sgpublish Documentation

Release 0.1

Western X

October 05, 2015

Contents

CHAPTER 1

Overview

This package is a toolkit for creating work exporters, or “publishers”, which report the work to Shotgun, and ease the flow of work products (e.g. caches, cameras, renders, etc.) from one department to the next.

The basic idea is for the artist’s tools to know where the products are so that the artist doesn’t have to dig through the file system looking for caches they need, and for the artist to be automatically warned whenever something they are using in the scene/script becomes out of date.

To streamline the process for artists, the tools may also create a Shotgun `Version` of the current shot for review, create QuickTimes for editorial, and log their time worked.

Streams

Publishes are grouped into streams, consisting of Shotgun `PublishEvent` entities sharing the same Shotgun task, type (e.g. `"maya_geocache"`) and name. Any publish attached to the same task, of the same type, and with the same name are considered incremental versions of the same thing.

While publishes have version numbers, and these version numbers can be automatically generated to be increasing, there is no meaning behind the version numbers except what the users/artists put upon them.

Note: Even if they are on the same shot, a "rigging" stream is seperate from a "animation" stream is seperate from a "lighting" stream, and so the version numbers do not line up across departments.

For example, imagine that we are working on lighting a shot. We have version 1 of the lighting, which is using version 3 of the animation, which is using version 5 of the rig. If an animation revision is required, it will bump up to version 4. Lighting will then upgrade to using the newer animation publish, and will bump to version 2.

CHAPTER 3

Example Publishers

The following publisher/importer pairs exist within the WesternX pipeline (and most of which are either availible in this package or in the mayatools):

- generic Maya scenes;
- Maya geocaches;
- Maya cameras (also to Nuke);
- Maya reference edits;
- Maya transformations (as locators);
- Maya render layers.

CHAPTER 4

Contents

4.1 Publishing via a Publisher

4.1.1 Creating a Publish

Publishing is assisted via a `Publisher` instance, which manages the creation of the folder for the publish (via SGFS templates), copying of files into that folder, permissions, and creating the final entity on Shotgun itself.

The added complication is so that we can always be sure that if a publish is on Shotgun with a non-zero version number, we know that the export finished and the data is in a valid state.

We use the publisher as a context manager around our export code:

```
>>> task = sg.find('Task', filters_to_get_a_task)
>>> with Publisher(link=task, type="your_publish_type", code="Demo") as pub:
...     # Export into pub.directory
```

4.1.2 API Reference

4.2 Import and Export

4.3 Utilities

CHAPTER 5

Indices and tables

- genindex
- modindex
- search