data at this stage. Some of the data processing operations can also be done outside the application. Transfer function lookup tables and fragment shaders are typically created in the Initialize stage of the application.

After initialization and every time viewing parameters change, the proxy geometry is computed and stored in vertex arrays. When the data set is stored as a 3D texture object, the proxy geometry consists of a set of polygons, slicing through the volume perpendicular to the viewing direction (see Section 39.4.2). Slice polygons are computed by first intersecting the slicing planes with the edges of the volume bounding box and then sorting the resulting vertices in a clockwise or counterclockwise direction around their center. For each vertex, the corresponding 3D texture coordinate is calculated on the CPU, in a vertex program, or via automatic texture-coordinate generation.

When a data set is stored as a set of 2D texture slices, the proxy polygons are simply rectangles aligned with the slices. Despite being faster, this approach has several disadvantages. First, it requires three times more memory, because the data slices need to be replicated along each principal direction. Data replication can be avoided with some performance overhead by reconstructing slices on the fly (Lefohn et al. 2004). Second, the sampling rate depends on the resolution of the volume. This problem can be solved by adding intermediate slices and performing trilinear interpolation with a fragment shader (Rezk-Salama et al. 2000). Third, the sampling distance changes with the viewpoint, resulting in intensity variations as the camera moves and image-popping artifacts when switching from one set of slices to another (Kniss et al. 2002b).

During the Update stage, textures are refreshed if the rendering mode or the transfer function parameters change. Also, opacity correction of the transfer function textures is performed if the sampling rate has changed (see Equation 3).

Before the slice polygons are drawn in sorted order, the rendering state needs to be set up appropriately. This step typically includes disabling lighting and culling, and setting up alpha blending. To blend in opaque geometry, depth testing has to be enabled, and writing to the depth buffer has to be disabled. Volume and transfer function textures have to be bound to texture units, which the fragment shader uses for input. At this point, shader input parameters are specified, and vertex arrays are set up for rendering. Finally, after the slices are drawn in sorted order, the rendering state is restored, so that the algorithm does not affect the display of other objects in the scene.