

Editorial Pipeline Conversion: Animal Logic’s Transition to OpenTimelineIO

Tim Lehr
Animal Logic
Vancouver, BC, Canada
tim.lehr@animallogic.ca

Barish Balachandran
Animal Logic
Sydney, NSW, Australia
barishb@al.com.au

Oliver Dunn
Animal Logic
Vancouver, BC, Canada
oliver.dunn@animallogic.ca

Nathan Lacey
Animal Logic
Sydney, NSW, Australia
nathan.lacey@al.com.au

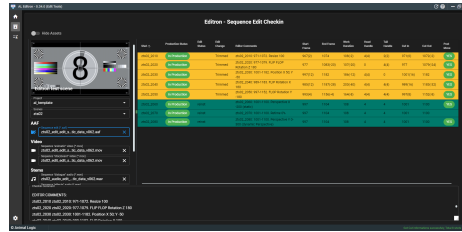


Figure 1: Editron user interface. Built using Electron.js and OpenTimelineIO.

ABSTRACT

We introduce Animal Logic’s editorial pipeline refactor from a rigid and overly complex in-house solution, towards a more modern, flexible approach, based on the open source technologies *OpenTimelineIO* and *Electron.js*. This upgraded design greatly increases flexibility over the previous effort, enabling cross-platform user adoption and further decoupling our tools from the editorial software of choice. The new pipeline is now rolled out onto our most recent productions and we are already starting to see the benefits of its extensibility and ease of troubleshooting.

CCS CONCEPTS

• Computing methodologies → Computer graphics; Editorial.

KEYWORDS

Editorial OpenTimelineIO Electron Pipeline

ACM Reference Format:

Tim Lehr, Barish Balachandran, Oliver Dunn, and Nathan Lacey. 2022. Editorial Pipeline Conversion: Animal Logic’s Transition to OpenTimelineIO. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Talks (SIGGRAPH ’22 Talks)*, August 07-11, 2022. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3532836.3536278>

1 INTRODUCTION

While Animal Logic’s pipeline has gone through several major refactors in recent years, namely its conversion over to a fully Pixar™

Universal Scene Description (USD) based solution, due to several limiting factors, the editorial pipeline had not. With the advancements of *OpenTimelineIO (OTIO)* (<https://github.com/PixarAnimationStudios/OpenTimelineIO>), Animal Logic’s growth, and with it a pressing need to upgrade our code base to Python 3, it was time for a redesign of our editorial tools.

There were two main requirements to be considered with this refactor: The user interface for editorial needed to be cross-platform; developers and TDs would be using the tools on Linux, while the editorial department would be running either Windows or Mac. The second main requirement was to reduce the large amount of in-house custom Python code and remove dependencies on any particular editorial file format, allowing us to greatly improve our agility and flexibility in this area going forward.

2 USER INTERFACE

After investigating multiple frameworks and deployment technologies, the decision made was to develop a native application. A modern web frontend framework (Vue.js) in combination with Electron.js was chosen for this task, giving us the ability to develop the application once and deploy it across multiple platforms. *Editron* (frontend application) takes media files from the editorial application, converts the cut data to OTIO and compares it against previous versions. Finally Editron packages the files and sends them to the backend for publishing. Shown in Figure1.

2.1 Evaluation Options

There were several things to consider for the user interface. Web vs native, in this instance native was chosen to allow for speed of processing the large files the users would be uploading. Web applications lack the native file-system support that was required here. Electron vs. Qt/PySide-2, in this instance Electron was chosen for its portability to other platforms. While Qt is known territory for developers at Animal Logic, internal build systems were not designed for cross platform and upgrading to Electron was the preferred path.

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 ’22 Talks, August 07-11, 2022, Vancouver, BC, Canada
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9371-3/22/08.
<https://doi.org/10.1145/3532836.3536278>

