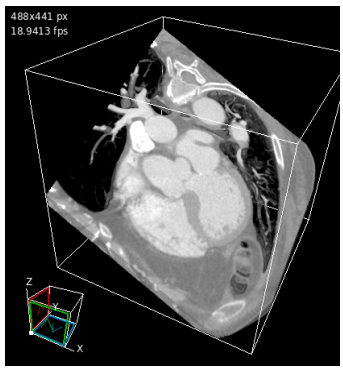
## Assignment 4

Due at 11:59 pm on May 21, 2013.
**Email screenshots to Stephan Arens <stephan.arens@uni-paderborn.de>**

# Volume Visualization

On a Linux pool PC download and unzip *volumestudio.tar.gz* from koaLA. Download and unzip the content of *hdz2_1_256.mhd.tar.gz* right into the previously extracted folder *bin/release*. Run Volume Studio (click on *launcher*) and open the last opened pipeline *Vis_Assignment4.pipeline.json*.

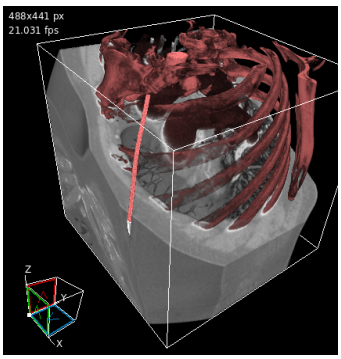
1. Transfer Function: Create a "Custom" element that maps data values to color (vec3) and alpha (float).
   - Color: The data range from 0.1 to 0.35 shall be linearly mapped to gray tones between black and white (i.e. values smaller than 0.1 are mapped to black, values bigger than 0.35 are mapped to white and e.g. 0.225 is mapped to a 50% gray).
   - Alpha: The data range from 0.1 to 0.35 shall be mapped the same way to alpha values between completely transparent (0.0) and completely opaque (1.0).
2. Slice View: Create another "Custom" element that only shows a slice of 10mm thickness by modifying the opacity values of the first element.
   - Make the slice parallel to the xy-plane by taking into account the global variable samplePosition.z when deciding if a sample stays opaque or gets transparent.
   - Let the user define the z-position of the slice by connecting a "Constant" element to your slicing "Custom" element.
   - For advanced students: Define an arbitrary oriented slice by computing the sample's distance to a user-defined plane. Use the "Constant" element for user-defined 3-component vectors. **Hint**: use the *dot* product to compute the distance *D* to the plane:




$$D = \left| \begin{pmatrix} n_{0_x} \\ n_{0_y} \\ n_{0_z} \\ -d \end{pmatrix} \cdot \begin{pmatrix} s_x \\ s_y \\ s_z \\ 1 \end{pmatrix} \right|$$

   where $n_0$ is the normalized normal to the plane, $d$ the directed distance of the plane to the origin and $s$ the sample's position.

3. Create a second transfer function that shows different data values (e.g. only the bones) and map it to a reddish color ramp. Combine all techniques showing one transfer function above the cutting plane and the other one underneath.



Some syntax of GLSL:

```
// vector operations
vec3 a = vec3(2.0, 1.5, -0.5);
vec3 b = vec3(1.0); // same as vec3(1.0, 1.0, 1.0)
vec4 c = vec4(a, b.x); // use .xyzw or .rgba to adress components
c *= c; // = vec4(4.0, 2.25, 0.25, 1.0)

// useful functions
float d = dot(a, b); // dot product
a = normalize(a); // same as a /= length(a)
d = clamp(d, 0.0, 1.0) // same as d = min(max(d, 0.0), 1.0)
a = abs(a); // use the absolute values of a
```