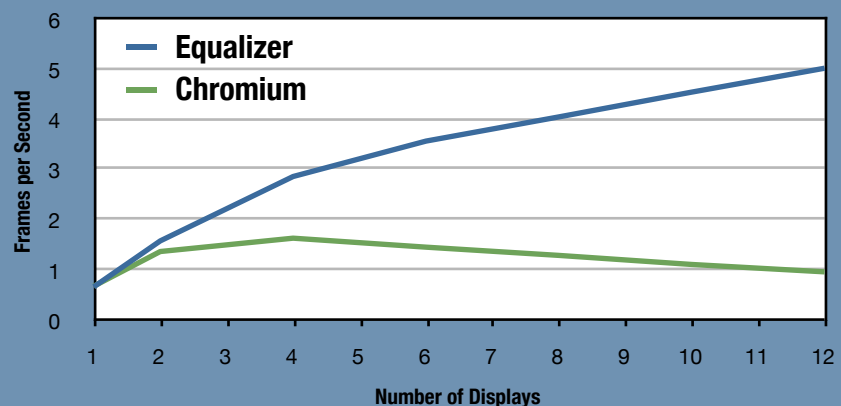


# Two Methods for driving OpenGL Display Walls

Parallelizing an OpenGL application provides significant performance benefits even for display walls.

Rendering Performance



## High-Resolution 3D Visualization

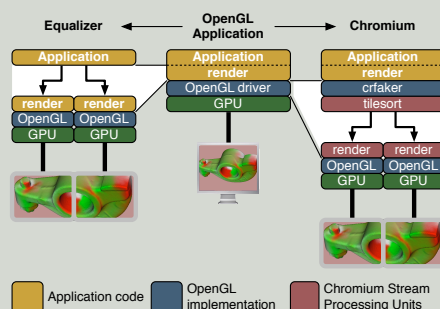
Chromium provides a transparent solution for using applications based on OpenGL 1.5 on display walls. Equalizer provides a minimally invasive API for high-performance, scalable OpenGL applications.

### Introduction

Chromium is an OpenGL implementation which streams graphics commands over the network. It implements the OpenGL 1.5 API and some extensions.

Equalizer is a framework for the development and deployment of parallel, scalable OpenGL applications.

Display walls and multi-projector systems are used to visualize complex data sets at a high resolution. Chromium provides a way to run legacy applications without any code modifications, while Equalizer allows the creation of high-performance, parallel applications with minimal code changes.



### Benchmark Setup

This benchmark used a Linux cluster with two AMD Opteron 2.4 GHz CPU's, Quadro fx4500 GPU's and 8GB RAM per node, connected using Gigabit ethernet. Each node was driving a display with a resolution of 1280x1024.

The benchmark application, eqPly, renders static polygonal data using display lists, organized in a 3D kD-tree for efficient view frustum culling. The data set for the benchmark was the David statue, consisting of 56.2M triangles. The same executable, view frustra, data set and hardware was used for both test runs.

The Chromium setup uses the tile-sort SPU to stream the OpenGL command produced by a single application instance to the display nodes. The Equalizer setup uses one application instance to drive the rendering clients, which are deployed locally on each node.

### Results

Running the application using Equalizer provides truly scalable rendering performance. The frame-rate increases to 5 frames per second with the number of

displays, since each render client submit less OpenGL commands due to view frustum culling.

Using Chromium provides a simple way to run applications, since no code changes are required. The performance increases slightly to 1.65 FPS for four nodes due to the frustum culling performed for each display by Chromium. With a higher number of displays, the OpenGL stream processing and network transmission becomes the bottleneck and the performance decreases.

### About

The benchmark was conducted by the [Visualization and MultiMedia Lab](#) at the Department of Informatics, University of Zürich.

Equalizer is a product of Eyescale Software GmbH. Please visit [www.eyescale.ch](http://www.eyescale.ch) and [www.equalizergraphics.com](http://www.equalizergraphics.com) for more information.

