

Lighting Pipeline for One - Or How to Keep Sane in a Discworld

Björn Siegert
Troll Bridge
Wellington, New Zealand
bj.siegert@gmail.com



Figure 1: Two shots from the film *Troll Bridge*. There are over 500 VFX shots in this movie and it was entirely crowdfunded

ABSTRACT

Troll Bridge is a crowdfunded live-action short film more than 15 years in the making and involving the work of more than 400 volunteers across the world. In order to light and render over 500 shots it was necessary to build a workflow driven pipeline with minimal development overhead. In this talk we discuss how we brought the full CG environment of the bridge and the talking horse character to life by designing a lighting and rendering pipeline tailored to the needs of an independent working artist. We give insights into the challenges we encountered and how we kept our render budget low while still delivering feature film quality.

CCS CONCEPTS

• Computing methodologies → Computer graphics;

KEYWORDS

pipeline design, independent film making, lighting, rendering

ACM Reference Format:

Björn Siegert. 2018. Lighting Pipeline for One - Or How to Keep Sane in a Discworld. In *Proceedings of SIGGRAPH '18 Talks*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3214745.3214759>

1 INTRODUCTION

With over 500 FX shots ranging from CG characters over crowd shots to full CG environments, *Troll Bridge*[tro 2018] is an ambitious

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGGRAPH '18 Talks, August 12-16, 2018, Vancouver, BC, Canada

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5820-0/18/08.

<https://doi.org/10.1145/3214745.3214759>

project. However, in contrast to big movie productions there is no budget and the project lives through the work of the many volunteers, bringing unique challenges as well as opportunities. While modern movie pipelines are built on the design goal to manage big work loads from a coordinated team, flexibility and agility is the dominant factor when working in a single-person-team.

2 PRODUCTION SET-UP AND CHALLENGES

Working on multiple operating systems with no face-to-face communication was a major challenge in this project. This meant we had to remain flexible and encourage artists to proactively solve issues.

To coordinate such an endeavour we relied heavily on Shotgun so each artist knew what task he had to do and what notes needed to be addressed regardless of his timezone. For general questions we used a Skype group chat, which minimized communication overhead for all participants.

Keeping project files up to date and in one structure required relying on DropBox as a unified project server because it is easy to use, platform independent, scaleable and has a built-in versioning system. Over the project we accumulated over 3TB of data. Since space was precious we took care that we only produced data we needed.

3 BUILDING A PIPELINE FOR ONE

Two major tasks in the production of this Discworld adventure were the lighting and rendering of the bridge environment and the talking horse character. Here we knew that we needed to layout a pipeline that helped the artist but was easy to maintain and had no development overhead. Since there was no R&D or pipeline team in this project we needed to write the tools during the job and figure out which steps would benefit from full or partial automation to help the artist. The design of the tools previous to the start of

the task ranged from several days to a week with every script being between 200 and 500 lines of code. All scripts were also managed through a repository on Github. The tools were written in Python. Maya scripts used the Pymel API for its ease of use. Although automatisaton is normally associated with reducing time for specific tasks, we noticed that its bigger benefit is actually the reduction of user errors and this leads directly to workflow speed ups.

3.1 The Bridge

For the CG bridge environment this meant that a simple light rig was built which could be shared and modified if needed on a shot to shot basis. Lighting and rendering was all done in Modo and we decided to use its existing tools to set up the rig. The initial light rig was managed as a separate scene file and shared between two artists. When lighting a shot it was then imported, adjusted and modified per shot. We decided against an automated rig transfer because no across-the-shot updates were necessary for this simple setup.

Only a few scripts were developed which made rendering previews and shotsubs easier for the artists.

3.2 The Horse

The horse character was a different challenge. Not only was it a talking and screen-filling horse head, it was necessary find a standard structure between artists for modelling, grooming and animation.

To make scene setup for lighting as easy as possible we automated the generation of animation caches and created a simple, but effective tool box to create a scene for lighting. Various solutions for animation caches were tested. We decided against Maya's built-in geometry cache because of compatibility issues with the cloud based render farm GarageFarm[gar 2018]. In the end we opted for Alembic because of its good integration, its robustness and small file size.

Animation rig and the horse head asset were always kept seperated so that artists only had to re-import a new animation cache for updates. This process was automated, but this also meant that names in the scene tree had to stay the same accross the lifetime of the project.

To keep shots up to date with look development changes the horse was referenced into the lighting scenes. This way updates could be adopted across shots without further user overhead.

To keep the rendering budget on GarageFarm[gar 2018] to a minimum we needed to find a sweet spot for shading and lighting complexity. We chose the Arnold renderer because it has good integration with the hair plugin Yeti, which we used for hair and fur as it was perfect for the outdoor scenes. A simple light set-up was key for effective lighting. Surprisingly, we managed to match the plate with only three area lights including one IBL for all the shots.

During handover to the lighting artist we discovered that Yeti had various issues switching operating systems. We decided against developing an update process because grooming had almost finished and we could leave the remaining work for shading and rendering with one artist.

4 CONCLUSION

Big projects do not have to be complex and sometimes the simplest solution right next to you is the best one. By spending more time at the start of a project, many problems encountered at a later stage could be solved upfront. Even in small and independent films it pays off to evaluate plugins and tools. The main focus should not only be their price, but also how big their userbase is. Once you rely on freelancers, exotic tools might give you a bigger benefit, but it can be hard to find artists who have knowledge of those tools. The ability to write code is important for everyone, not only for software developers, because automatisaton is not only done to reduce set-up time, but also to reduce user errors which lead to general time savings.

ACKNOWLEDGMENTS

We would like to thank Daniel Knight and Ahren Morris who made this project possible. Special thanks goes to to Andrea Weidlich for her technical expertise and moral support, Christian Bloch and all the amazing artists who dedicated their time, knowhow and knowledge to bring one of Terry Pratchett's stories to life.

REFERENCES

2018. GarageFarm. (2018). Retrieved Februray 11, 2018 from <https://garagefarm.net>
 2018. Troll Bridge. (2018). Retrieved Februray 11, 2018 from <http://www.trollbridge.film>