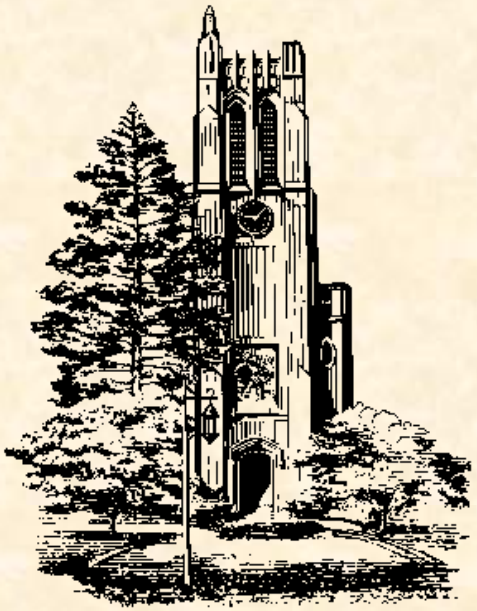
MICHIGAN STATE UNIVERSITY

Project Plan
Sparse Virtual Texturing



Team 2. Boeing

CSE 498, Collaborative Design

Adam Starks
James Drallos
Patrick Ohren
Alexander Kobylarek

Department of Computer Science and Engineering
Michigan State University

Fall 2009

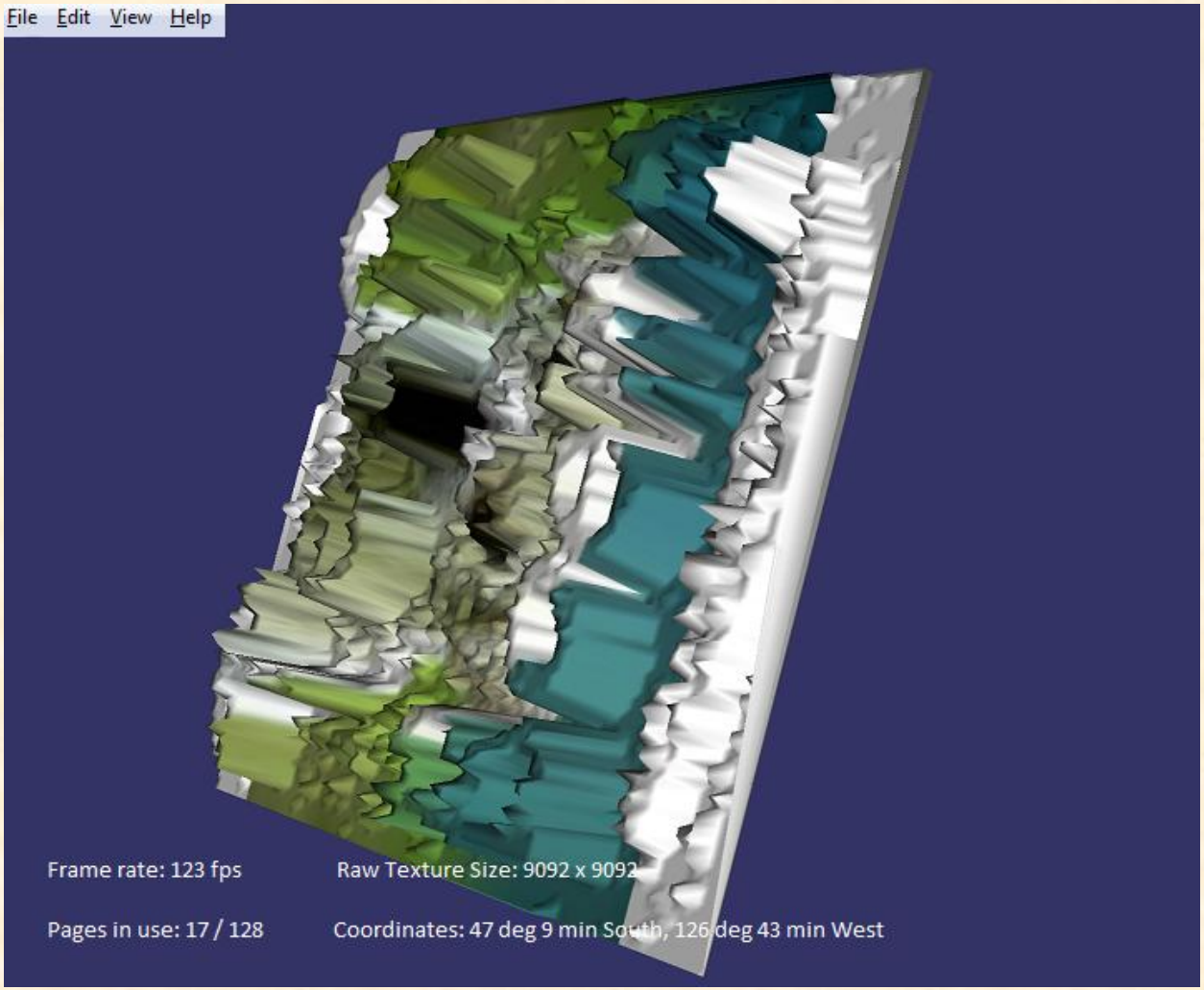
Functional Specifications

- Create a graphics library that implements Sparse Virtual Texturing (SVT); the use of large and/or highly detailed textures that wouldn't normally fit in texture memory.
- Convert SVT graphics library into an Open Scene Graph plug-in for use with all general OSG applications.
- Develop a visual application to demonstrate the functionality of our SVT graphics library.

Design Specifications

- Compatible with Open Scene Graph (OSG), ideally as a plug-in.
- Flexible enough for general use in a variety of Boeing visual applications.
- Display debug information, such as frame rate, page table info, mip-level
- Allow users to change key parameters to control quality and performance.

Screen Mockup



Team 2. Boeing

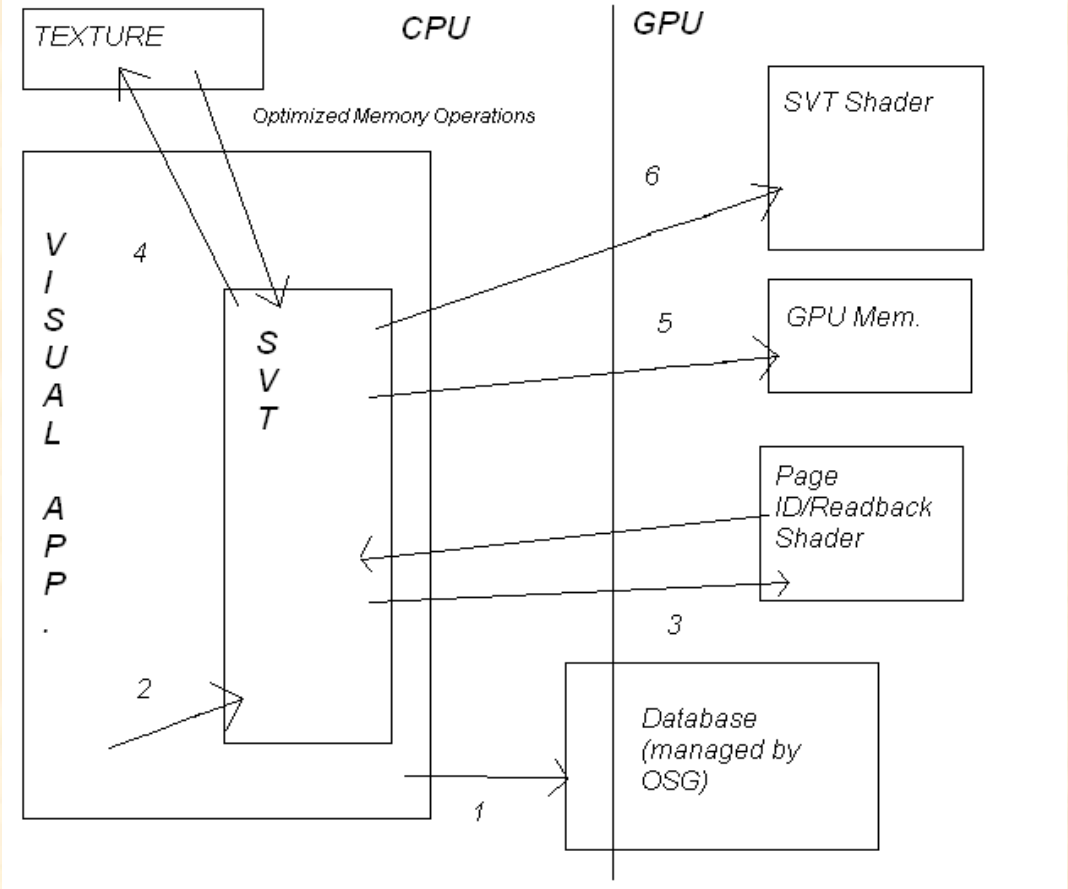
Technical Specifications

- Divide texture into pages of a uniform size
- Generate mip-map levels until highest mip-level occupies a single page
- Page Table stores addresses of each page, as well as whether a page is active, and if so what its mip level is
- Custom Readback shader stores requested texture coordinates in red and green channels
- CPU translates that data into sample counts for each page, used along with other information to determine desired mip level to load

Technical Specifications

- Requested texture pages spliced into one texture, with the page table keeping references to where each page is in the final texture
- Compress the texture real time for loading into texture memory, or load the texture in raw?
- Indirection shader will use page table to associate requested texture coordinates with actual loaded texture

Architecture Illustrated



Team 2. Boeing

System Components

- Hardware Platforms
  - x86, x64
- Software Platforms / Technologies
  - OpenGL
  - OpenSceneGraph
  - Visual Studio 2005

Testing

- Testing on Pixel Shader v2.0
- Multiple image formats
- Test on Little Endian and Big Endian systems
- Frequent demos to Boeing on their machines
- Performance testing on as many different machines as feasible

Risks

- Scene Processing
  - SSE intrinsics
- Moving Memory
  - Different memory layouts
  - Trial and error
- Lack of documentation
  - Utilize available literature and open source