

**Title of Your Paper/Talk**

Issues In Introducing Micro-programmable Graphics Hardware Into the Animated Production Process

**First name and Last name**

Yahya H. Mirza

**University of**

Not Applicable

**Department of**

Not Applicable

**Affiliation**

Aurora Borealis Software

**ZIP code & City**

Redmond, Washington 98052

**Country**

USA

**e-mail:**

[yahya\\_mirza@hotmail.com](mailto:yahya_mirza@hotmail.com)

**Website**

Not Applicable

**Keywords:**

Cinematic Productions, Micro-programmable graphics hardware, DX9, High Level Shading Language, Shader Effects, Procedural Rendering, Open GL2.0, Open GL Shading Language, C# and .NET.

**Abstract:**

Today, a great opportunity is becoming available for content creators interested in near cinematic quality interactive 3D rendering. With the recent introduction of floating point calculations in graphics hardware, the functionality required to approach the capabilities of Pixar's Renderman shading language in real time are just now starting to become achievable. The mechanism being used to attain this goal is multi-pass rendering techniques enabled through compilation technology. A key player in this arena is Microsoft with their DirectX 9.0 API, which for the first time introduces many new real time capabilities which were once exclusively in the realm of high-end offline software renderers.

This paper will begin by taking the perspective of a computer animated production company interested in introducing micro-programmable graphics hardware into the animated production process. The aim will be to take a pragmatic view of how micro-programmable graphics hardware can actually be used practically, what some of the current limitations are, and what needs to be addressed to make micro-programmable solutions truly useful in a animated production setting. Throughout the paper concrete production scenarios will be used to illustrate the technical issues involved.

The paper will continue by comparing the various micro-programmable graphics hardware available today including the Sony Playstation2 as well as technologies available on the PC platform. The aim will to explain the key technical issues involved in the micro-programmable approach to graphics. I will then illustrate how the DirectX 8 API to a certain extent “virtualized”, these graphics hardware solutions, through a standard graphics instruction set. Tradeoffs between the DirectX execution-engine will be made with the newly proposed OpenGL2 architecture.

Furthermore I plan to illustrate the technical issues involved in targeting high-level graphics languages to these various micro-programmable graphics architectures. The technical issues differentiating these various shading languages will be explored. Finally, I will show through a concrete example how the C# programming language can be used to abstract a shader component which itself is implemented using Microsoft’s new High Level Shading Language (HLSL) and Effect (FX) file format.