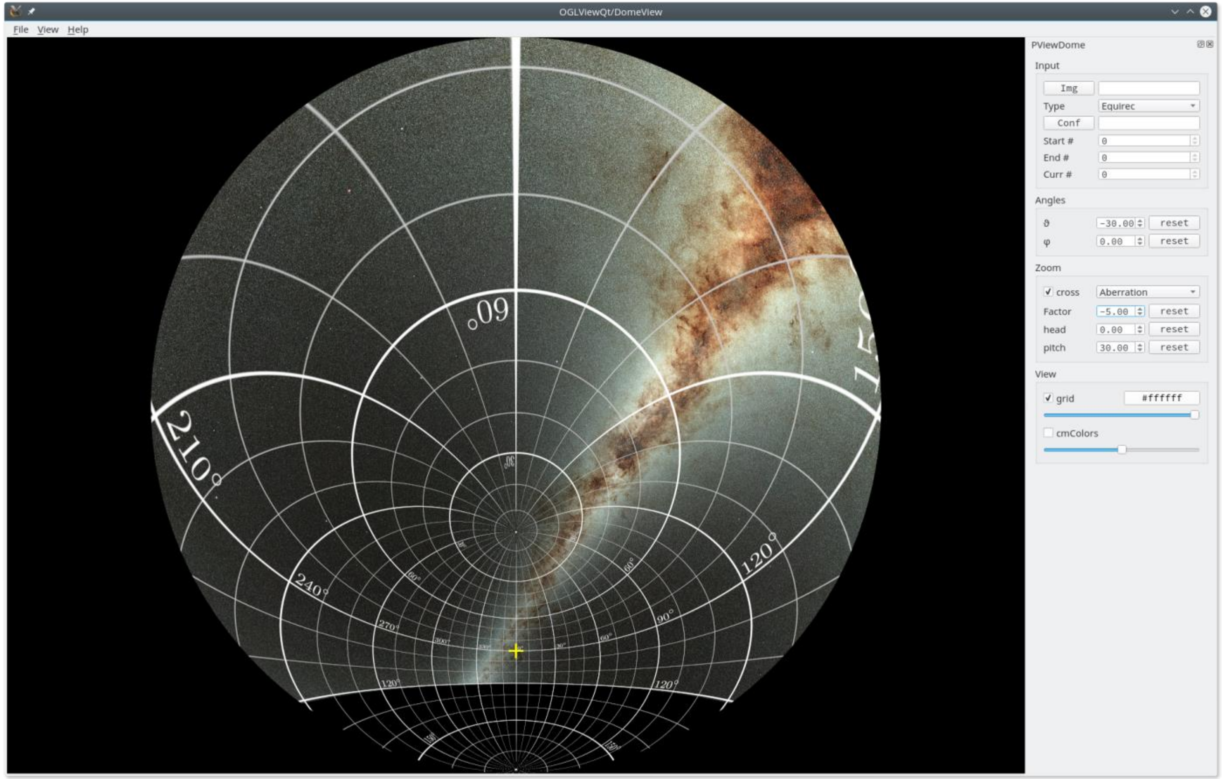
# Zoom into spherical image

Zoom into image via special relativistic aberration



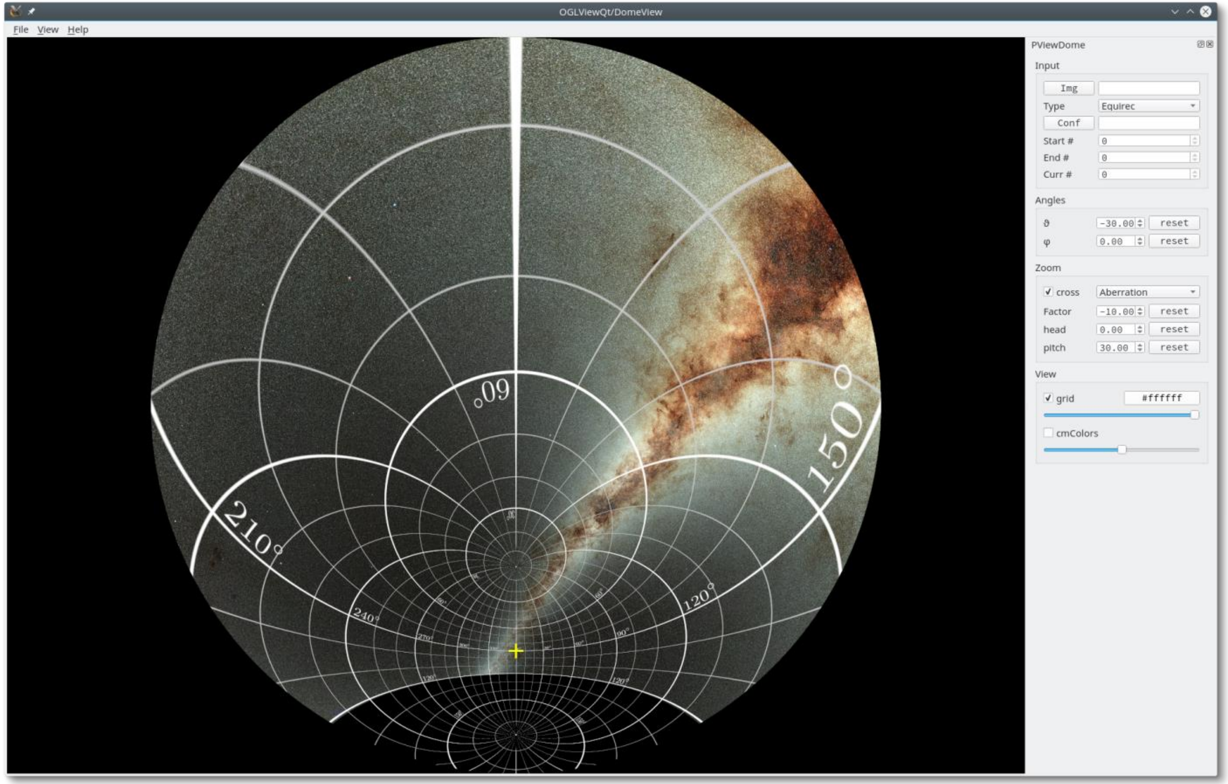
# Zoom into spherical image

Zoom into image via special relativistic aberration



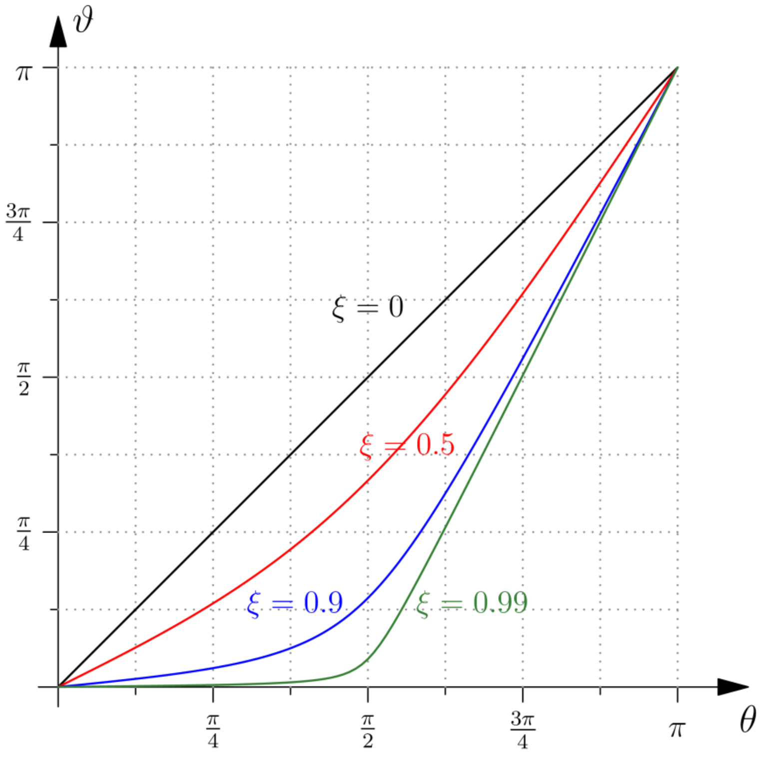
# Zoom into spherical image

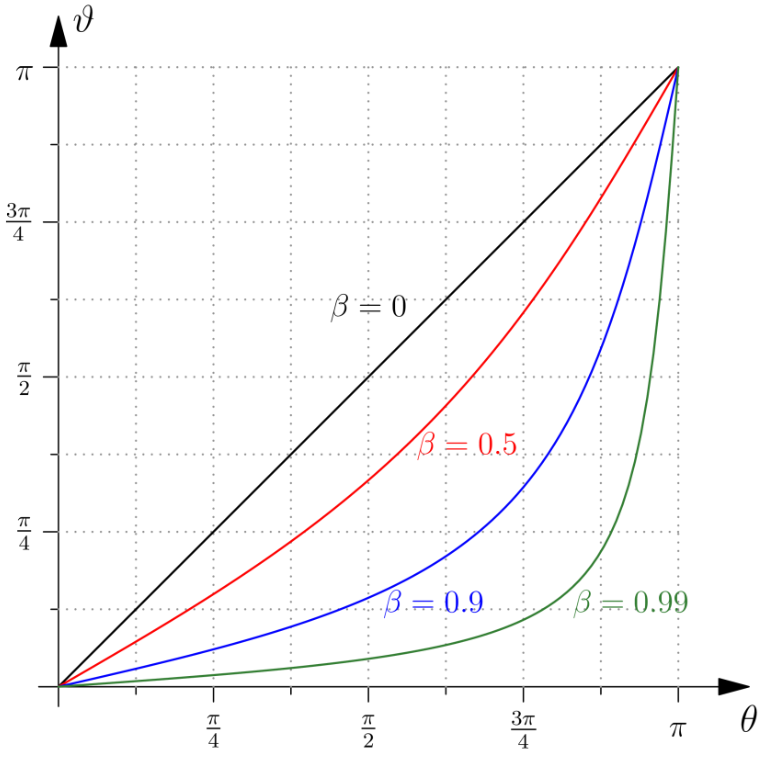
Zoom into image via special relativistic aberration



# Zoom into spherical image

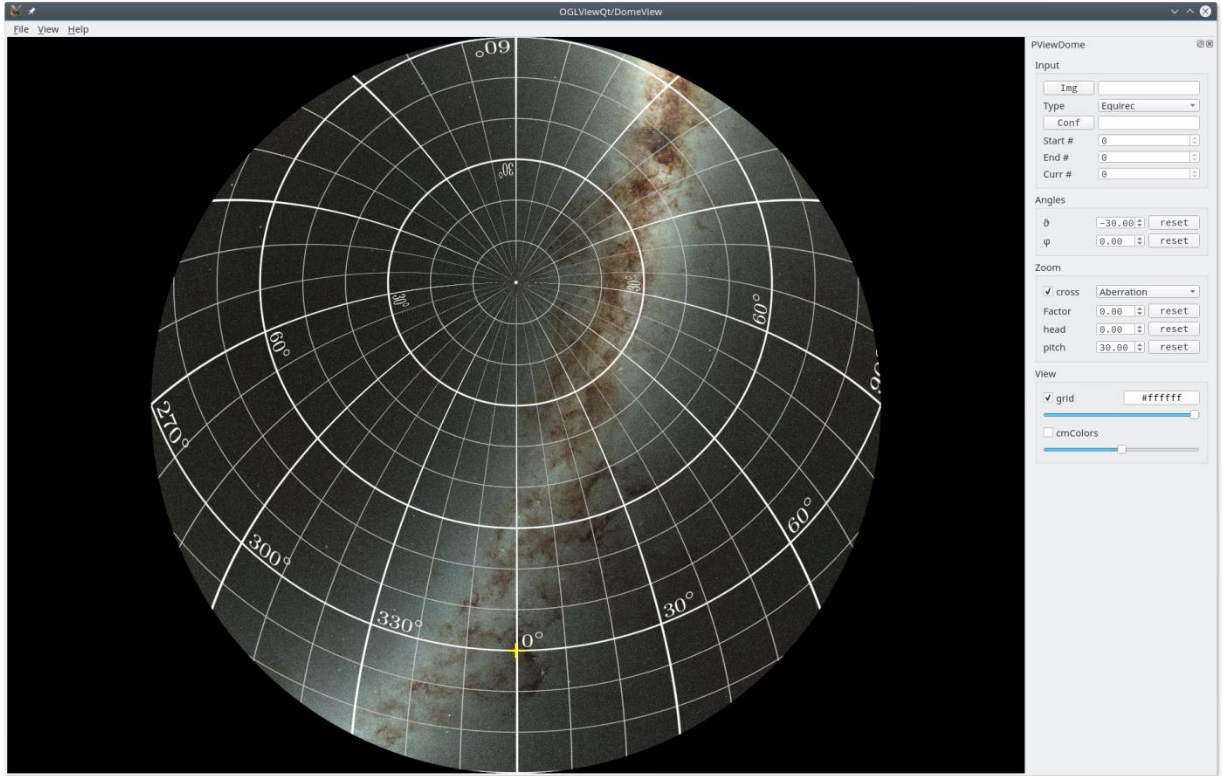
sphere-to-sphere vs special relativistic aberration



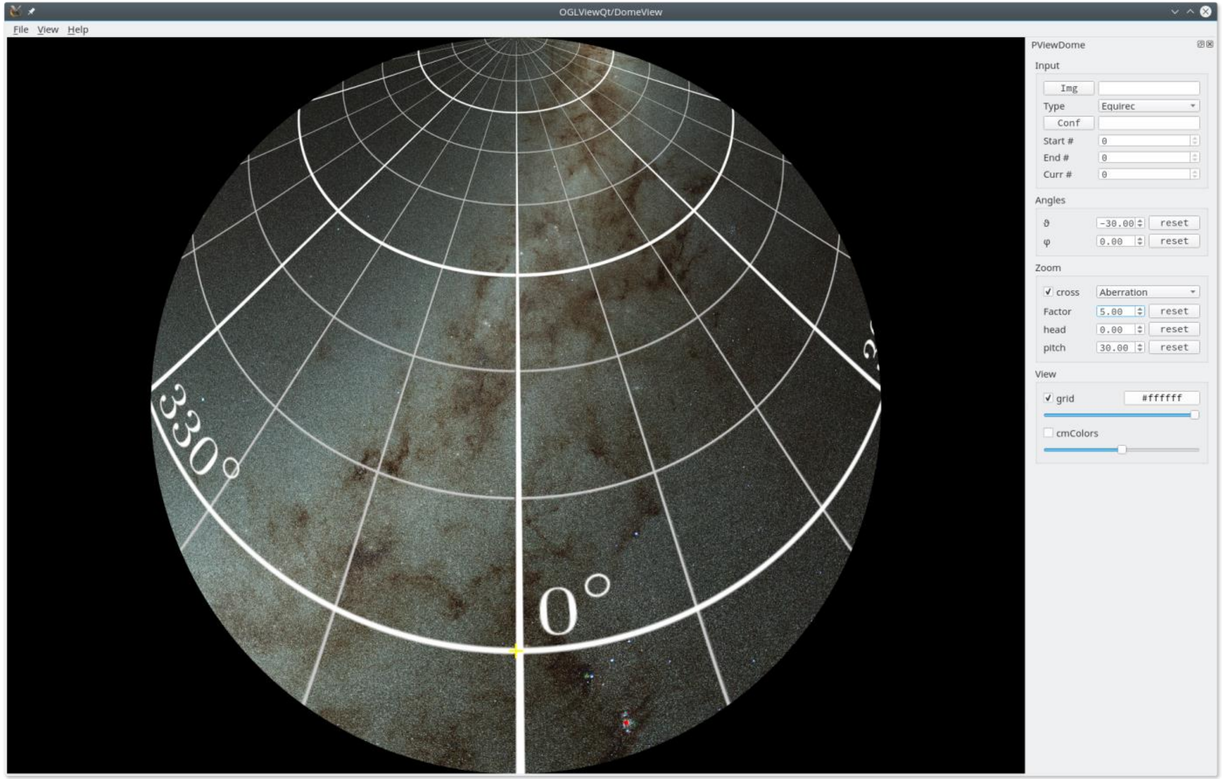
# Zoom into spherical image

Zoom into image via special relativistic aberration



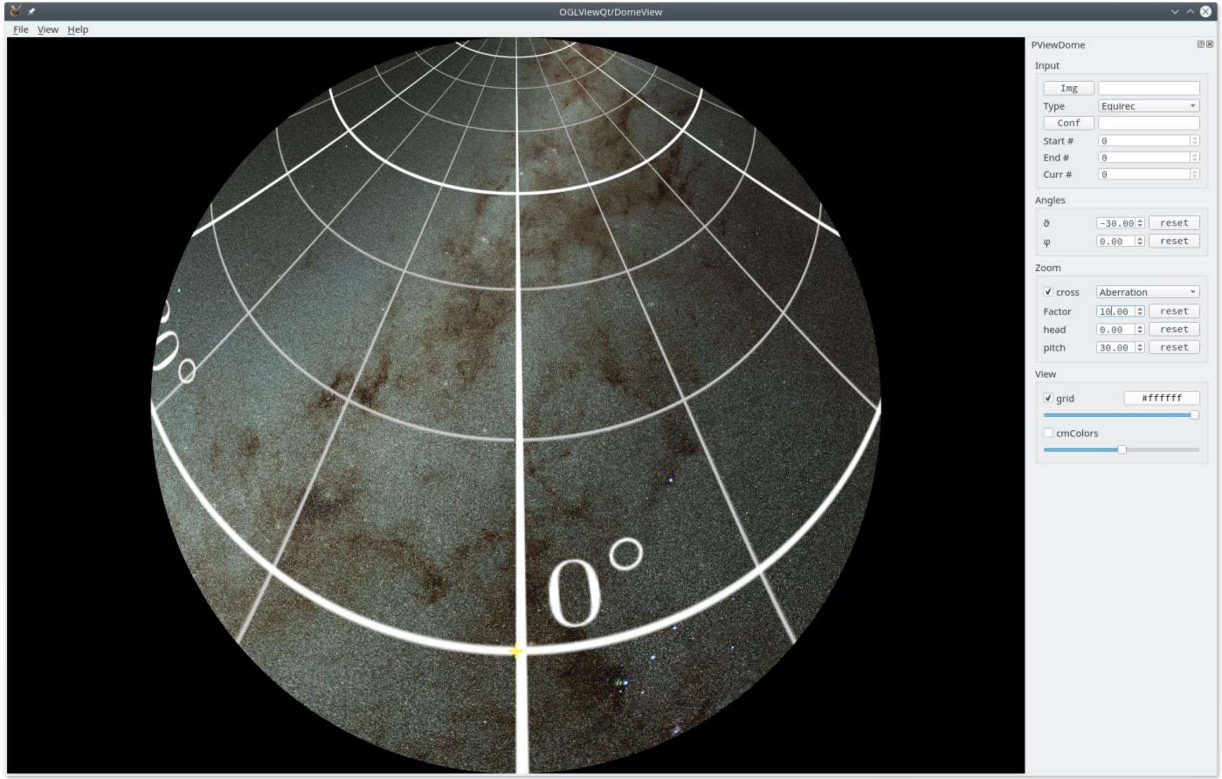
# Zoom into spherical image

Zoom into image via special relativistic aberration



# Zoom into spherical image

Zoom into image via special relativistic aberration



# Very large images

Generate hierarchical image pyramid; caching and preloading

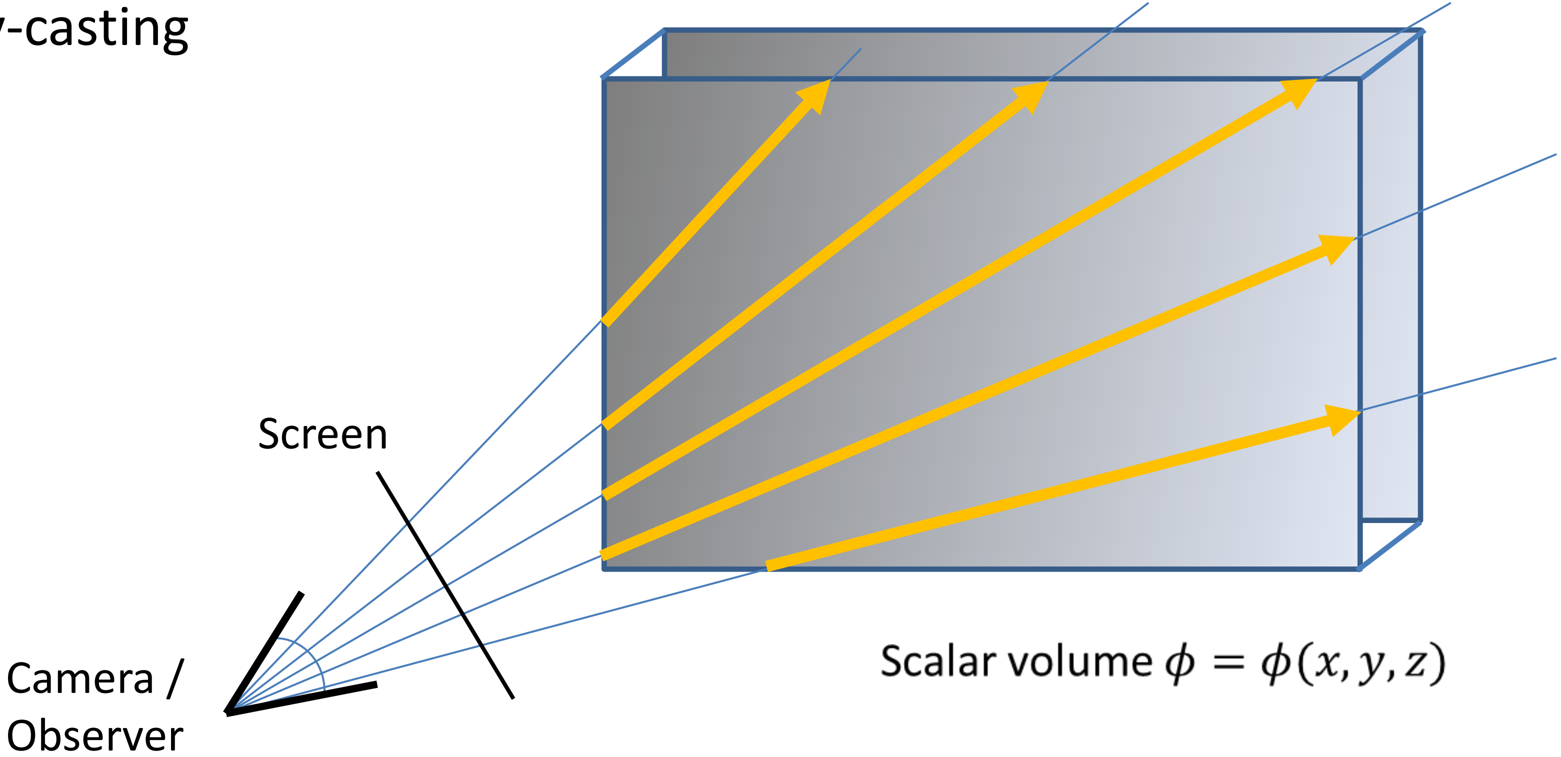
| | |
|---|---|
| LOD 0 | $1024 \times 1024$ |
| LOD 1 | total: $2048 \times 2048$ |
| LOD 2 | total: $4096 \times 4096$ |
| LOD 3 | total: $8192 \times 8192$ |

Image licence is not clear

http://flylib.com/books/2/124/1/html/2/files/figure15-11.jpeg

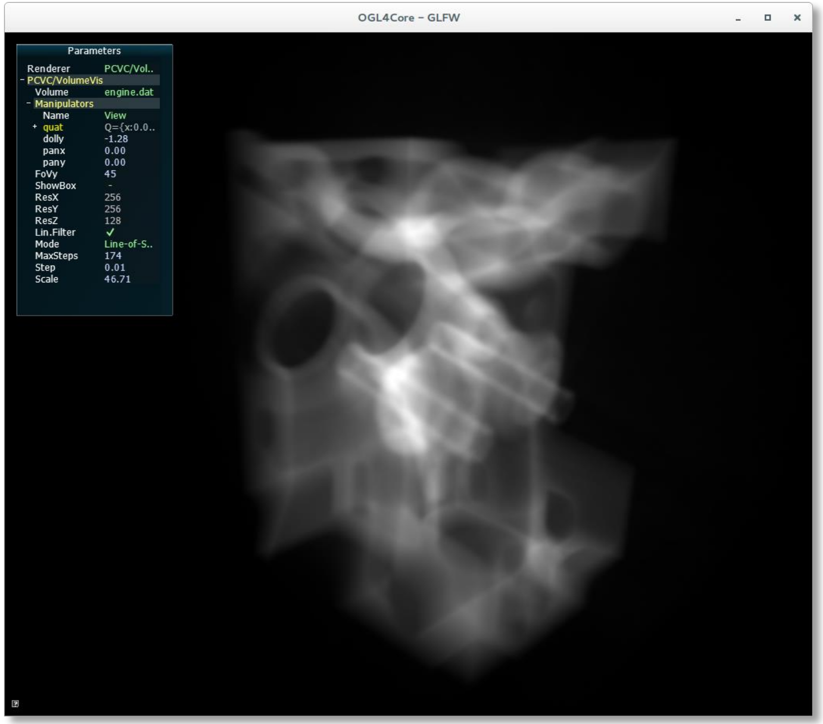
# Direct Volume Rendering

Ray-casting

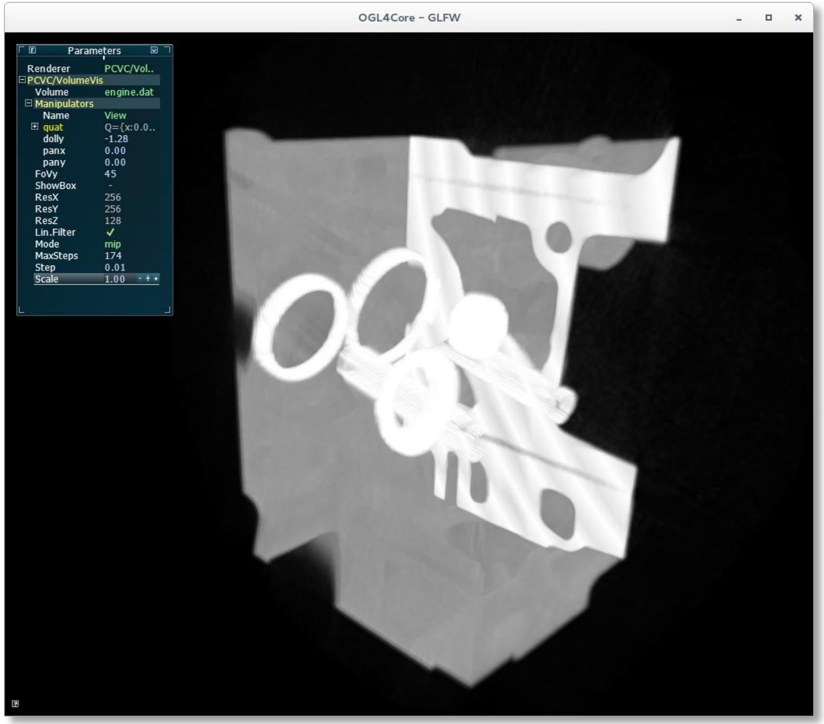


# Direct Volume Rendering

Line-of-sight integration

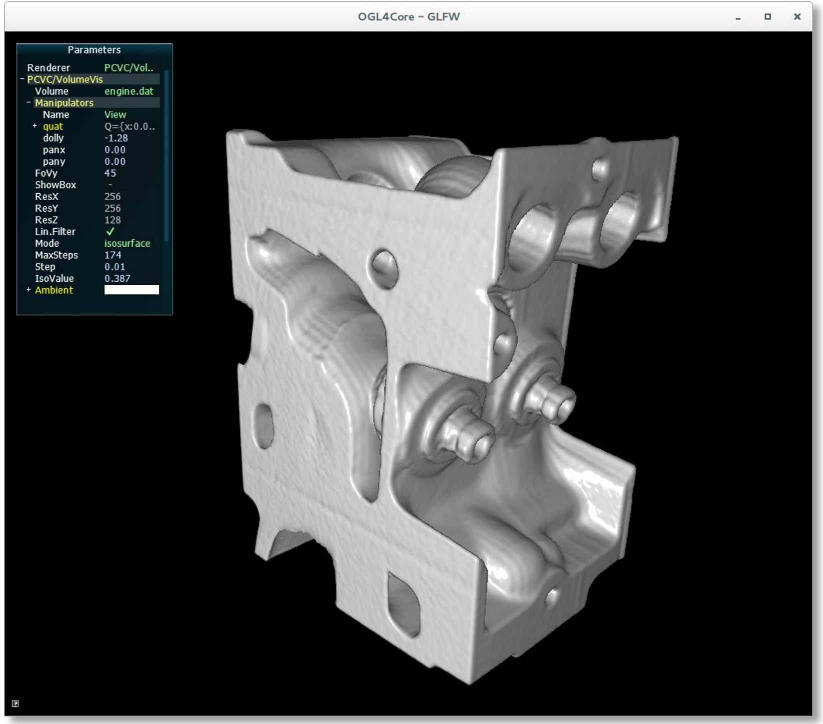



Maximum-Intensity-Projection



# Direct Volume Rendering

## Isosurface with Phong-Illumination



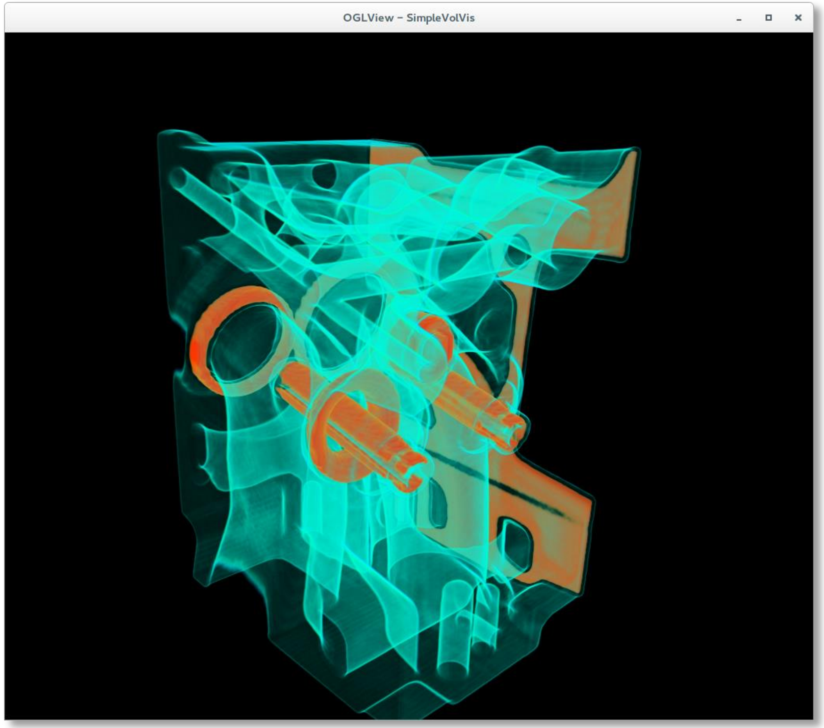

For each ray:

- find scalar value $\phi = \phi_{iso} \Rightarrow P = (x, y, z)$
- calculate gradient $\nabla\phi$ at point $P$
- calculate angle between gradient and direction of light source
- Phong shading based on ambient, diffuse and specular light

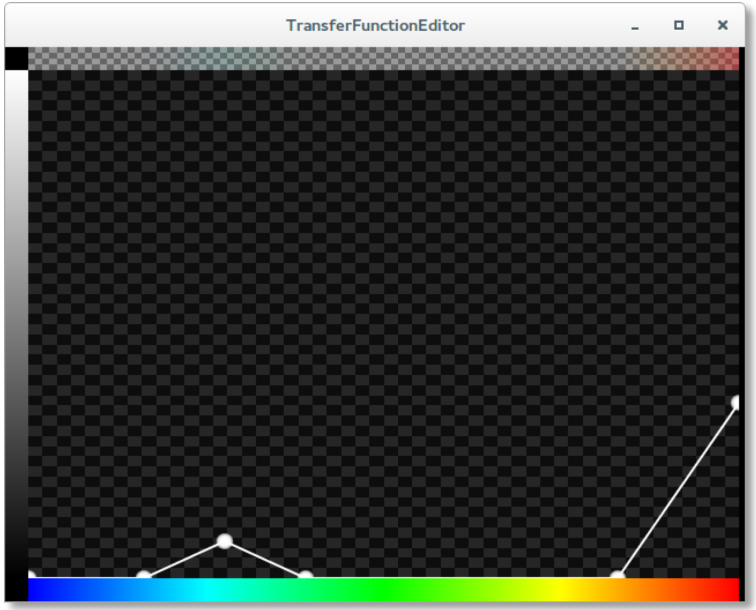
Disadvantage: isosurface is opaque

# Direct Volume Rendering

VolumeVis with Transferfunction




Transferfunction maps scalar value to color and opacity



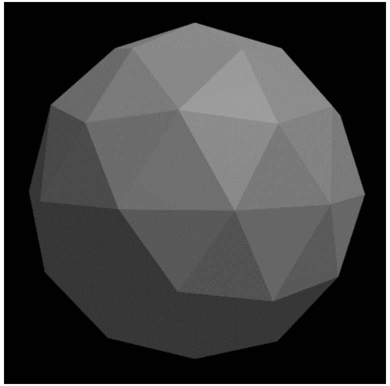
# Direct Volume Rendering

**Challenges:**

- data sets become very large:  do not longer fit into GPU memory
- out-of-core rendering techniques
- dynamic data sets become e.g. too complex to select isosurfaces manually
- visualization of multivariate data: 3D vector or tensor fields
- more advanced transfer functions (2D: scalar + gradient)
- …

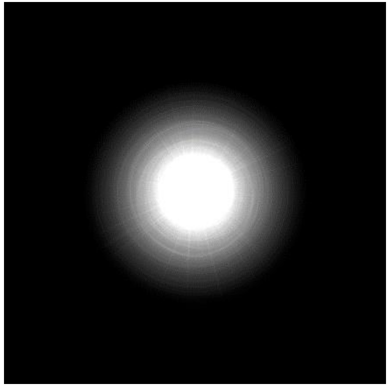
# Particle Rendering

Particle rendering using single vertices.



Particle rendering using small spheres:

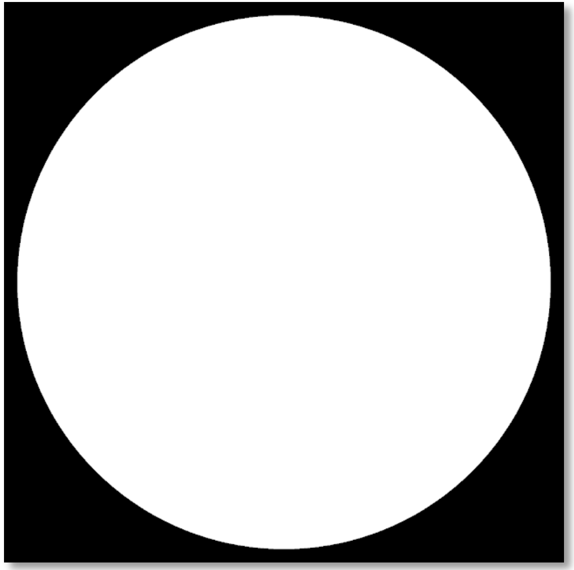
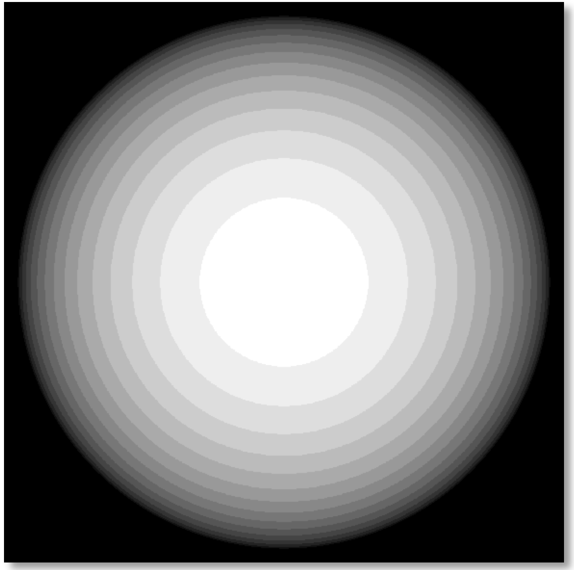
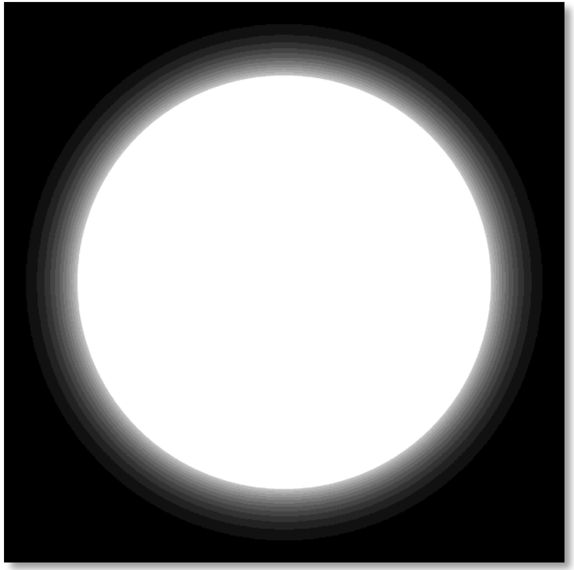
- approximated by triangle mesh;
  too many geometry!
- slow



# Particle Rendering

Particle rendering using billboards (camera oriented, shaded quad)

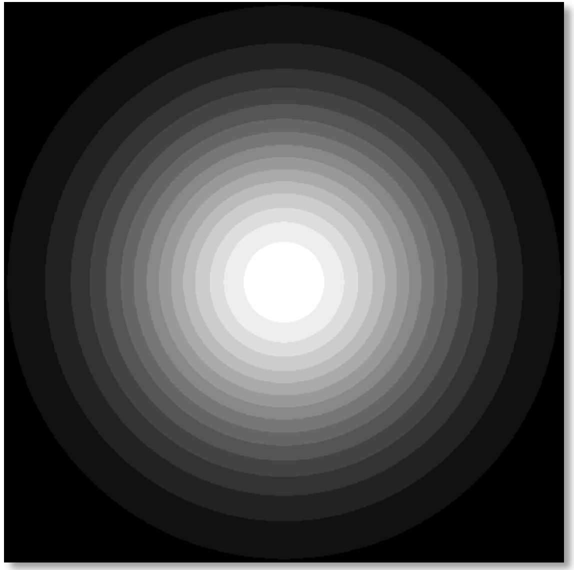
- very fast

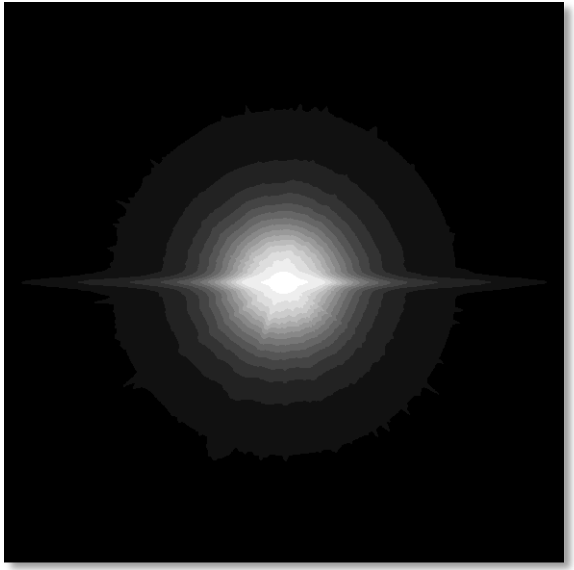




# Particle Rendering

Particle rendering using billboards (camera oriented, shaded quad)

- very fast



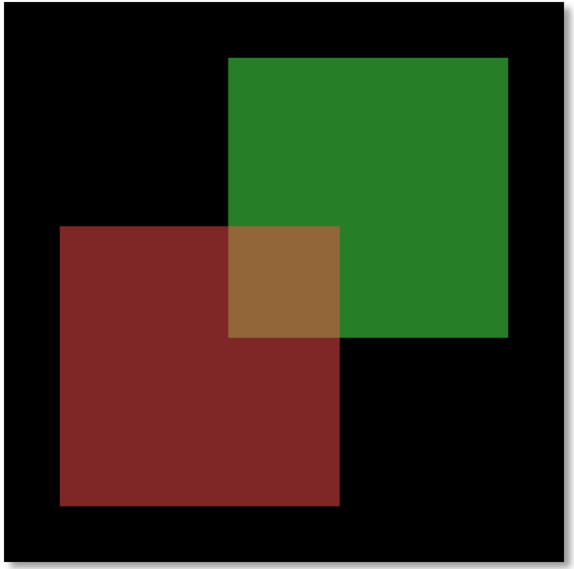


# Particle Rendering

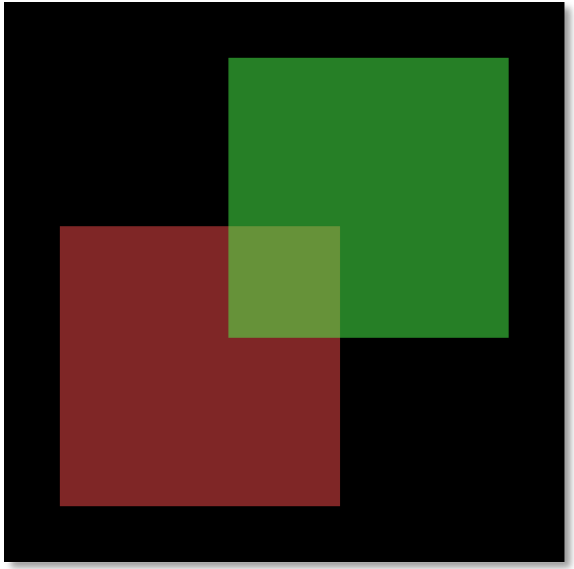
Particle rendering using billboards

- Blending → depth sorting (very expensive)

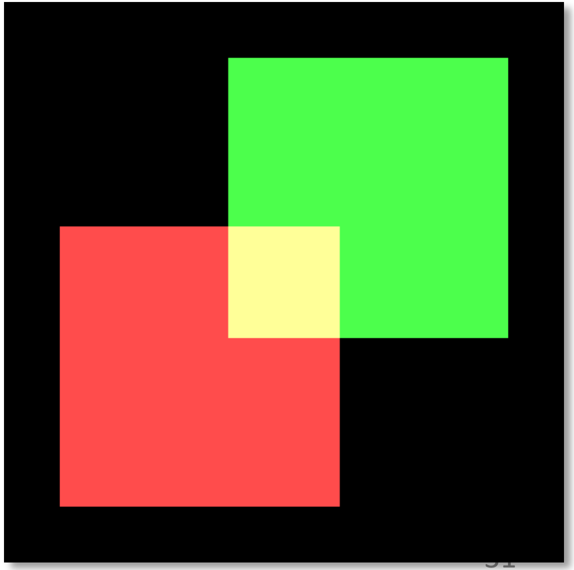
red in front of green



green in front of red

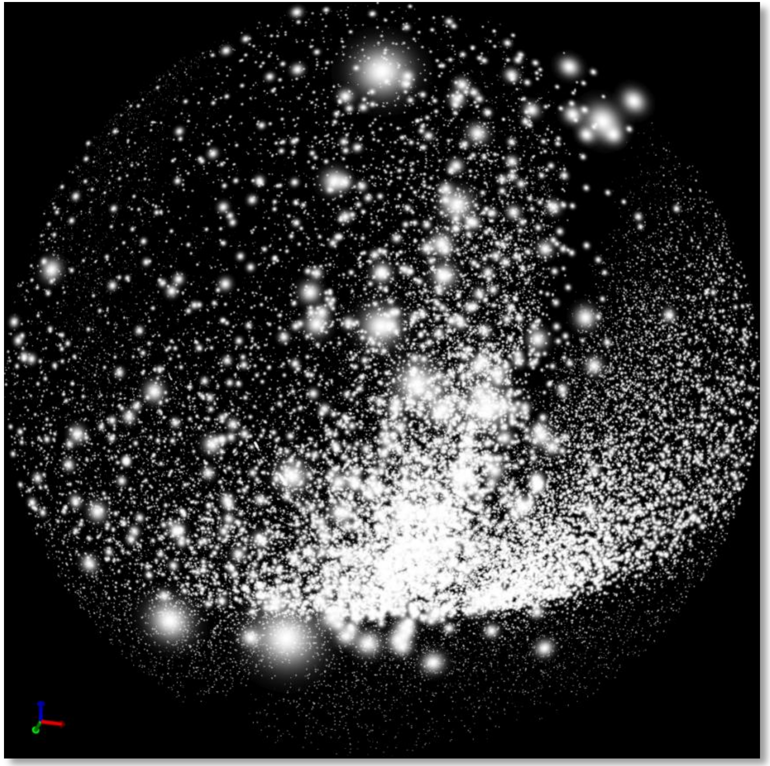


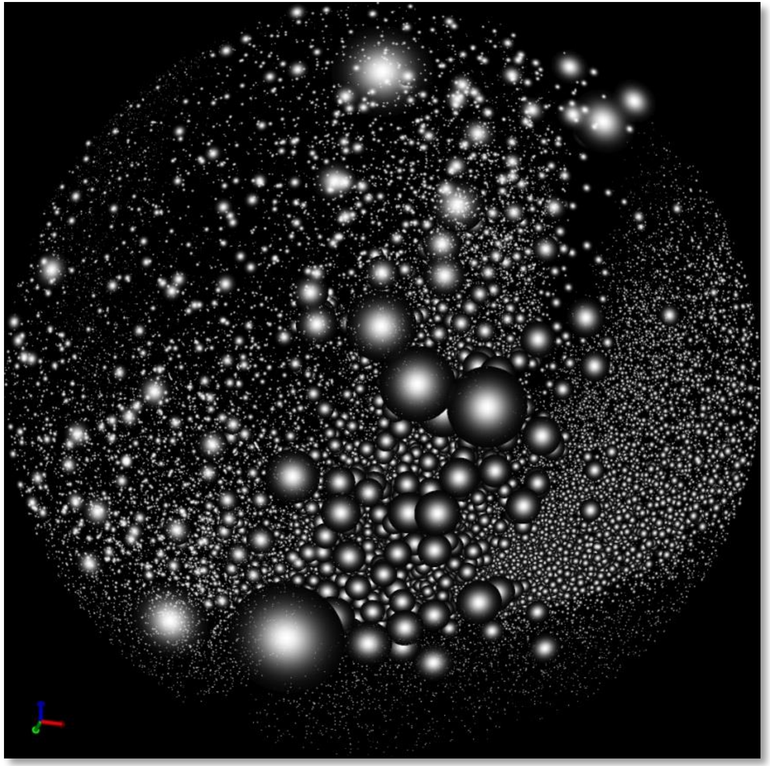
additive blending



# Particle Rendering

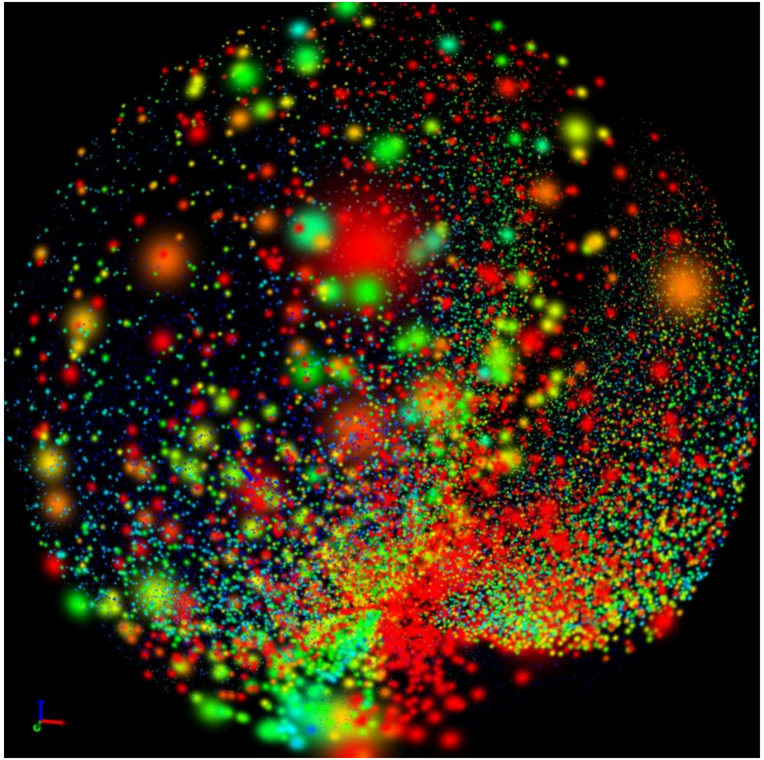
Star rendering using billboards (only white)

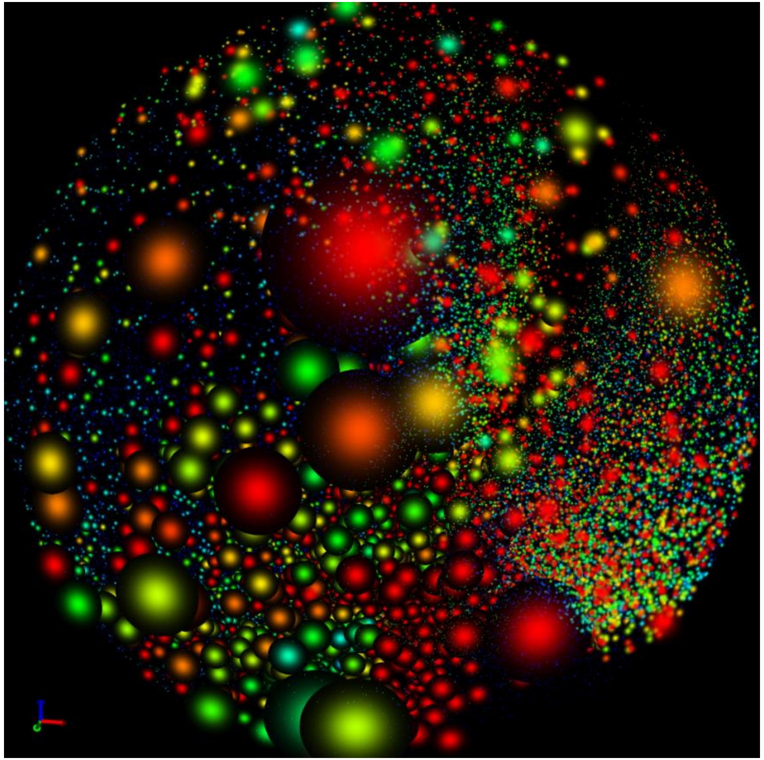




# Particle Rendering

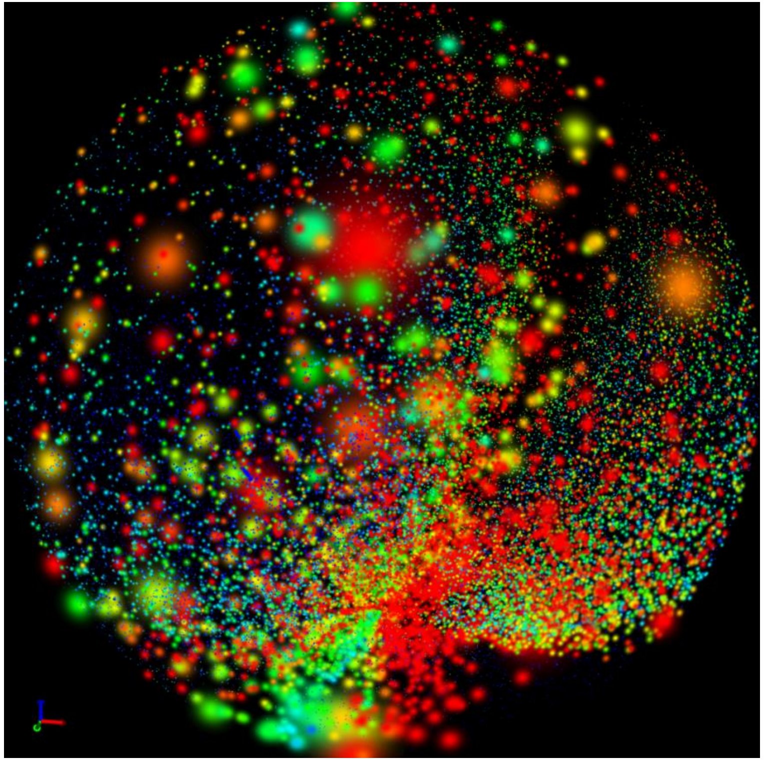
Star rendering using billboards (colored stars)

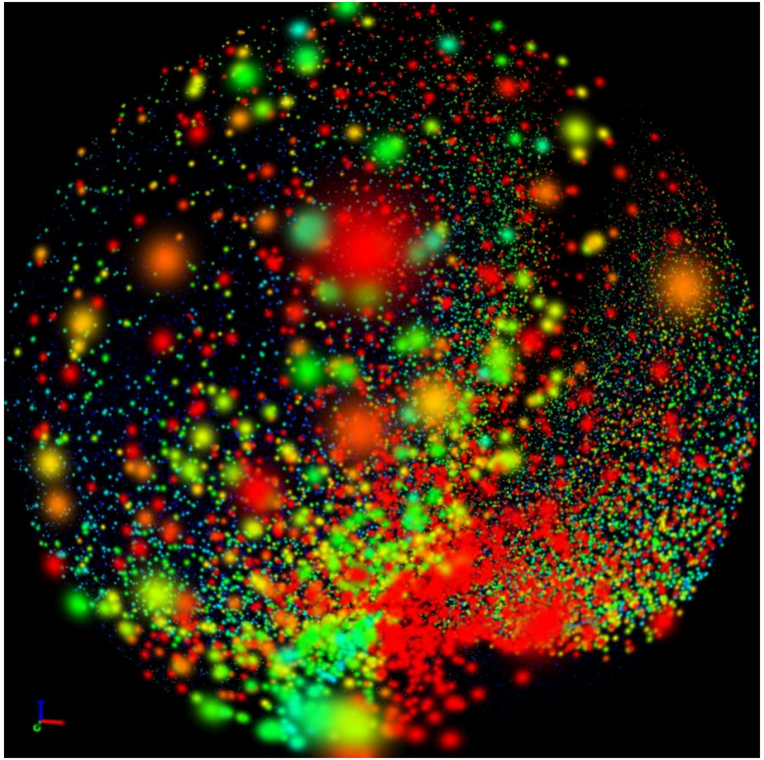




# Particle Rendering

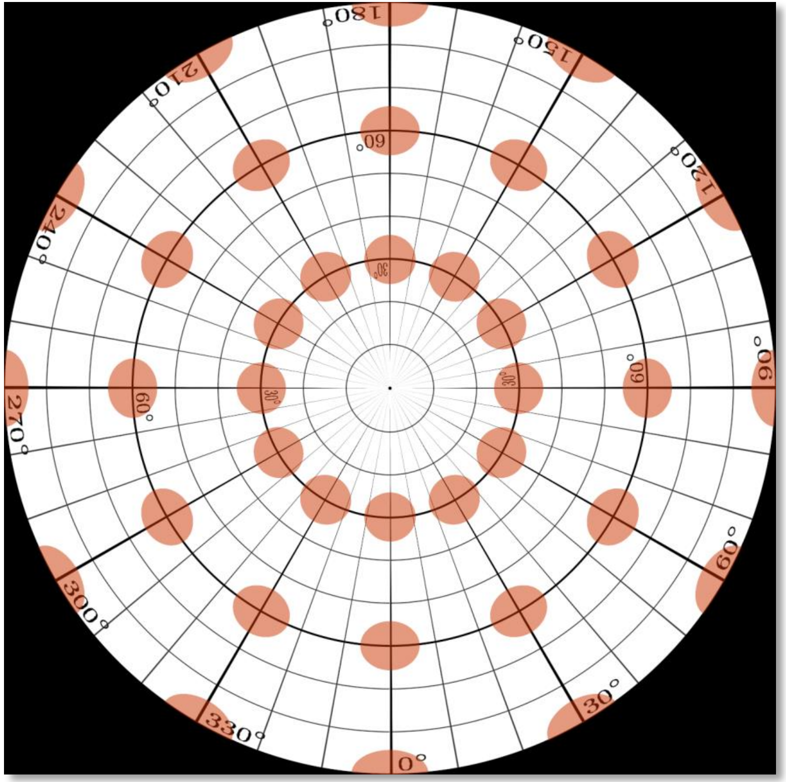
Star rendering using billboards (colored stars)





# Particle Rendering

Star rendering using billboards (geometry correction)

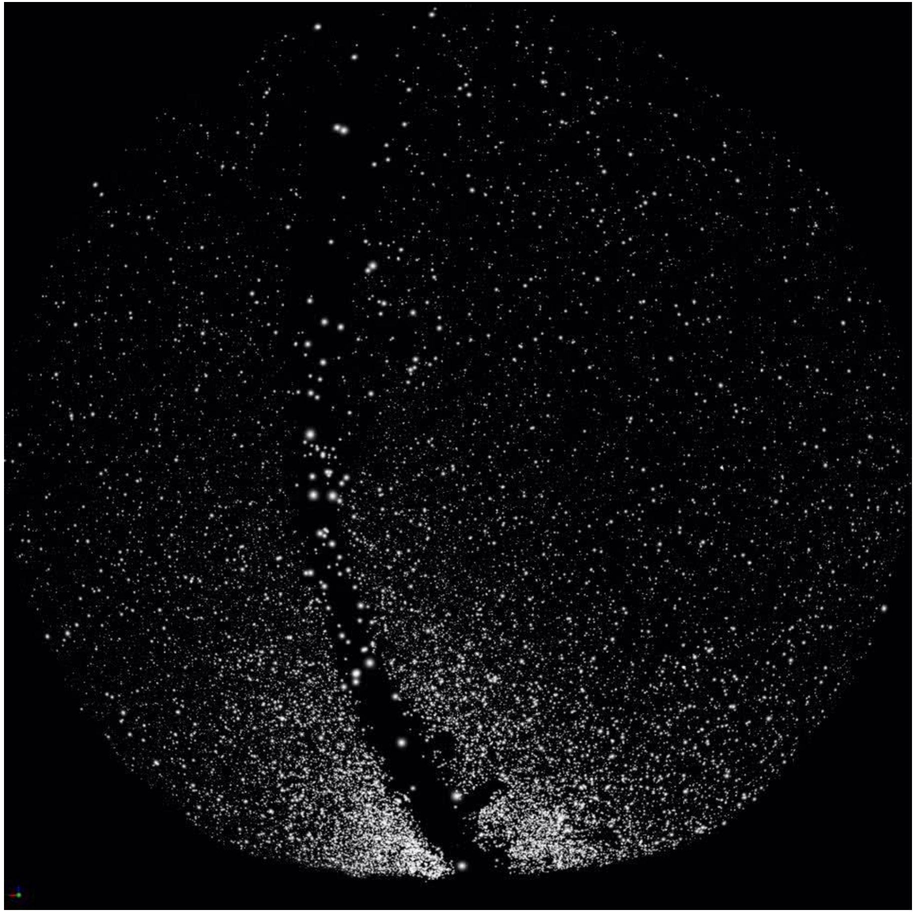


# RRLyrae stars from PanSTARRS (work in progress)

Visualization of RRLyrae stars (H.W. Rix)

- about 44000 stars

- Star sizes depend on distance to observer
- Star brightness decreases by distance and are faded to full transparent
- Sorting on CPU; better on GPU



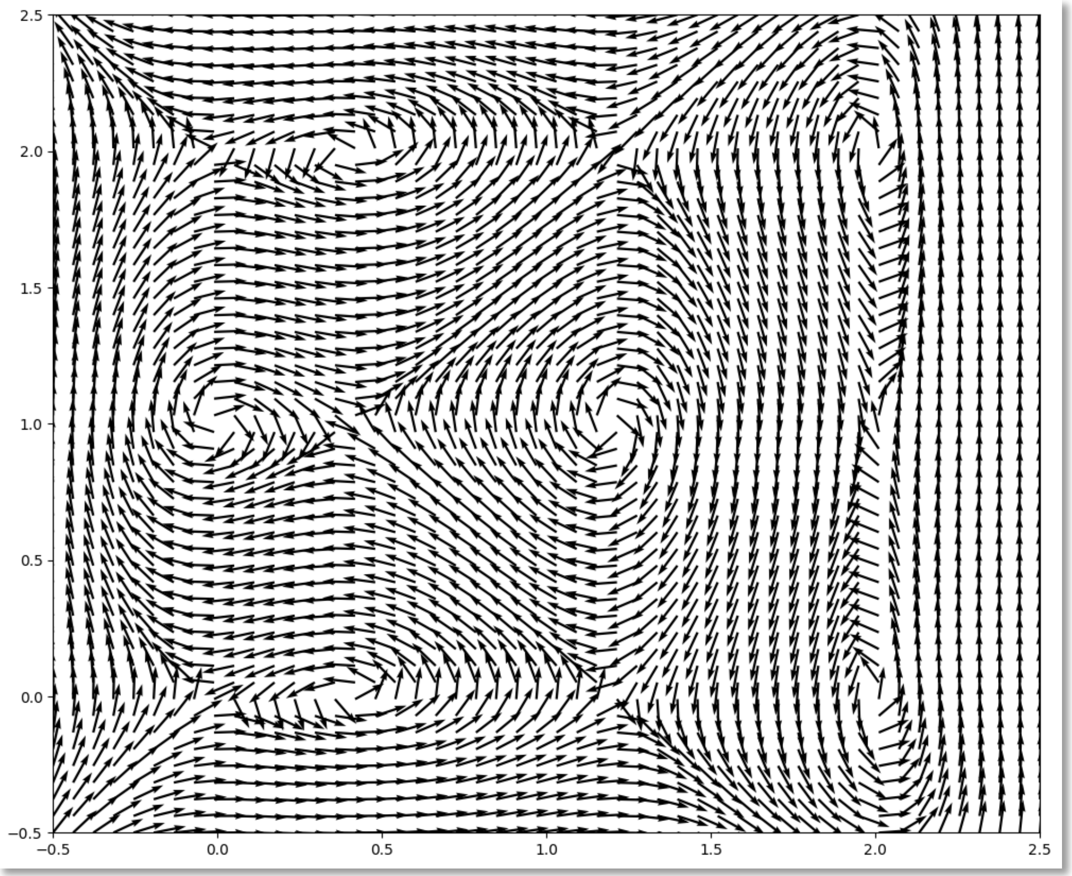
# Visualization of vector fields

Vector fields

$\vec{x} \mapsto \vec{\sigma}(\vec{x})\ \forall \vec{x} \in \mathbb{D}$

- Every arrow indicates the vector $\vec{\sigma}$ at $\vec{x}$.
- $\vec{\sigma}$ can be normalized.

- critical points are difficult to determine



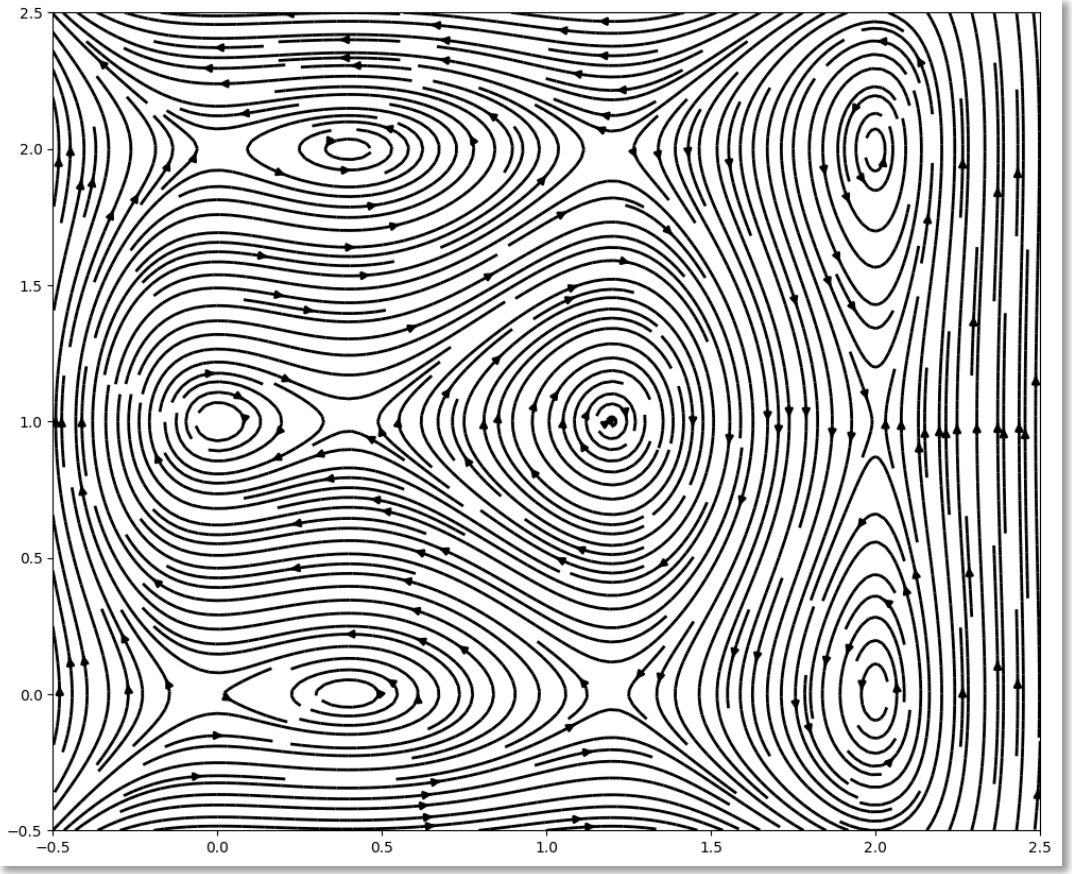
# Visualization of vector fields

Vector fields

$\vec{x} \mapsto \vec{\sigma}(\vec{x})\ \forall \vec{x} \in \mathbb{D}$

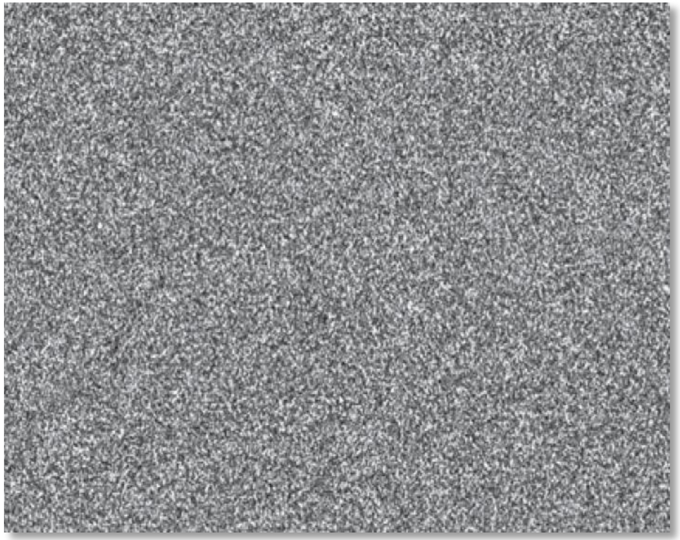
- Stream lines are integral lines along the vector field $\vec{\sigma}$

- Stream line placement is important!

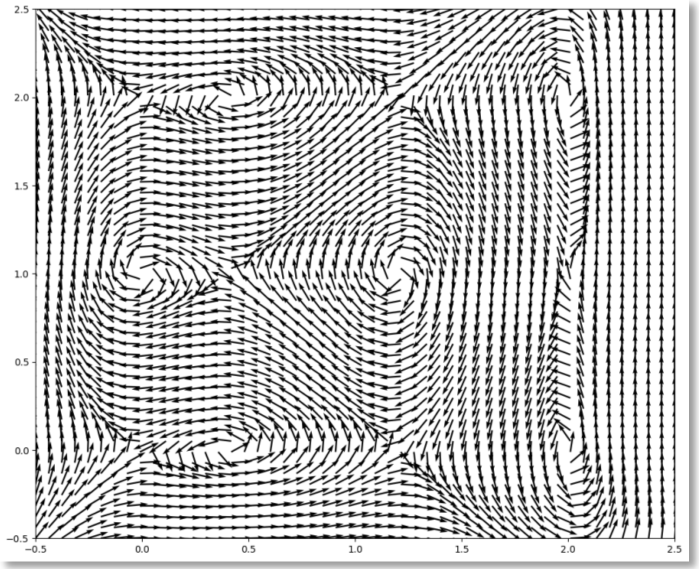


# Visualization of vector fields

Line integral convolution (LIC)



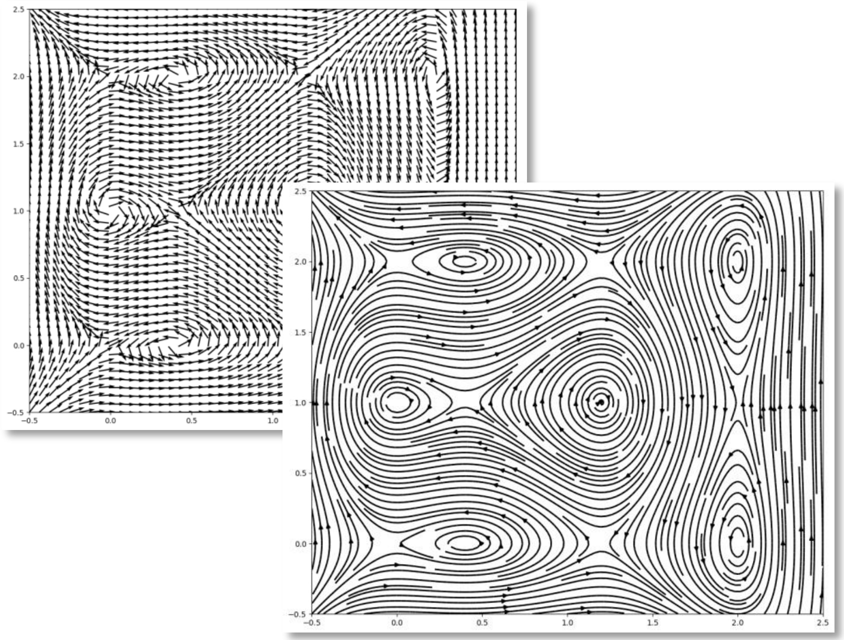
*

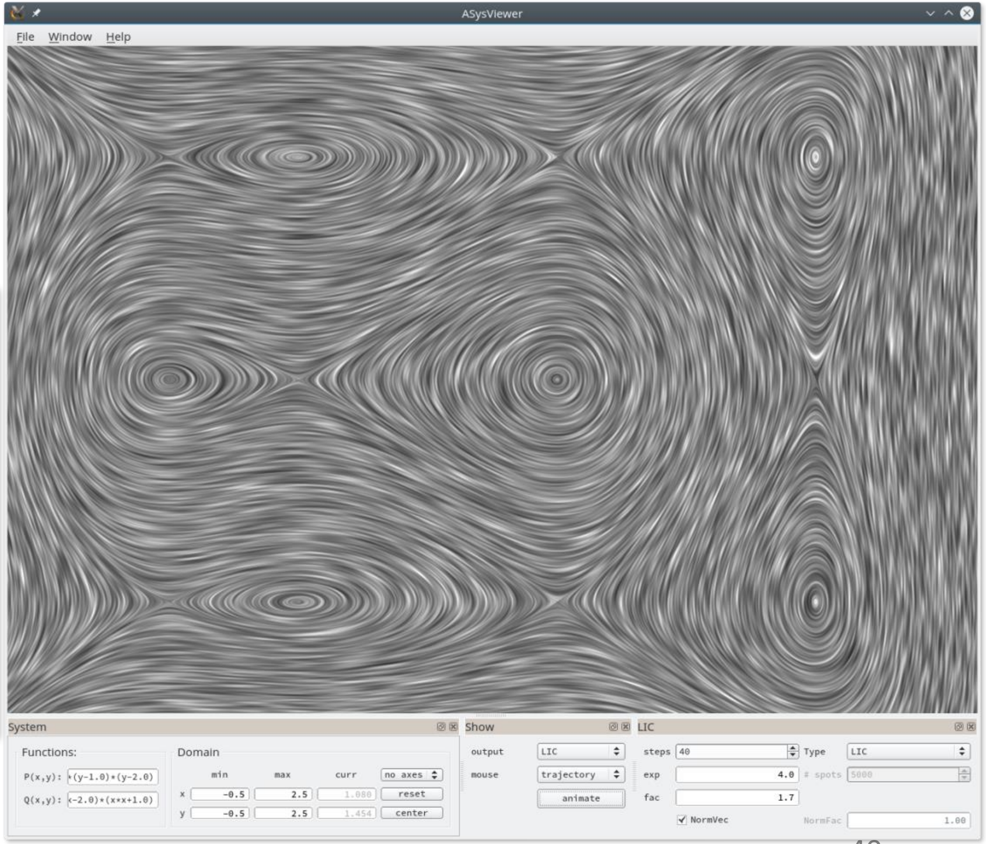



⇒ strong correlation along vector field;  no correlation perpendicular

# Visualization of vector fields

Line integral convolution





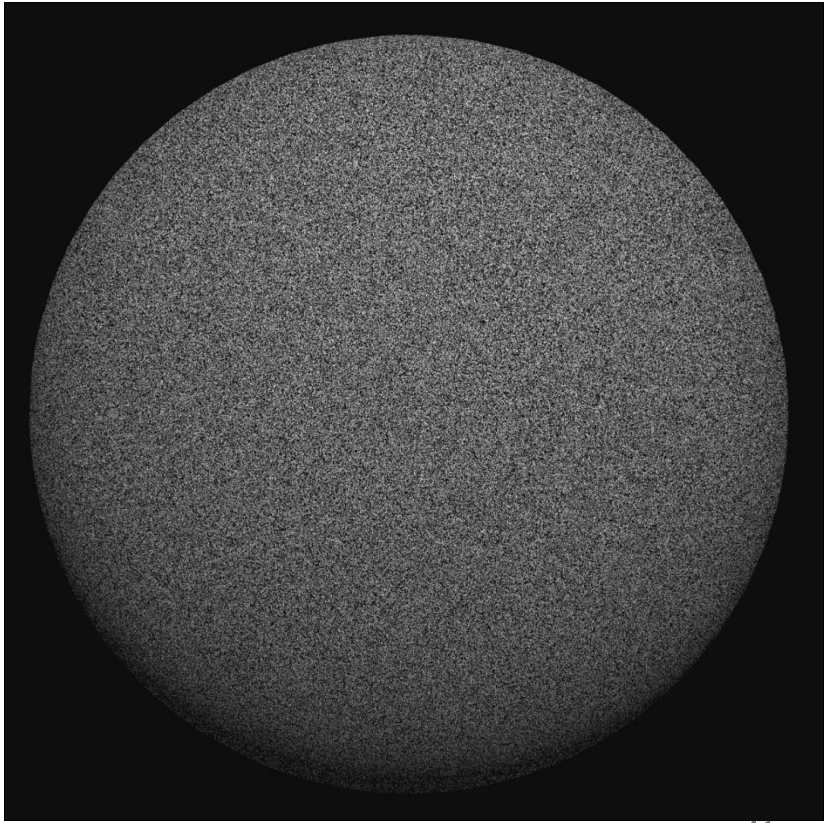

Animated / oriented LIC to keep direction information.

EJP 36, 035007 (2015)

# Visualization of vector fields (work in progress)

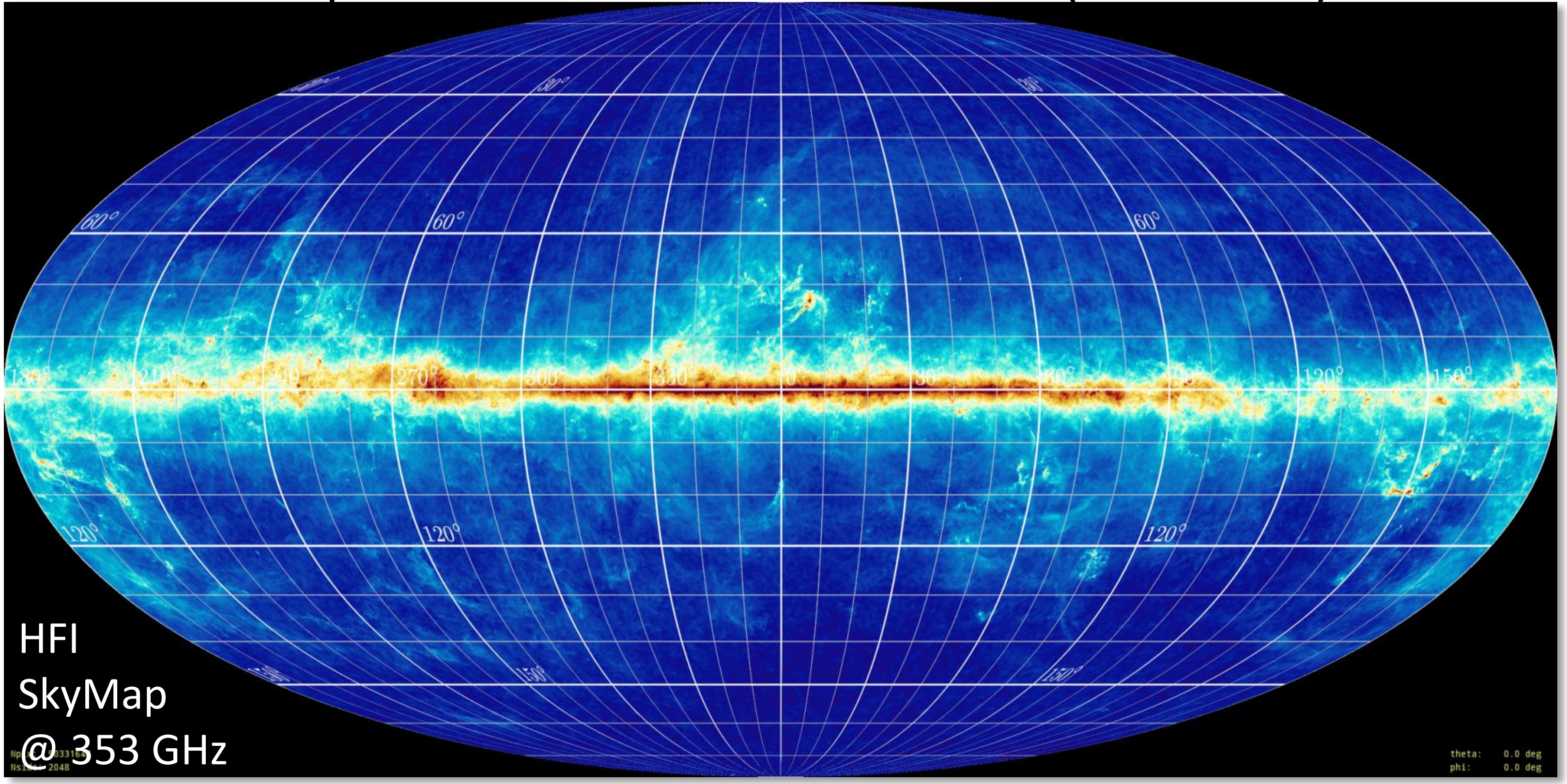
Line integral convolution on spheres

- Integrate along surface of sphere
- implement in CUDA
- calculate LIC before mapping



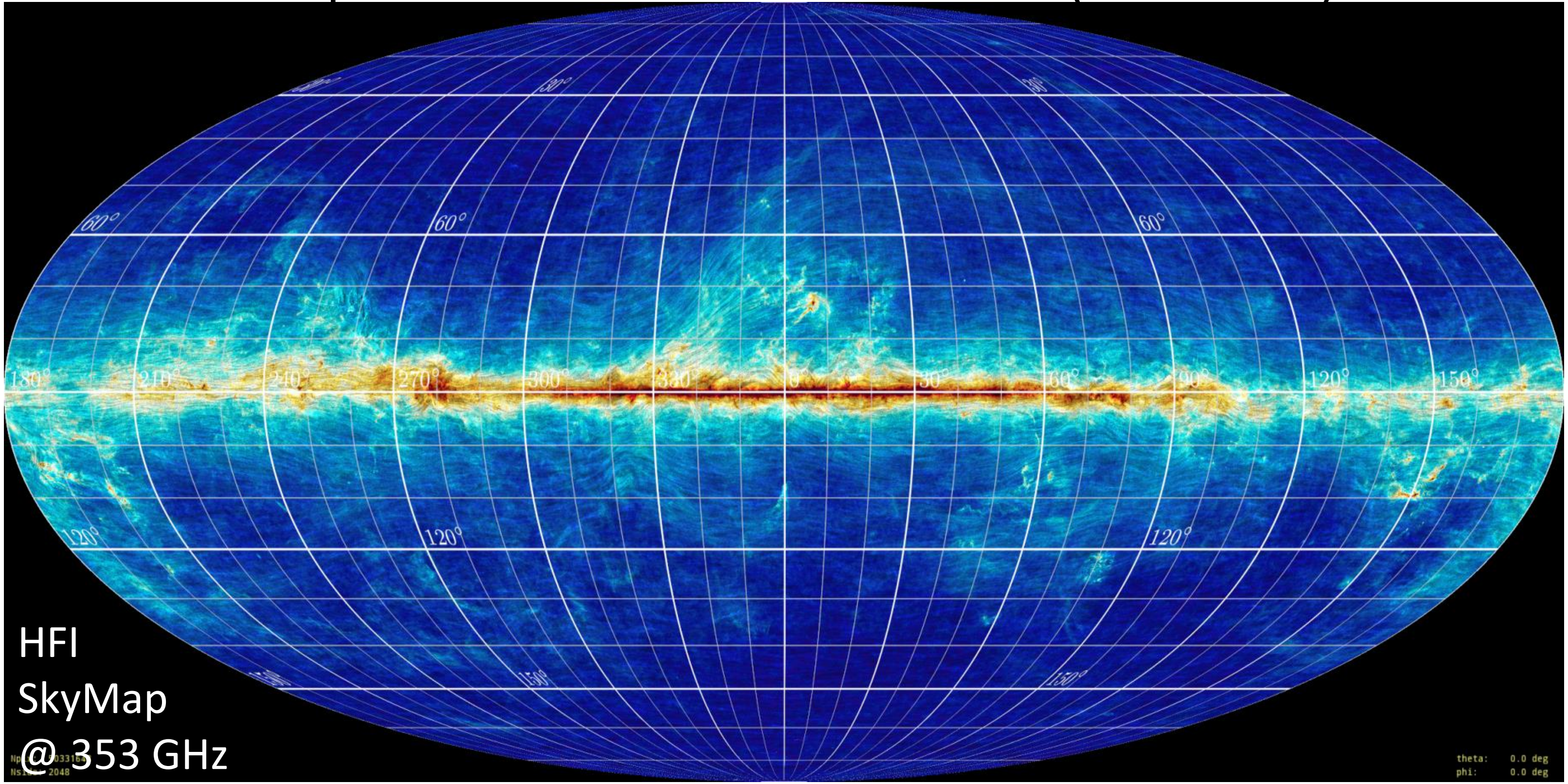
# Planck data / polarization

Visualization of polarization data of Planck data (Juan Soler)



# Planck data / polarization

Visualization of polarization data of Planck data (Juan Soler)

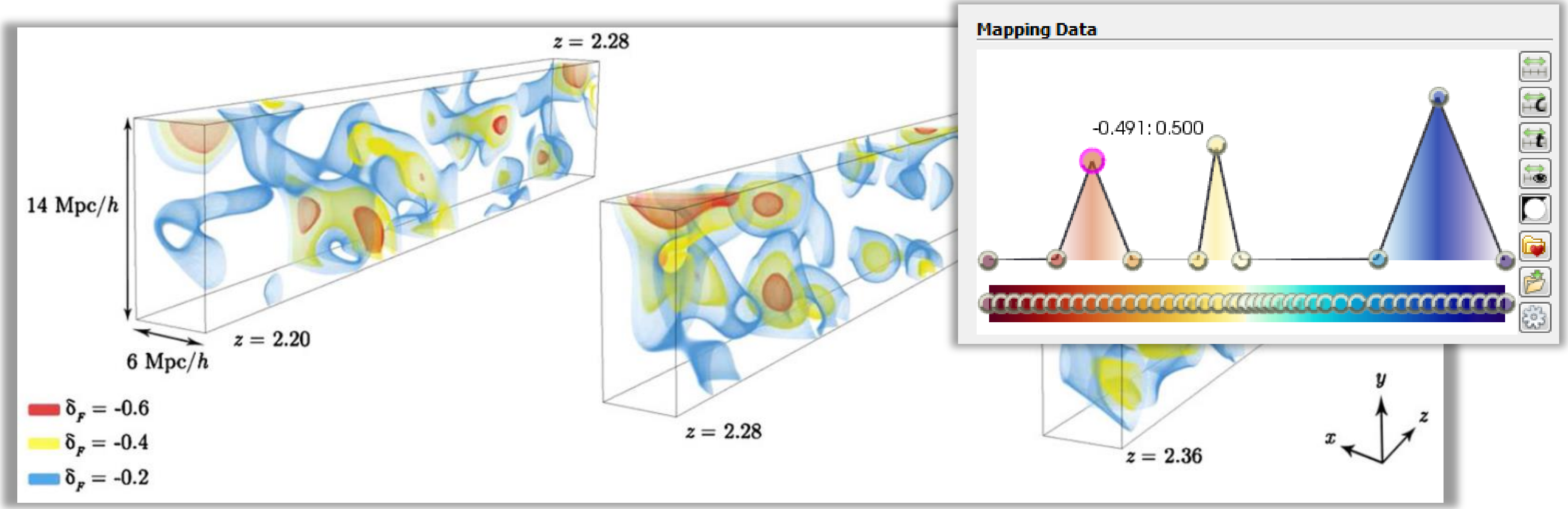



HFI
SkyMap
@ 353 GHz

# IGM Tomography (ongoing work)

High quality volume vis of IGM Tomography (K.G. Lee, Joe Hennawi)

- Volume data set (Cartesian grid, 680 x 48 x 36)

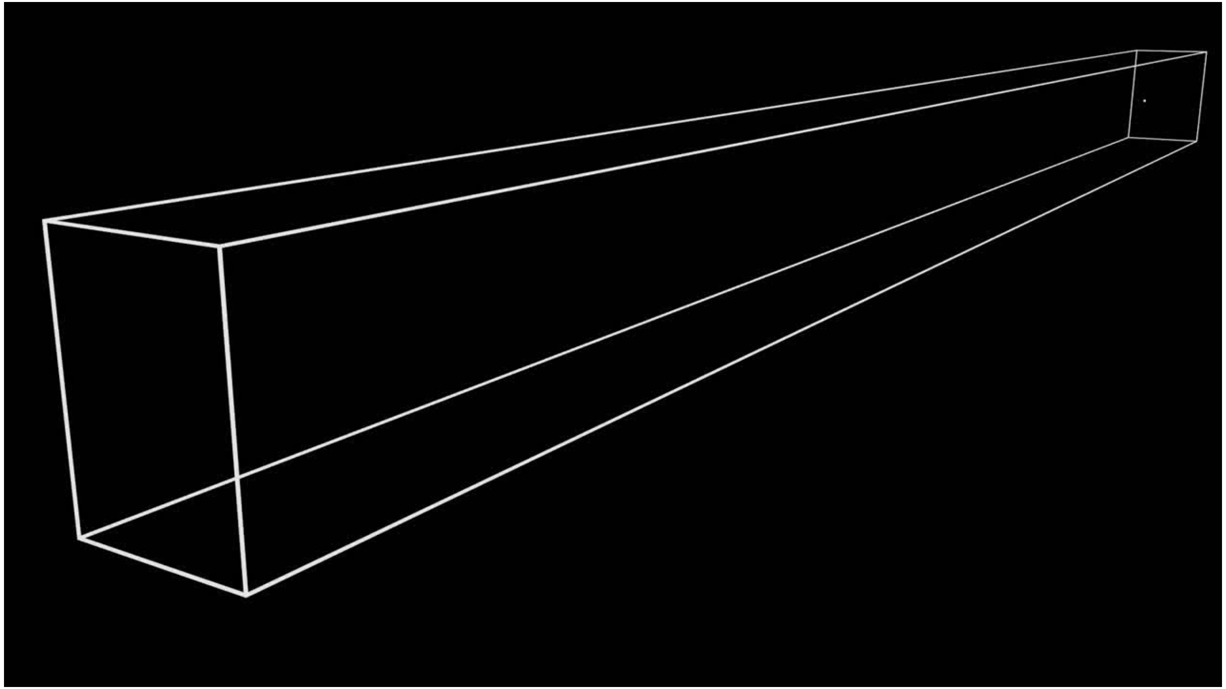



K.G. Lee et al, "LYalpha Forest Tomograpy from background galaxies: The first megaparsec-resolution large-scale structure map at z > 2" ApJL 795:L12 (2014)

# IGM Tomography (ongoing work)

High quality volume vis of IGM Tomography (K.G. Lee, Joe Hennawi)

- Volume data set (Cartesian grid, 680 x 48 x 36)

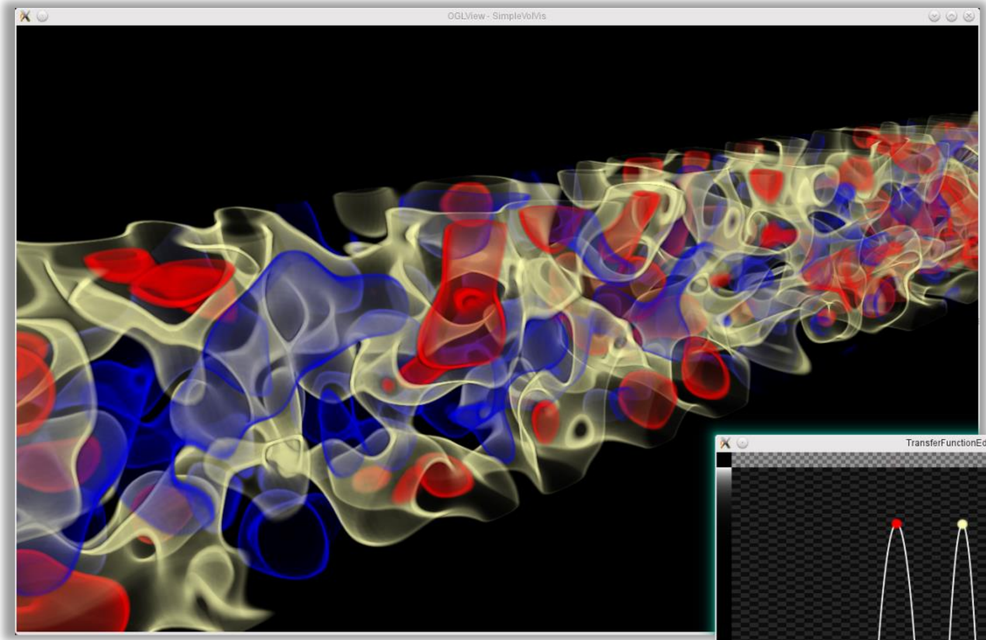


# IGM Tomography (ongoing work)

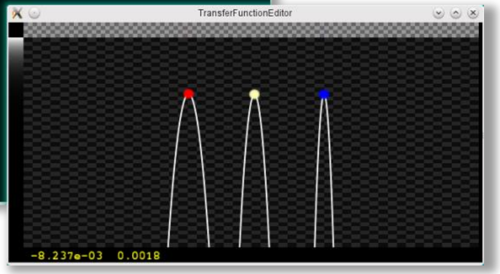
High quality volume vis of IGM Tomography (K.G. Lee, Joe Hennawi)

- Volume data set (Cartesian grid, 680 x 48 x 36)

VolVis with own framework (OGLView)




Gauss curve TF editor for isosurfaces



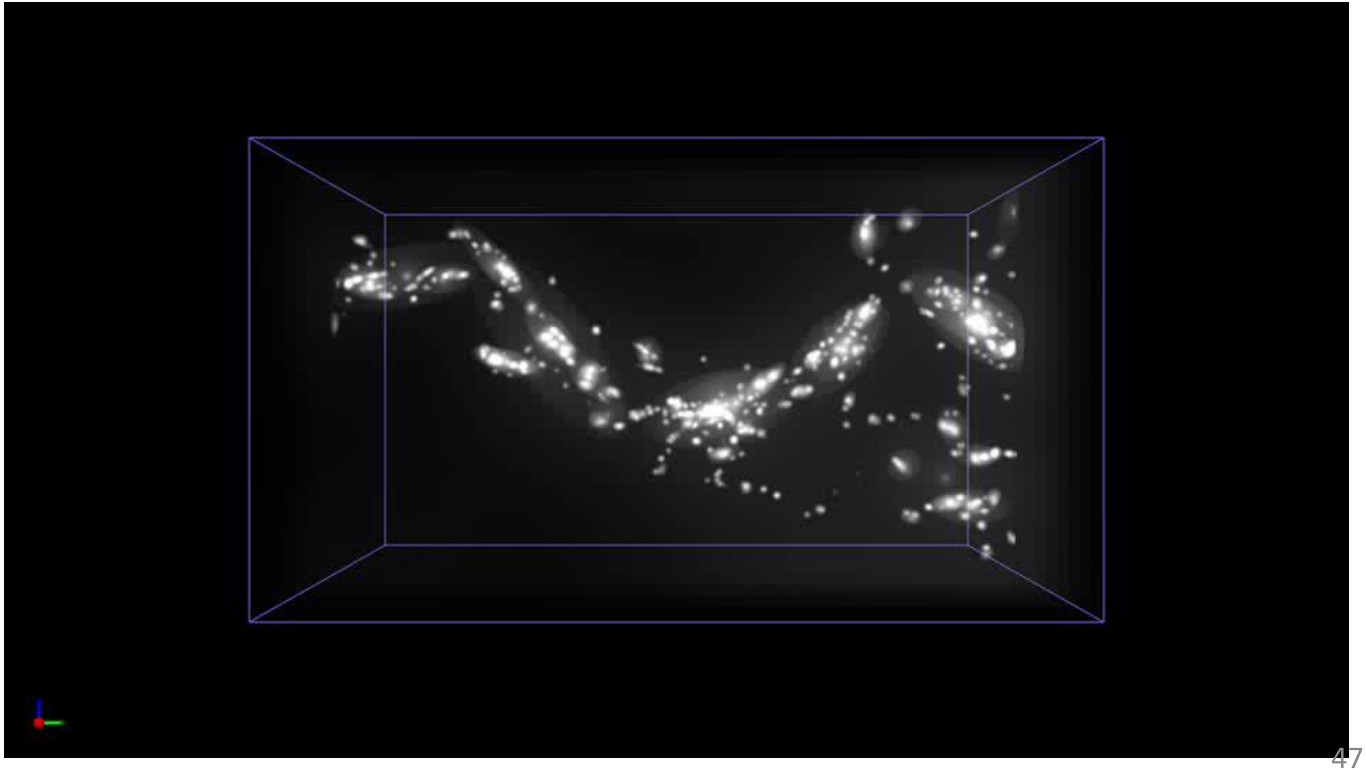
# Filamentary structures in the interstellar medium

Volume vis of filamentary structures in the ISM (Jouni Kainulainen)

Cartesian grid

1024 x 581 x 581

MIP-rendering



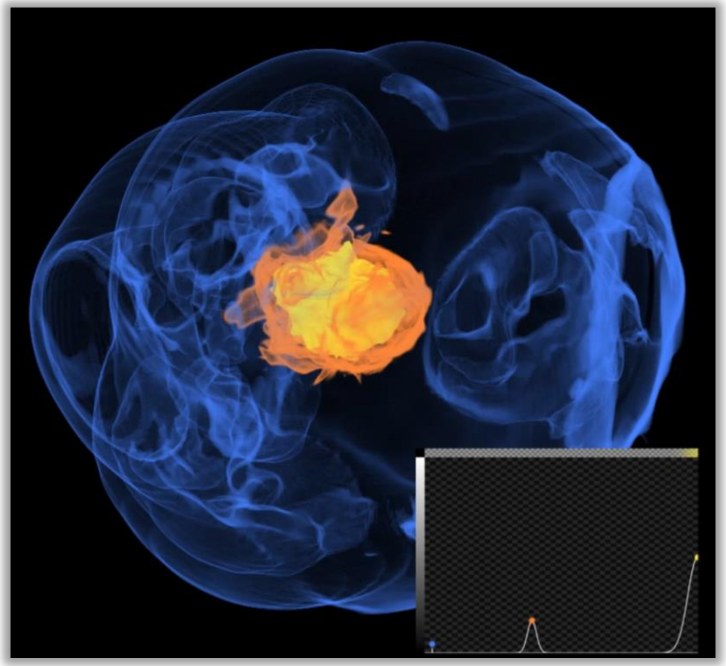
# Core collapse supernova (work in progress)

Volume vis of supernova simulation (A. Summa, T. Melson, MPA)

- Volume data set (spherical grid, 358 x 186 x 522)

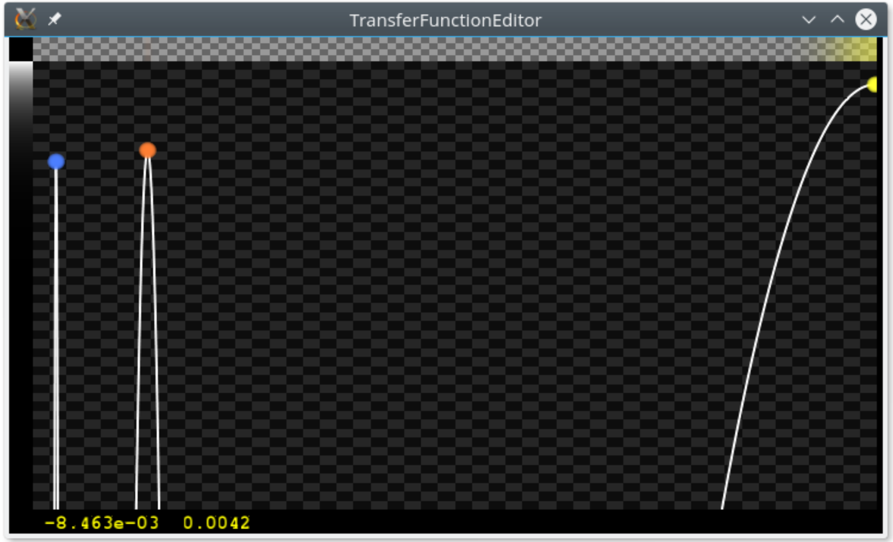
Aims:

- improve rendering performance, develop improved rendering techniques for spherical grids (OGLView: 1 sec @ 800 x 800 pixels)
- lower res during mouse interaction
- improve rendering quality (rendering artifacts due to undersampling, high freqs)
- improve visual depth perception (e.g. with ambient occlusion)

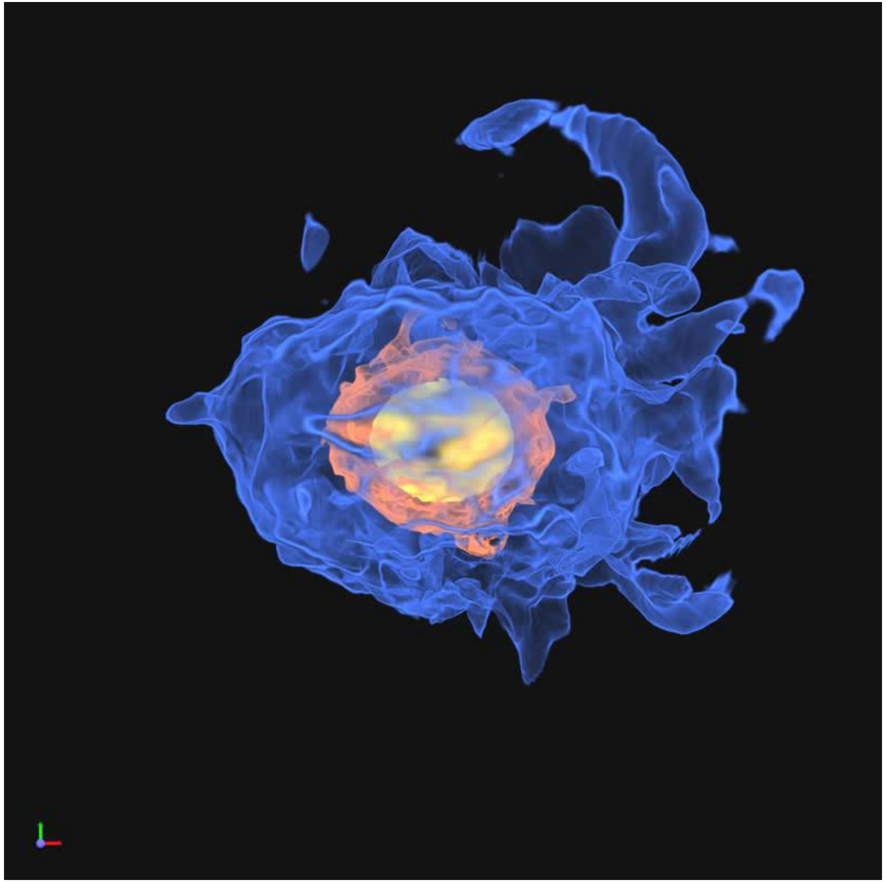


# Core collapse supernova (work in progress)

Volume vis of supernova simulation (A. Summa, T. Melson, MPA)

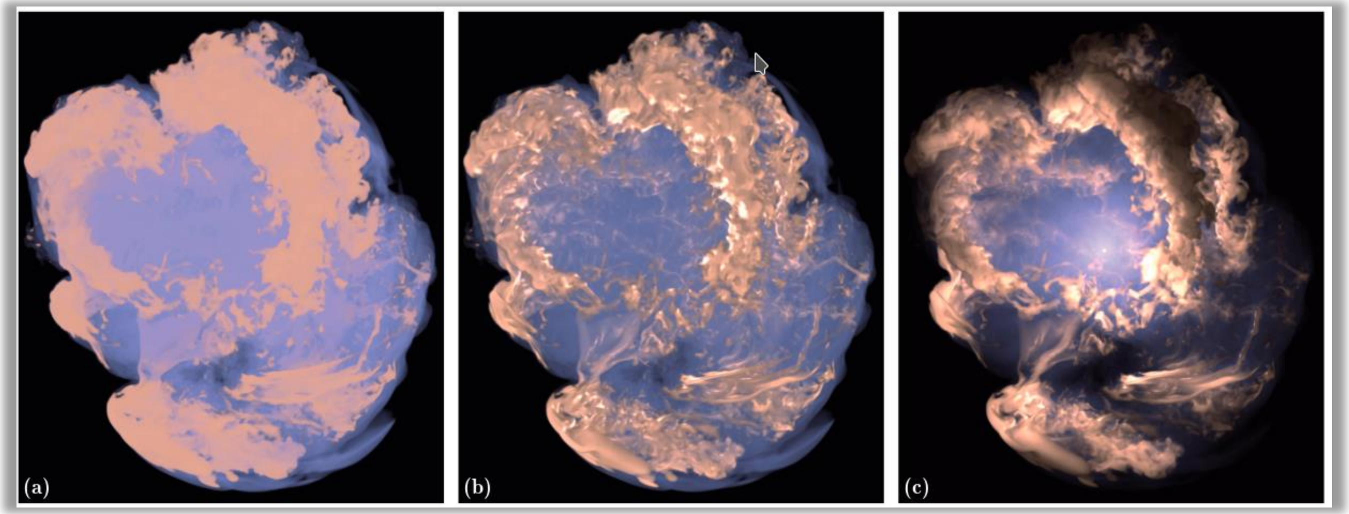



Gaussian functions (log-plot)



# Core collapse supernova (work in progress)

Improve visual depth perception (with different shading techniques)




Supernova simulation

a) Standard emission-absorption model
b) Volumetric ambient occlusion
c) Ambient scattering with light source

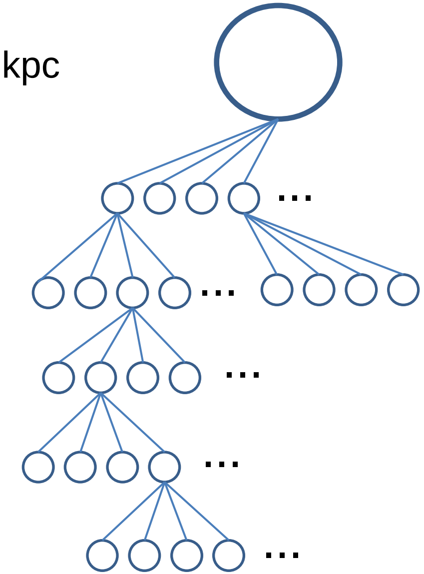
Ament et al., „Ambient Volume Scattering“,
IEEE Transactions on Visualization and Computer Graphics 19, Issue 12 (2013)

# Density structure of the CGM

## Hierarchical cloud data visualization (Jonathan Stern, Joe Hennawi)

- Hierarchical cloud data of circum galactic medium

Main cloud: R = 280 kpc



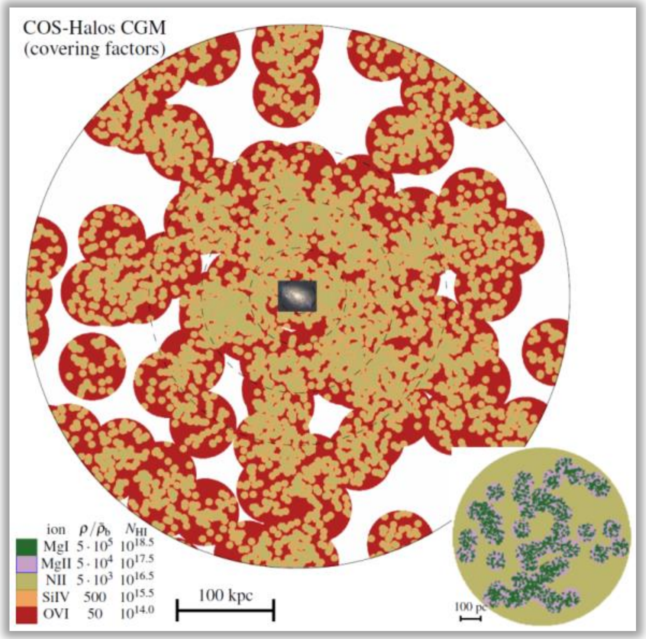
57 clouds = 57 (L0, 35 kpc)

57 x 45 clouds = 2,565 (L1, 3.9 kpc)

57 x 45 x 45 clouds = 115,425 (L2, 440 pc)

etc. = 5,184,125 (L3, 49 pc)

= 233,735,625 (L4, 5.5 pc)




* http://www2.mpia-hd.mpg.de/homes/stern/CGM.html

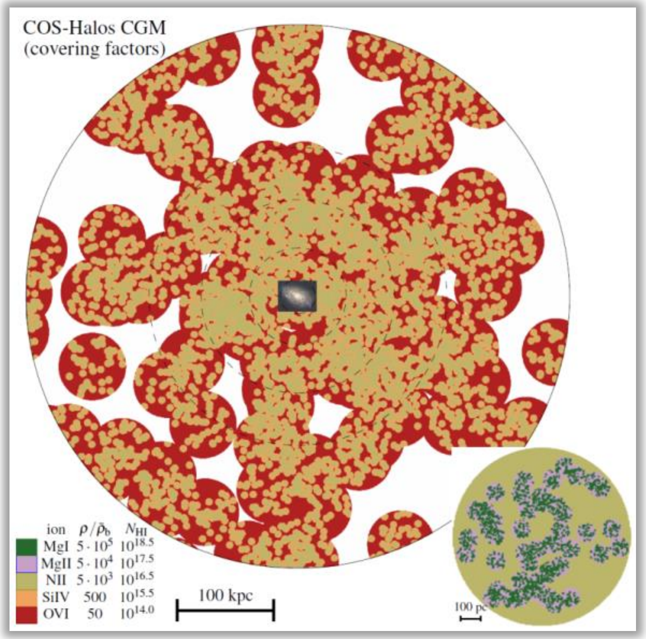
# Density structure of the CGM

Hierarchical cloud data visualization (Jonathan Stern, Joe Hennawi)

- Hierarchical cloud data
- Sampling on 1024 x 1024 x 1024, + VolVis

  1024 pix = 560,000 pc
  0.02 pix = 11 pc (L4)

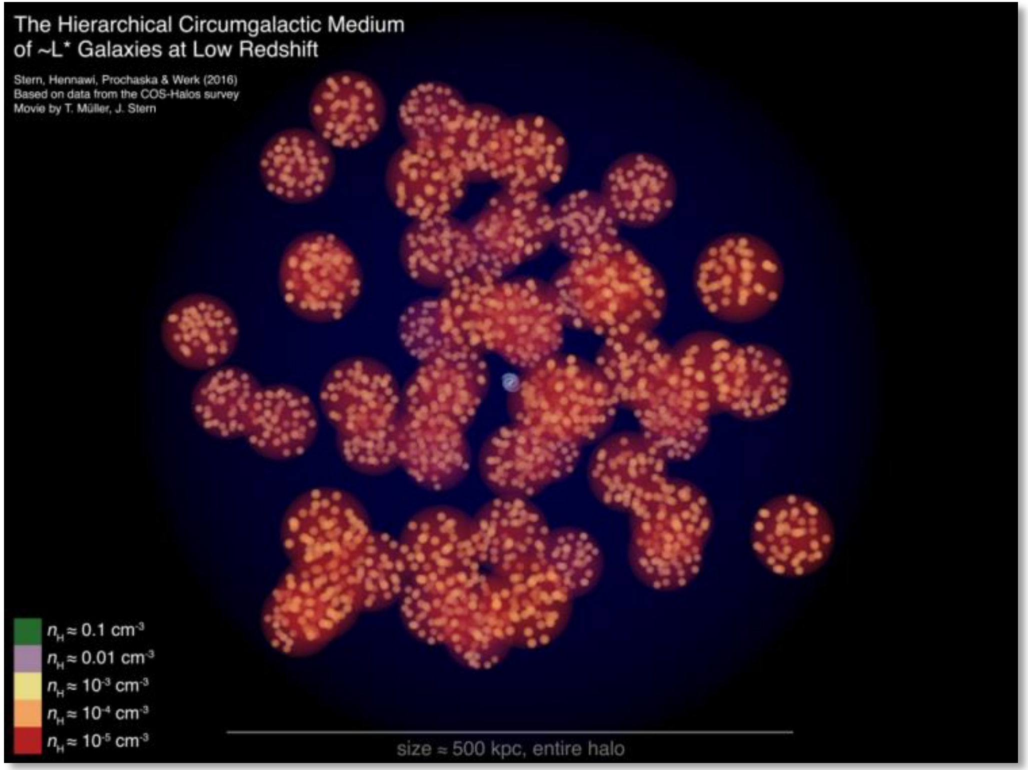
- Develop new rendering technique based on
  - view frustum culling
  - hierarchical loading (clouds largen than 1 pixel)
  - calculate sphere intersections
  - sort intersections for correct blending
  - ray casting with early ray termination




* http://www2.mpia-hd.mpg.de/homes/stern/CGM.html

# Density structure of the CGM

Hierarchical cloud data visualization (Jonathan Stern, Joe Hennawi)




Rendering performance:

40 fps @ 400 x 300 pixels
10 fps @ 1280 x 960 pixels

Annotations and movie editing with Adobe AfterEffects

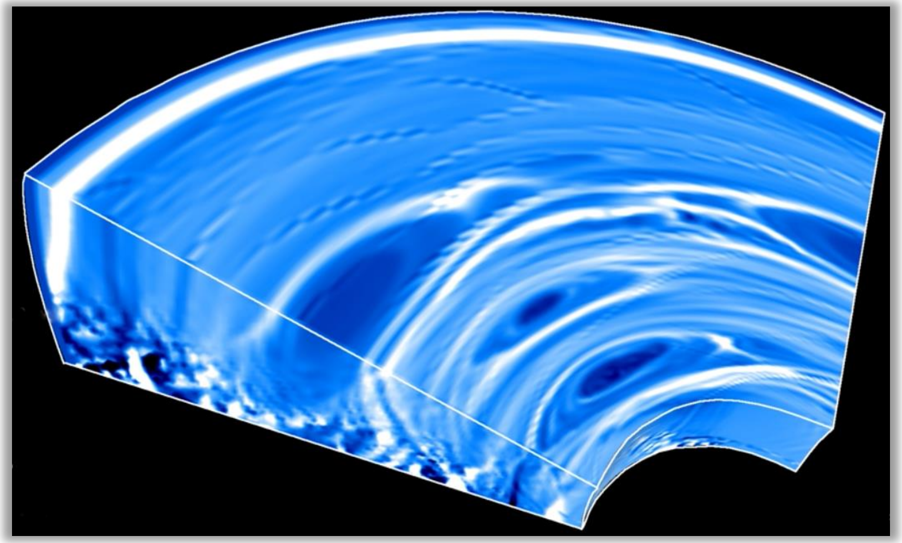
# Vorticity visualization (work in progress)

Vorticity visualization (Hubert Klahr) of high resolution data:

pluto code:

(density, vr, vt, vp, prs) @ 1024 x 512 x 256, spherical grid

- Vorticity: $\vec{\omega} = \vec{\nabla} \times \vec{v}$

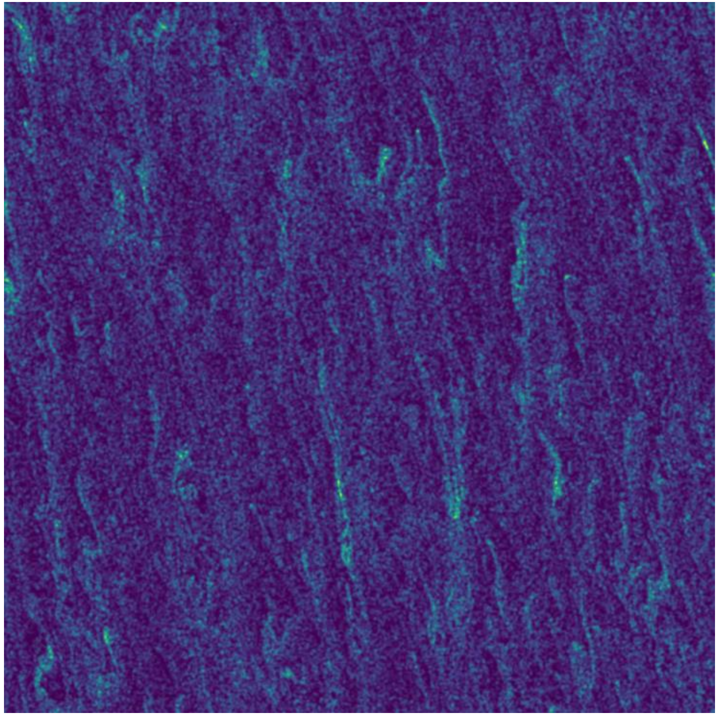


- Develop vortex detection algorithms for planetary disks
- Student project (Ramin Safarpour, master thesis) in cooperation with IWR (F. Sadlo)

# Planetesimal development (work in progress)

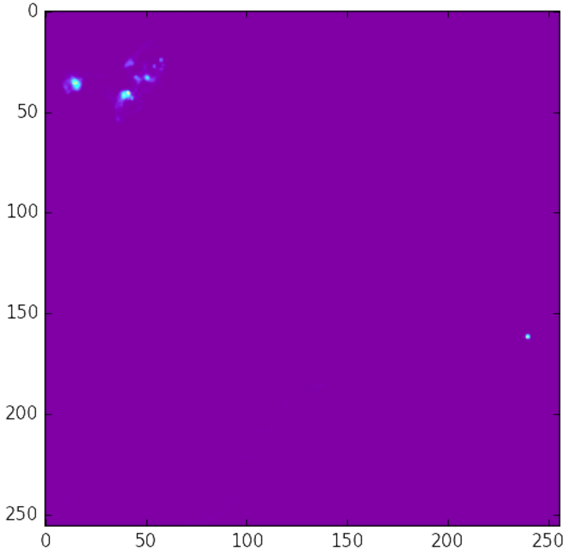
Visualization of planetesimal development
(Patrick Quicker (student apprentice), Andreas Schreiber)

- 2D particle simulation
- periodic boundary conditions in phi-direction; shear-periodic boundary conditions in radial-direction
- track planetesimal
- 3d-stack (x,y + time)

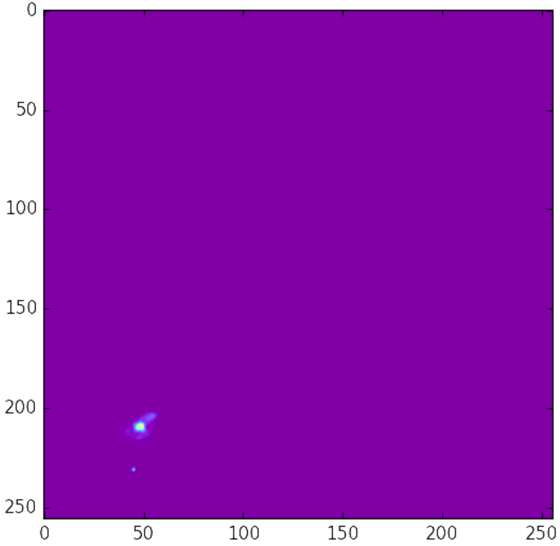


# Planetesimal development (work in progress)

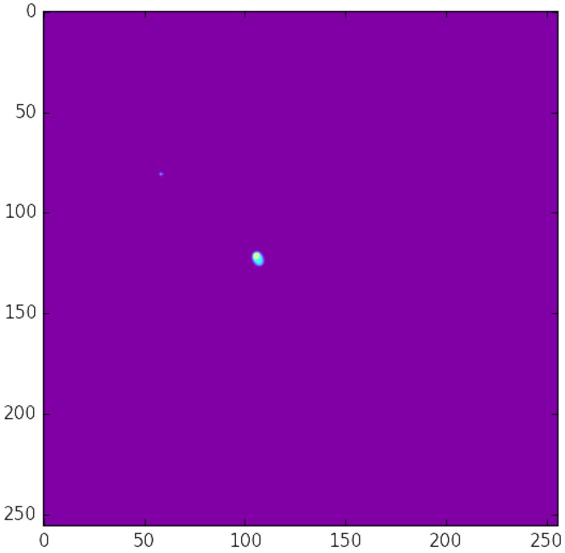
Visualization of planetesimal development
(Patrick Quicker (student apprentice), Andreas Schreiber)




$t = t_1$



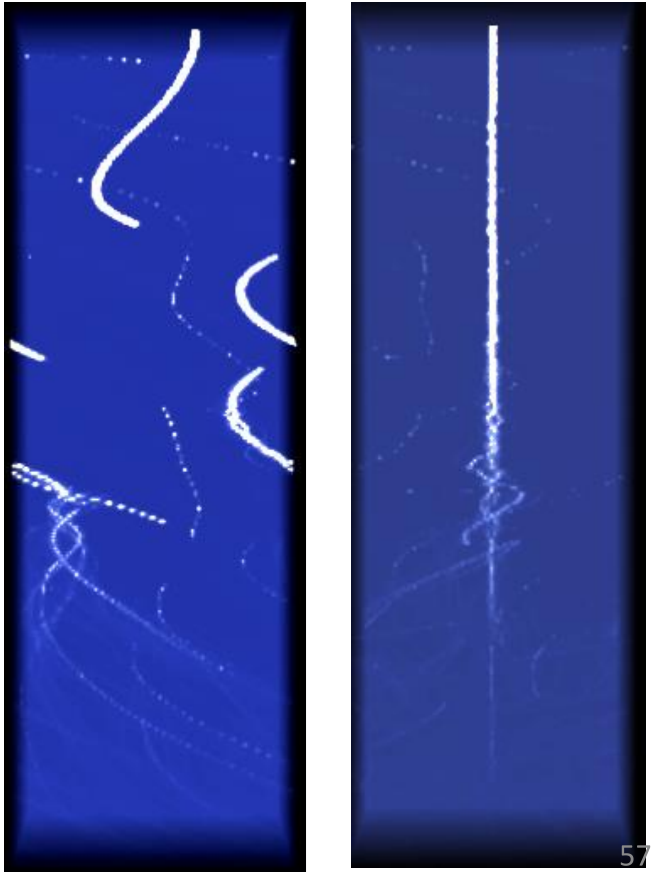

$t = t_2 > t_1$




$t = t_3 > t_2$

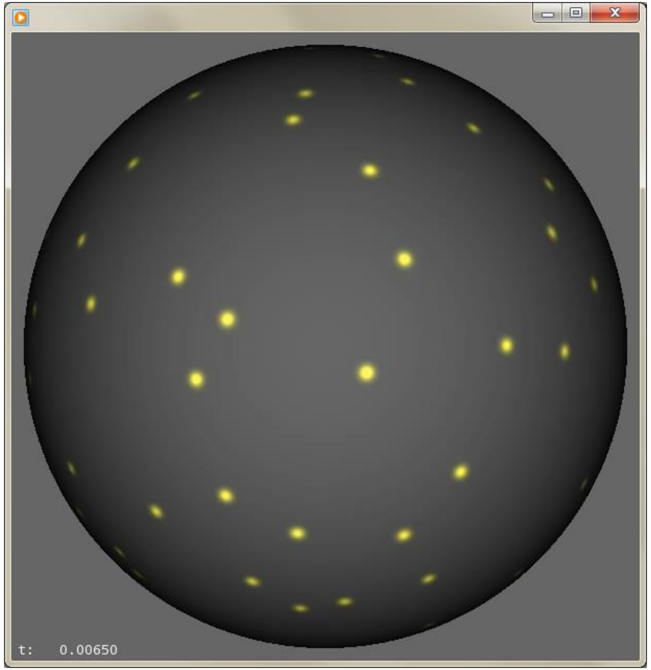
# Planetesimal development (work in progress)

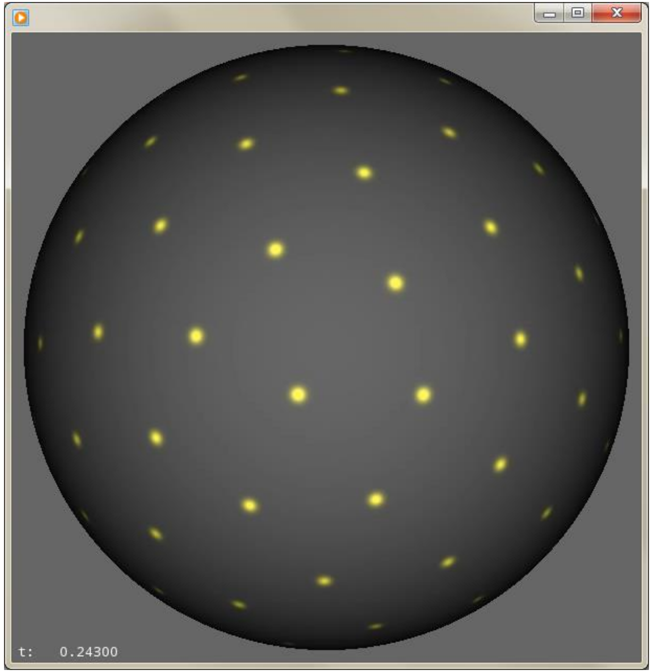
Visualization of planetesimal development

# Fulldome exoplanet orrery (Bachelor thesis)

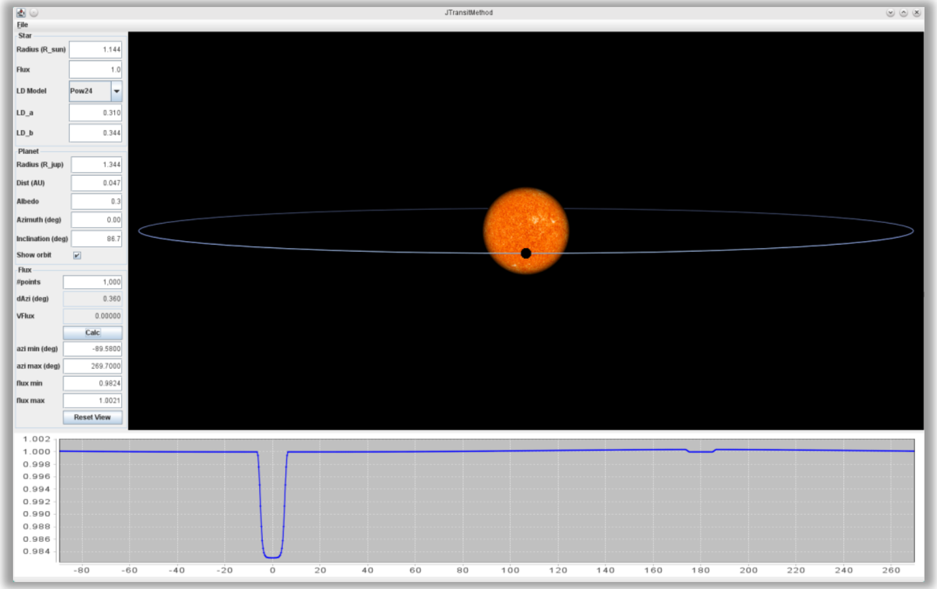
Charged particle motion on sphere



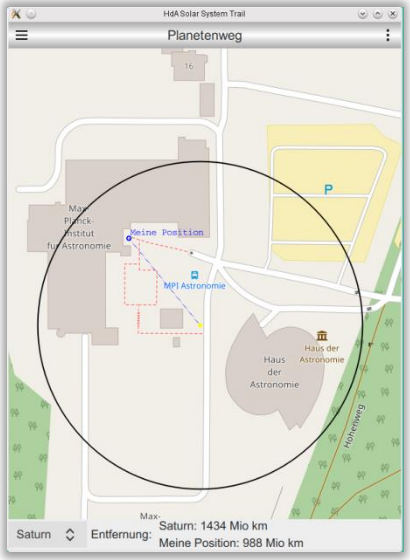



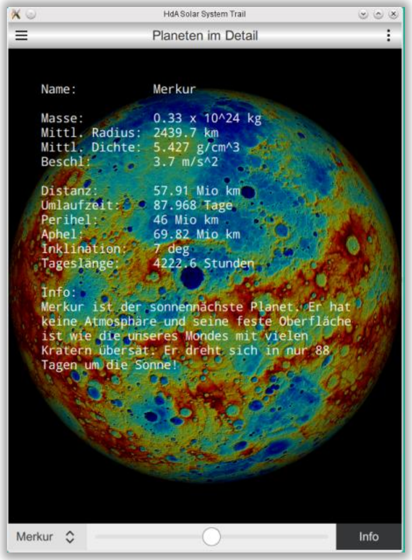
# Vis for education – interactive apps

Java-App to demonstrate the transit method



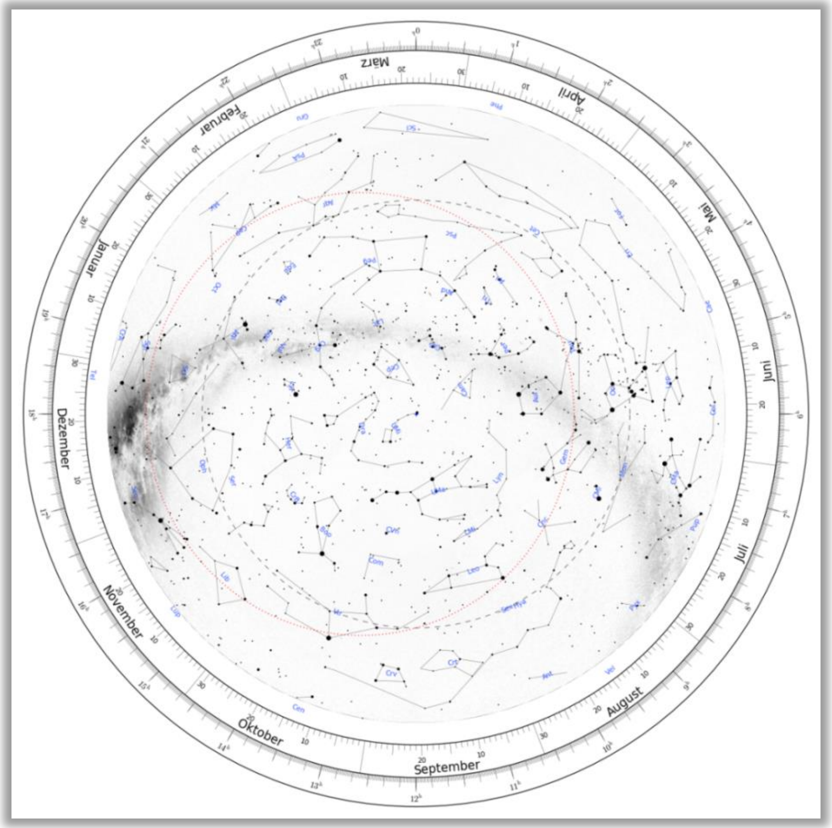

Android-App for our Solar System Trail

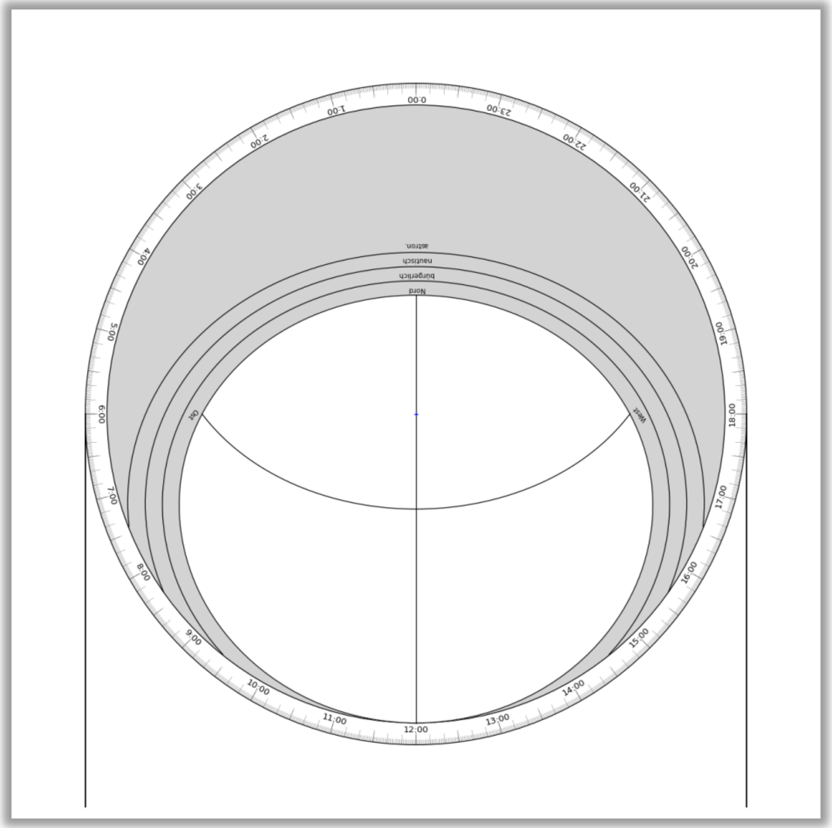


# Vis for education – diagrams

Highly configurable stellar map (python, matplotlib)





# Fulldome Demo