Volumetric Effects in a Snap

Joseph Hegarty Denis Teplyashin Peter Georges Daniel Heckenberg
Animal Logic*

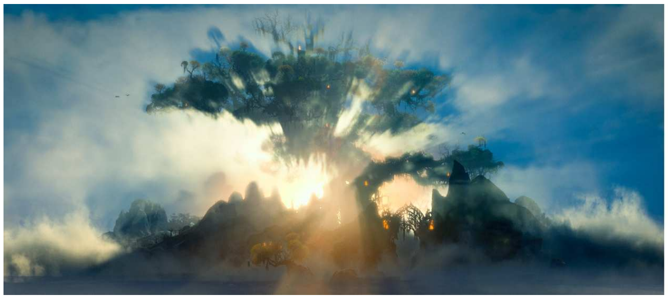


Figure 1: *Atmospherics and fluid simulation were used extensively in* LotG. *©Warner Bros. All rights reserved.*

Abstract

Volumetric effects have become a mainstay of the visual effects community over the past decade. With many third-party solutions available, what are the benefits and challenges of creating a proprietary system? We outline *Snap*, Animal Logic's grid-based simulation and volumetric rendering framework used in *Legend of the Guardians: The Owls of Ga'Hoole (LotG)* and *Sucker Punch (SP)*.

1 Production Motivation and Experiences

When assessing the effects requirements for *LotG* it was clear that existing in-house and third-party fluid simulation systems would not meet the scale and turnaround requirements for a number of important sequences, such as the climactic bush fire sequence (see fig. 1). Additionally, as the sky would be the stage for much of the film's action, efficient, directable procedural atmospheric effects would be required.

Development began with the relatively simple problem of creating procedural volume modeling tools for cloud and fog effects along with the necessary preview and rendering infrastructure. *Snap* was then augmented with disk-based caching and a modular set of algorithms for large-scale grid-based fluid simulation.

2 Overview

Snap is a plugin for *ALF*, Animal Logic's proprietary multi-threaded node-based computation framework. *ALF* can be used standalone or embedded inside *Maya* (among other environments). *Snap* simulation rigs may be created using the *Maya* Hypergraph, with realtime feedback through the *Maya* viewport.

Production quality simulations may be executed locally or on the render farm, distributed across any number of machines while maintaining a unified solve. Simulation caches and procedural rigs can then be rendered using PRMan (another environment where *ALF* can be embedded).

3 Benefits

The node-based nature of *Snap*, compared to a monolithic black-box solution, allows a great deal of flexibility when creating rigs. Many production requests were fulfilled simply with a combination of existing nodes, without requiring additional development.

Having complete control over the code-base allows for rapid turn-around of bug fixes and new features, e.g. support for wavelet turbulence [Kim et al. 2008] was added through the addition of a single new node in combination with several existing nodes.

There are a number of third-party libraries available which can greatly accelerate the process of creating a distributed grid-based simulation framework. For *LotG* and *SP* the distributed simulations made use of *PETSc* [Balay et al. 2011] to manage the decomposition of the simulation and perform the incompressible solve.

4 Challenges

Computational Fluid Dynamics, including the subset of techniques normally used to produce computer generated imagery, require considerable domain specific knowledge. Also, while creating a simple fluid simulation prototype is relatively straight-forward, creating a production ready solution which provides the user with enough creative tools takes time. If *ALF* did not already exist, it is questionable if the decision to create *Snap* would have been made.

5 Next Steps

Our experience with *Snap* in its first set of productions has been that the ability to modify and extend any and all aspects of the framework is invaluable. Integration with other *ALF*-based tools is an area of active development, e.g. *Snap* interoperability with *Nexus*, our particle-based simulation framework. Only by having complete ownership of a code-base are such opportunities available.

References

BALAY, S., BROWN, J., BUSCHELMAN, K., GROPP, W. D., KAUSHIK, D., KNEPLEY, M. G., MCINNES, L. C., SMITH, B. F., AND ZHANG, H., 2011. PETSc Web page. http://www.mcs.anl.gov/petsc.

KIM, T., THÜREY, N., JAMES, D., AND GROSS, M. 2008. Wavelet turbulence for fluid simulation. *ACM Trans. Graph. 27* (August), 50:1–50:6.

*e-mail: {joeyh,denist,peterg,danielh}@al.com.au