A Portal for Managing Reviews and Beyond

Danny Rerucha
Blue Sky Studios

Tommy Zhu
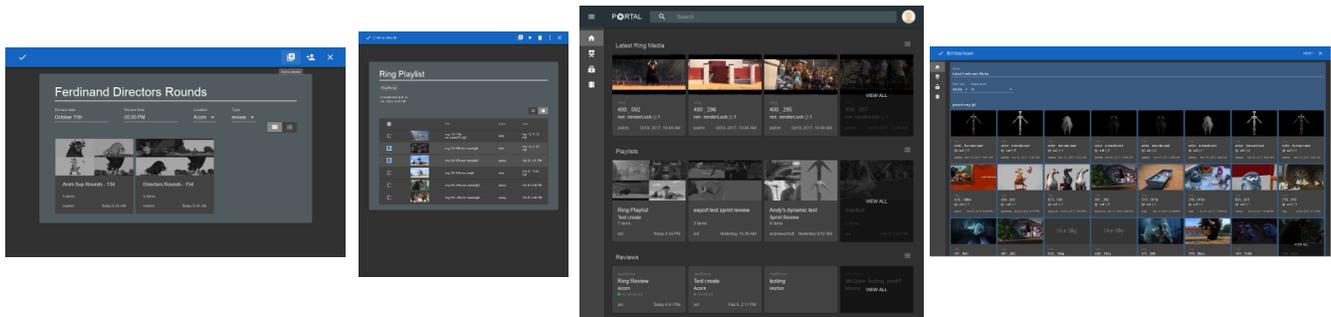Blue Sky Studios

Andy Schott
Blue Sky Studios

Marley Gilb
Blue Sky Studios

Tracy Priest
Blue Sky Studios

Jennifer Brola
Blue Sky Studios

Blessan Abraham
Blue Sky Studios

Mark McGuire
Blue Sky Studios

ABSTRACT

We present Portal, a modern web interface for creating and managing media and review sessions at Blue Sky Studios. Utilizing a horizontally scalable stack, Portal allows artists and production management to quickly search for and play back media, capture notes with draw-overs, and manage review sessions all from within a web browser. To help ensure success, our media tools team conducted rigorous user story mapping sessions with key management staff across the studio. As a result, Portal has become an integral part of the dailies workflow at Blue Sky.

CCS CONCEPTS

• **General and reference** → **Design**; • **Social and professional topics** → Systems planning; • **Computing methodologies** → Graphics systems and interfaces; • **Software and its engineering** → *Collaboration in software development*.

KEYWORDS

animation, media, microservices, reviews, open source, playback

1 INTRODUCTION

A critical component of creating a successful animated film is an efficient and effective review process where supervisors and directors directly collaborate with artists to improve the film. For years, Blue Sky's review process and the toolset supporting reviews has been department-centric and cumbersome. After the initial deployment of Blue Sky's next generation pipeline and services stack, Conduit, we had the opportunity to rethink how we approached media and reviews at the studio. To meet the needs of a variety of departments, we employed story mapping sessions with key representatives across the studio. The media tools team walked through each coordinator, supervisor, and director's day as it pertained to reviewing media to find inefficiencies and areas where a technology solution could help. We called the resulting web interface, Portal, and along with a new "review early and often" mantra, it has revolutionized how everyone from artist to director finds and reviews material.

2 CHALLENGES WITH EXISTING WORKFLOWS

During the story mapping process we quickly learned that no two departments reviewed media the same way. Even with differing workflows, however, a set of common problems began to emerge. Existing workflows required large amounts of prep time inside of screening rooms, which was being used to hunt for imagery,

construct playlists, troubleshoot hardware, and communicate with artists. For the people driving review sessions, identifying and locating imagery that was ready for review was a time consuming process, and trying to find images on-the-fly that hadn't been prepared ahead of time was a constant source of frustration.

A shared media submission framework did exist to register media with our pipeline, but not every department was using it. As a result, images were getting shown to directors during formal reviews that our pipeline wasn't tracking. We didn't know what fixes or changes went into creating these untracked images, and later, it was impossible to verify that the images on disc were the same images a director had approved. Even when departments were properly registering their media with our pipeline, our old media database was slow to search and wasn't integrated with any kind of playback tool.

3 PORTAL

We chose to build a web-based interface to address these problems because we wanted to make media accessible to everyone regardless of what hardware they were using. We chose React for the front-end for its simplicity and scalability and because its component-based architecture made it easy to build a library of reusable web components. Thanks to modern web technologies, Portal can run on pretty much every device we have in-studio that has a web browser. Another benefit of building Portal for the web is that we can embed pieces of the Portal interface directly into existing web-based tools like Shotgun and Confluence, which means users can reap the benefits of the new review paradigm without having to leave familiar ecosystems behind. In addition to being accessible, Portal also needed to be performant. Conduit, Blue Sky's new pipeline, provided the solution. Conduit is a microservices-based pipeline composed of Java/Spring services with Cassandra and Elasticsearch data stores. With Conduit, we had a set of services that was capable of searching through hundreds of thousands of records in fractions of a second. This speed opened the door to new ways of working.

Searching for media in Portal is simple thanks to the Lucene query language. Users can search by project, sequence, shot, asset, user, date, and any number of arbitrary tags that users attach to their media items. Users can even search via partial matches on description fields. Results are near instantaneous, and typos are handled gracefully thanks to fuzzy string matching. Our updated media submission process automatically generates web-friendly H.264 encoded movie representations for every image sequence that gets submitted, so after a user finds media in Portal, they can stream it right in the browser via the Portal Media Player (PMP). The PMP connects to a custom built Media Streaming Service written in Rust, which was chosen for its high performance and because the memory safety guarantees that Rust provides ensure that this production-critical service remains robust and secure. With the PMP, multiple clips can be played back on a shared timeline to check continuity, and individual clips can be scrubbed frame-by-frame. Future plans for the Media Streaming Service include direct streaming of on-the-fly transcoded EXR image sequences.

Portal is more than just a media browser though. It's also a fully featured playlist and review management system. With Portal, users can create rules-based dynamic playlists that automatically curate media based on user-defined filter sets. Users can also schedule reviews and book screening rooms via integration with the Google Calendar API, setup recurring reviews to minimize prep time for meetings that repeat, capture notes and annotations, and create customizable dashboard pages to highlight the content that individual users care about the most.

4 BSSPLAY

Portal provides a complete review workflow in a web browser, but sometimes the web interface isn't sufficient. For instance, if a user needs to approve high resolution or color-correct imagery, then viewing compressed videos optimized for web streaming won't cut it. That's where bssPlay comes in. bssPlay is a Python/Qt desktop application built directly into the RV playback tool. It leverages the same Conduit infrastructure as Portal, so it provides the same robust search interface for browsing media, playlists, and reviews. bssPlay and Portal are tightly integrated, so playlist edits in one interface are synchronized between the two applications. bssPlay is also tightly integrated with RV, so users have extensive control over playback when they need it. With bssPlay users can import and export playlists, side-by-side and A/B compare images, add continuity to view shots in context, swap between image types (story, layout, animation, etc.), versions, and levels of detail (EXR, JPG, etc.), and submit unregistered media from RV directly to Conduit.

5 CONCLUSION

Both Portal and bssPlay have become integral to the dailies workflow at Blue Sky. While the initial deployment of Portal was developed to manage media and review sessions, we intend for the Portal web interface to ultimately become the primary interface for both artists and production management to interact with all products in Conduit. And similar to the development of Conduit, we intend to share both the concepts and the code with the industry.