

# RVGun

## Installation & Usage

Marco Romeo ([marco-r@my.studionestbarcelona.com](mailto:marco-r@my.studionestbarcelona.com))

### Installing RVGun

To install RVGun in RV, please copy the *RVGun.zip* file into the RV's packages folder (*rv\_install\_dir/plugins/packages*) and run **RV**.

Once **RV** has started, open to the preferences menu and go to the packages tab. In the list you should see **RVGun** (figure 1). Turn on the *installed* checkbox.

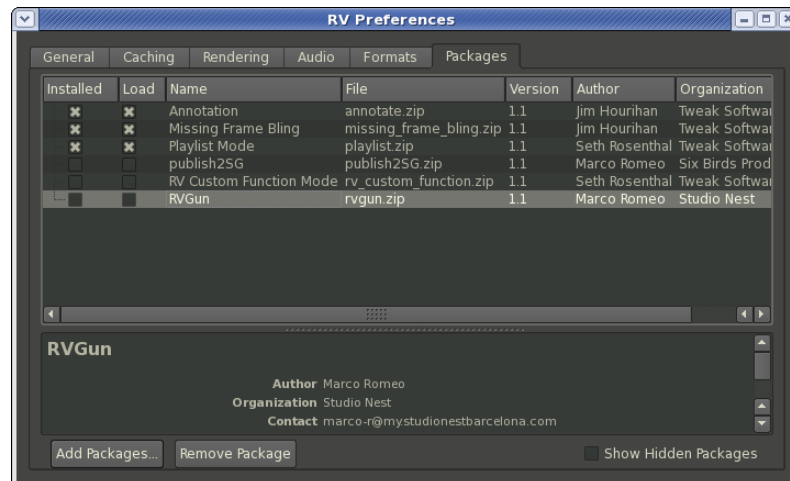


Figure 1. available packages list into RV's preferences window

RVGun is both a plugin for RV and a script operating on the Shotgun server. For that reason it is necessary to create a script entity in Shotgun that will be used to bind the RVGun python scripts.

Please refer to [link](#) for help in creating script entities in Shotgun.

Once the script entity is created, copy and paste (figure 2) the server url, script name and application key from Shotgun to the RVGun configuration file (*rv\_install\_dir/plugins/SupportFiles/RVGun/rvgun.cfg*).

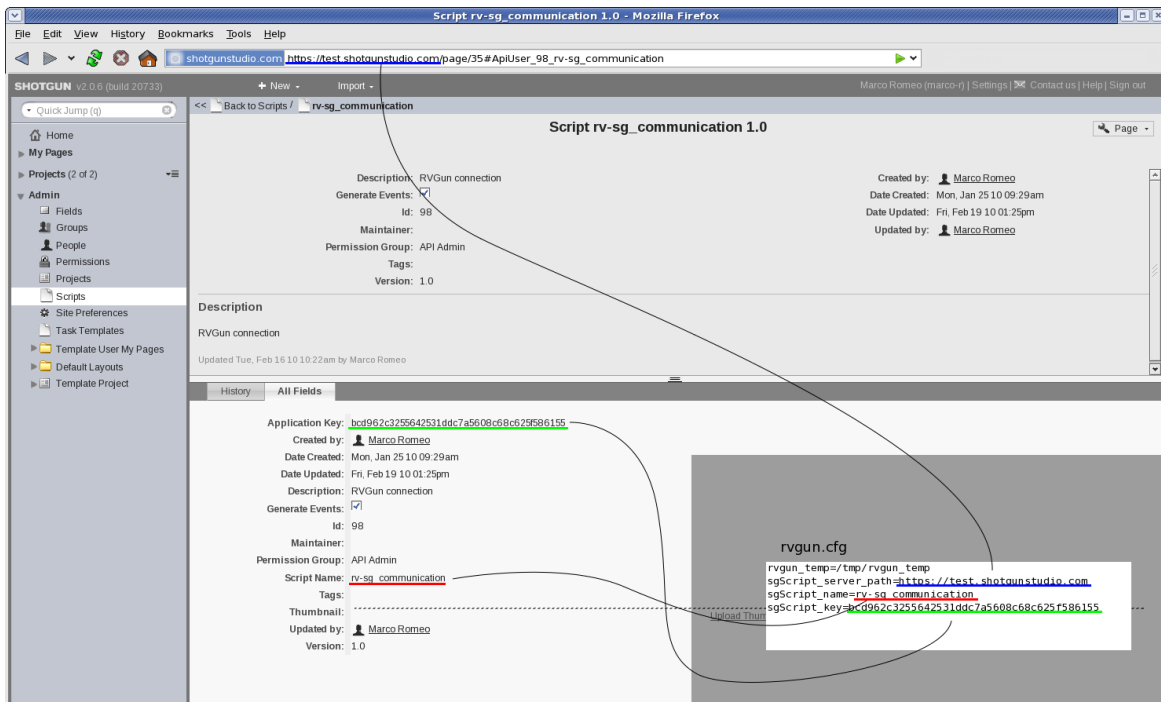


Figure 2. copy-pasting data from script entity to rvgun.cfg

If everything is fine, you should be able to access RVGun from RV's *tools* drop-down menu or pressing **F11**.

Now it is necessary to configure the temporary folder and the users mapping.

The temporary folder is the place RVGun will use to store the RV session and generate the thumbnail and quicktime video to upload.

The temp folder must be specified in the config file

(*rv\_install\_dir/plugins/SupportFiles/RVGun/rvgun.cfg*) filling in the *rvgun\_temp* field. RVGun will take care of creating the folder if it doesn't already exist.

The users mapping is necessary as RVGun heavily relies on filtering information depending on which user is using it. While a good approach would be to match the system user names to the shotgun user profiles, it doesn't happen always. For that reason, the mapping will allow RVGun to understand which shotgun user it has to look for depending on the system user. As an example, imagine we have a linux user with id *john\_smith*, while the same user is known in Shotgun as *John-S*. Including a row in the *users.txt* file (*rv\_install\_dir/plugins/SupportFiles/RVGun/users.txt*) specifying *john\_smith=John-S* will let RVGun understand that whenever user *john\_smith* uses RVGun, it has to look for *John-S* in Shotgun.

If the user names for the OS and Shotgun are the same, just leave the users file as it is. RVGun will remap users specified in the file only and use the system's user name for the others.

## Using RVGun

Open the image or image sequence you want to use as version instance in Shotgun and open RVGun through the *Tools* dropdown menu in RV (Figure 3) or by using the *F11* shortcut. Once RVGun opens, it is possible to choose between several attributes to publish your version on shotgun (Figure 4).

It is important to keep in mind that RVGun is connecting to the shotgun data through an internet connection, meaning that reading and writing data will take a while, so don't worry if it takes few seconds to respond to your interaction.

Moreover, if the internet connection is not working, the system will not be able to do anything.

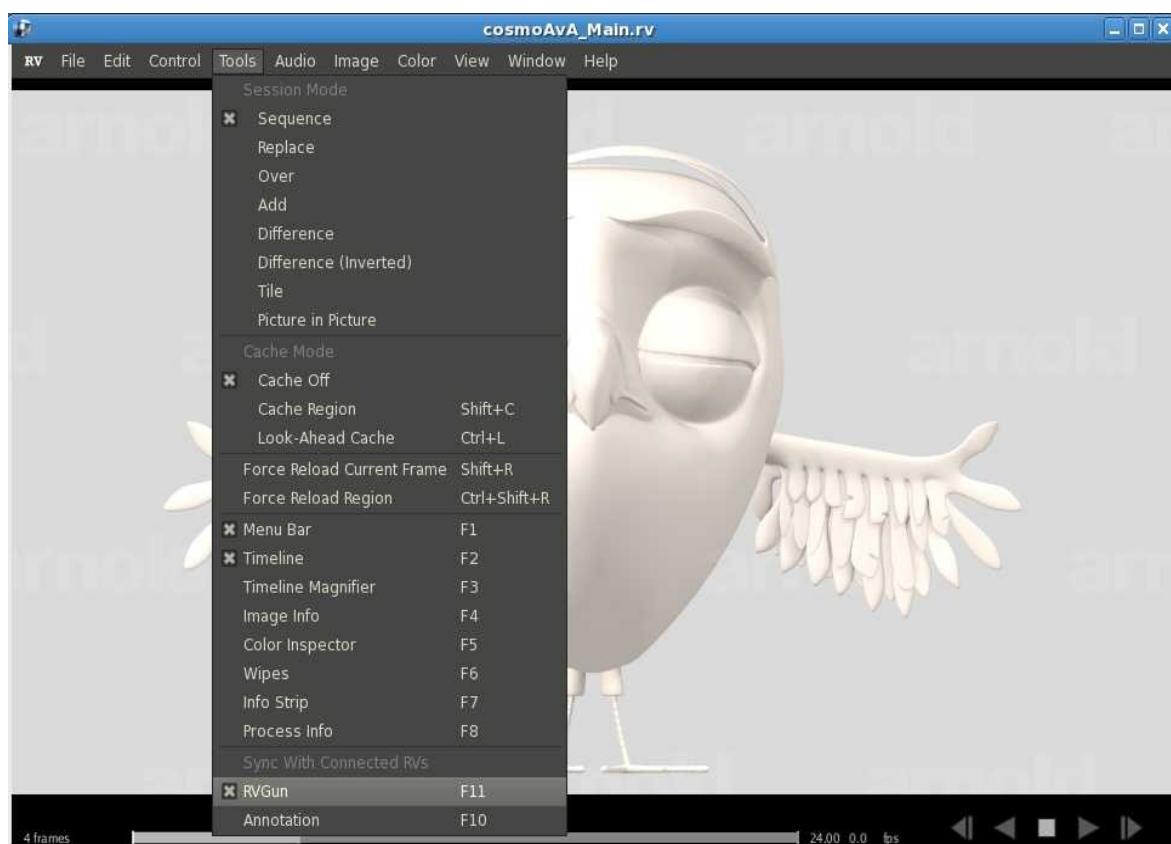


Figure 3. RVGun menu item

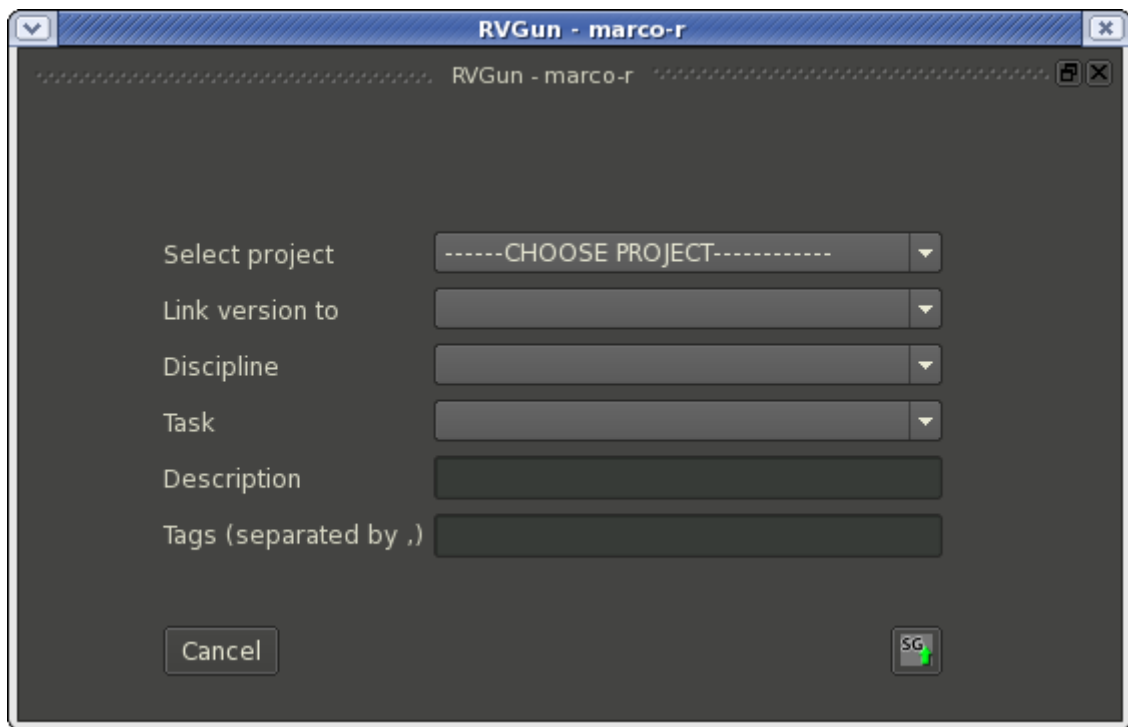
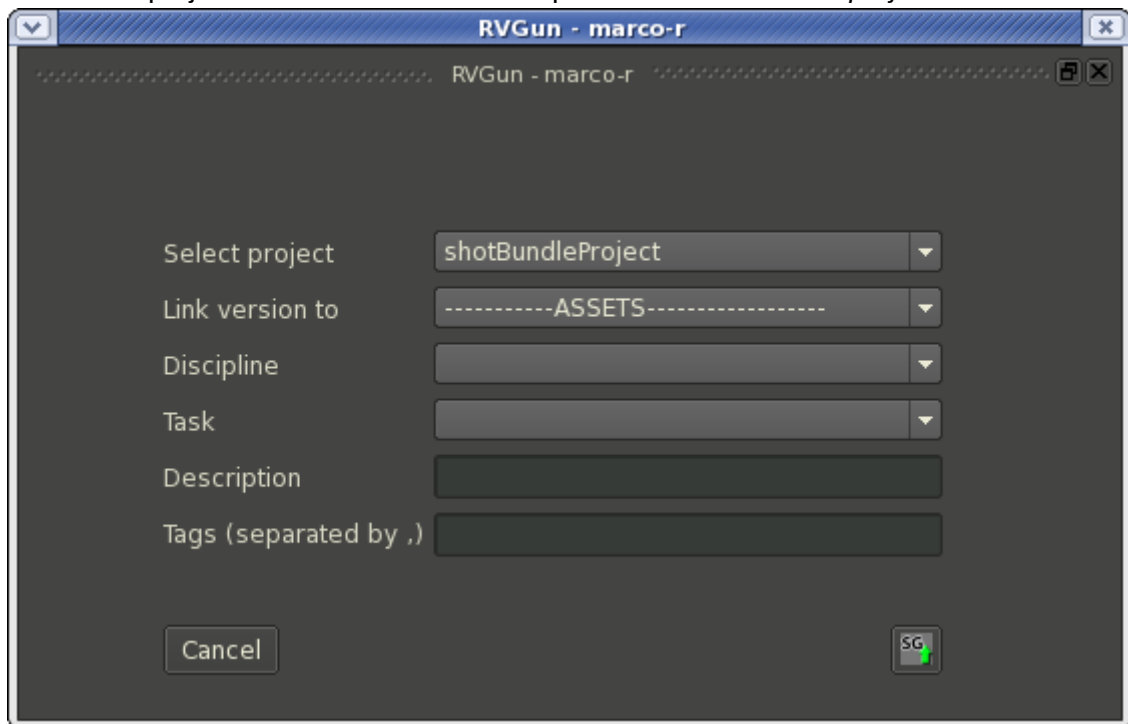
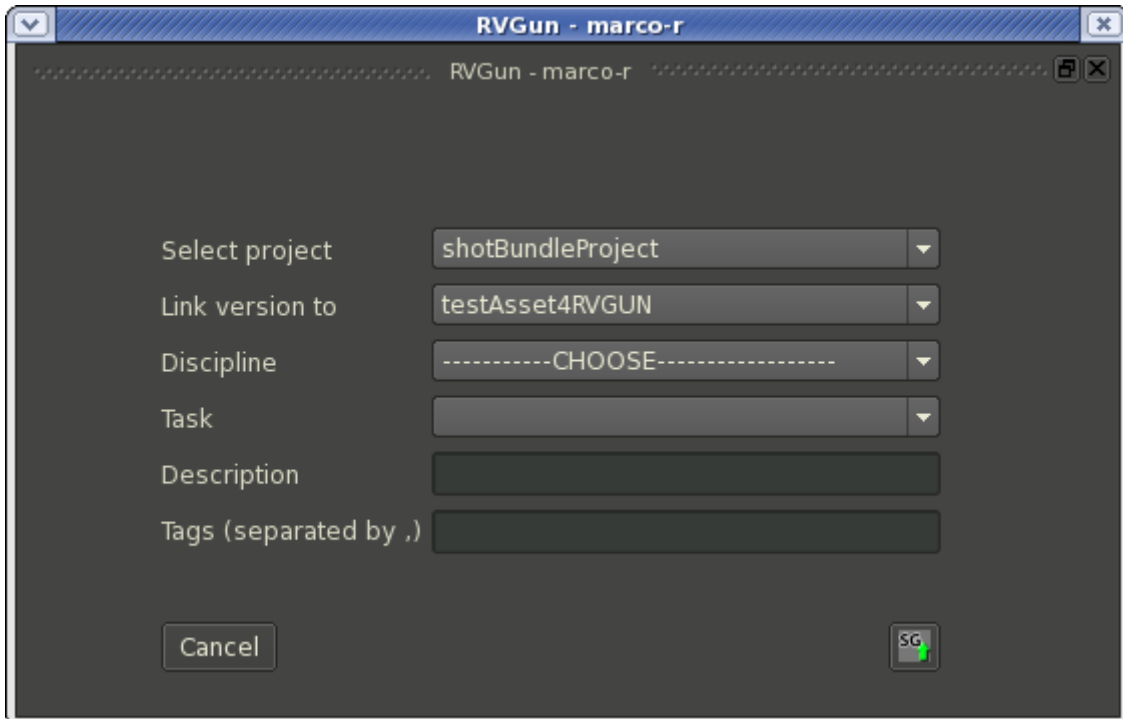


Figure 4. RVGun user interface

> Choose the project to which the version will pertain from the *Select project* list.



> Then choose the data you want to link the version to. In the *Link version to* list there is a list of assets or shots.



RVGun - marco-r

Select project: shotBundleProject

Link version to: testAsset4RVGUN

Discipline: -----CHOOSE-----

Task:

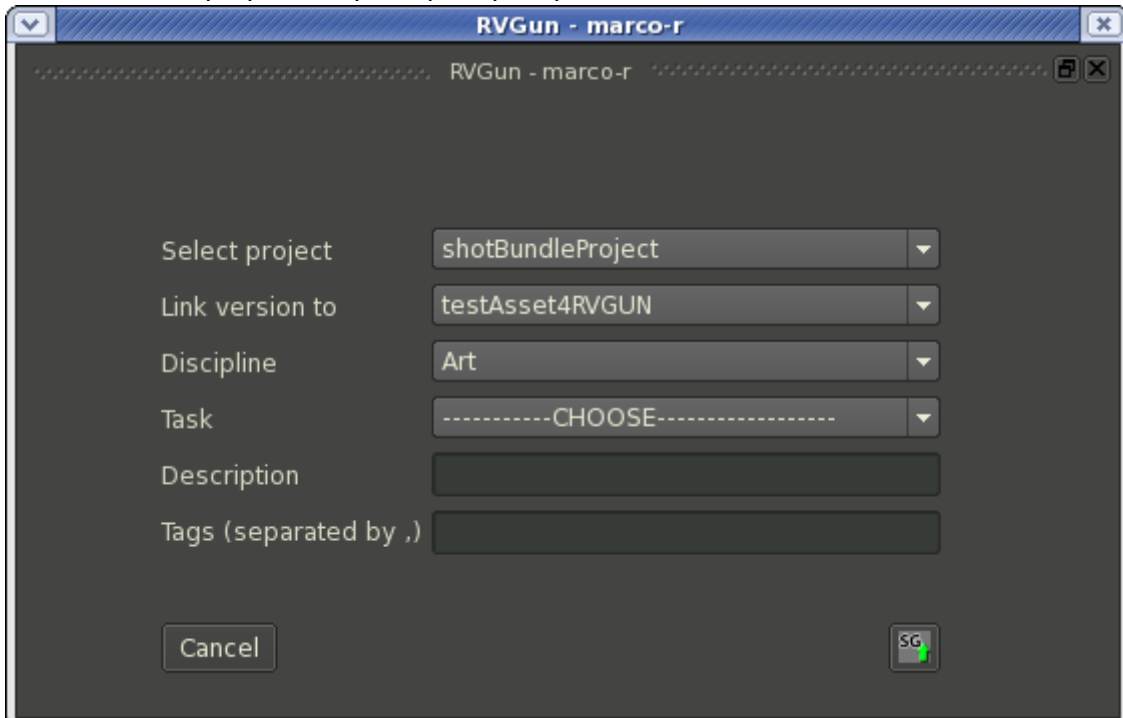
Description:

Tags (separated by ,):

Cancel

SG

> Now choose the proper discipline (*Discipline*).



RVGun - marco-r

Select project: shotBundleProject

Link version to: testAsset4RVGUN

Discipline: Art

Task: -----CHOOSE-----

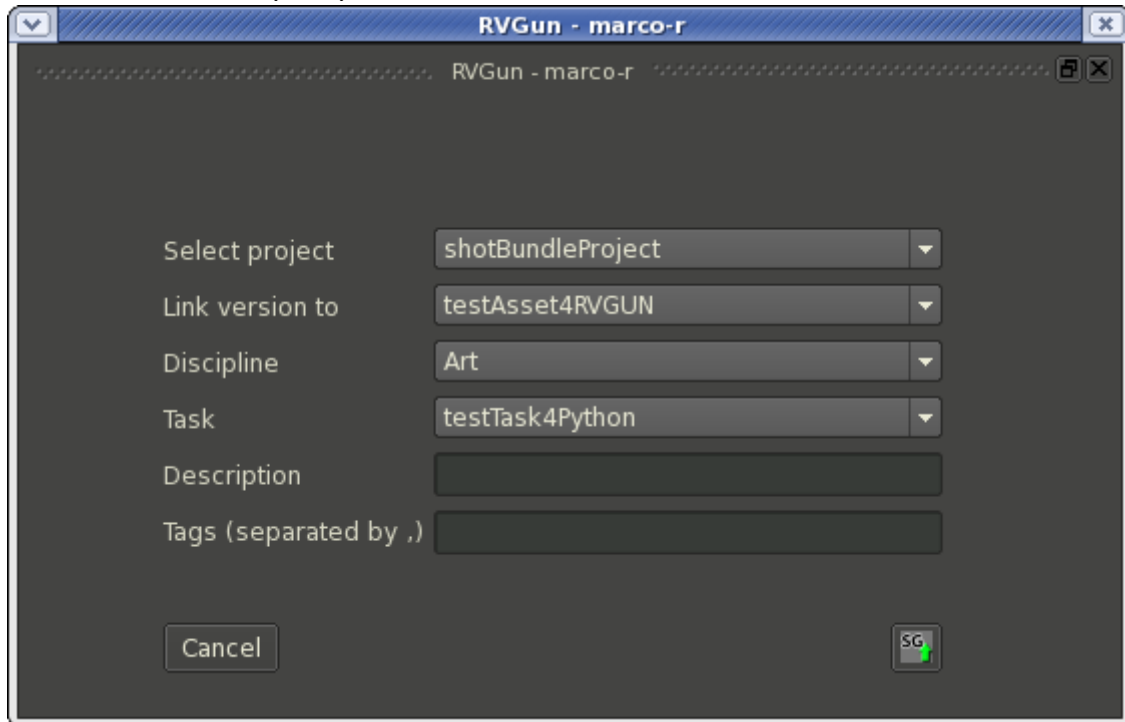
Description:

Tags (separated by ,):

Cancel

SG

> Now choose the task (*Task*).



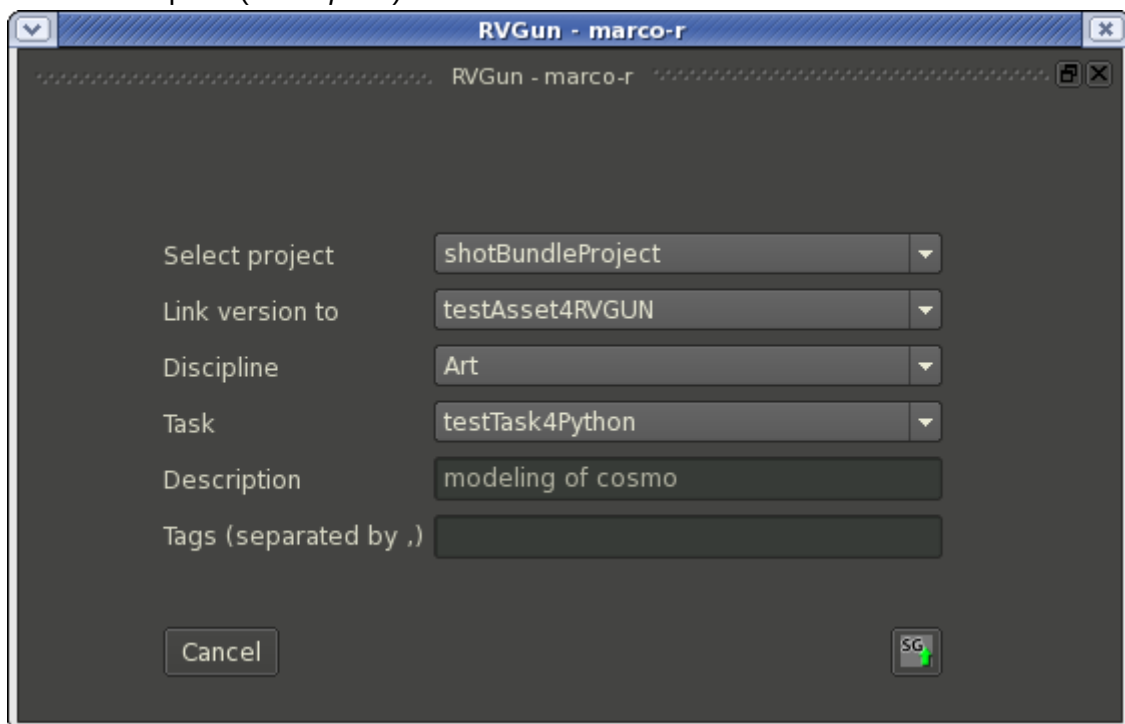
The screenshot shows a dialog box titled "RVGun - marco-r". It contains several dropdown menus and text input fields. The "Task" dropdown menu is selected, showing "testTask4Python". The other fields are: "Select project" (shotBundleProject), "Link version to" (testAsset4RVGUN), "Discipline" (Art), "Description" (empty), and "Tags (separated by ,)" (empty). There is a "Cancel" button at the bottom left and a small "SG" icon with a green arrow at the bottom right.

Select project	shotBundleProject
Link version to	testAsset4RVGUN
Discipline	Art
Task	testTask4Python
Description	
Tags (separated by ,)	

Cancel

SG

> Write a description (*Description*) of the version.



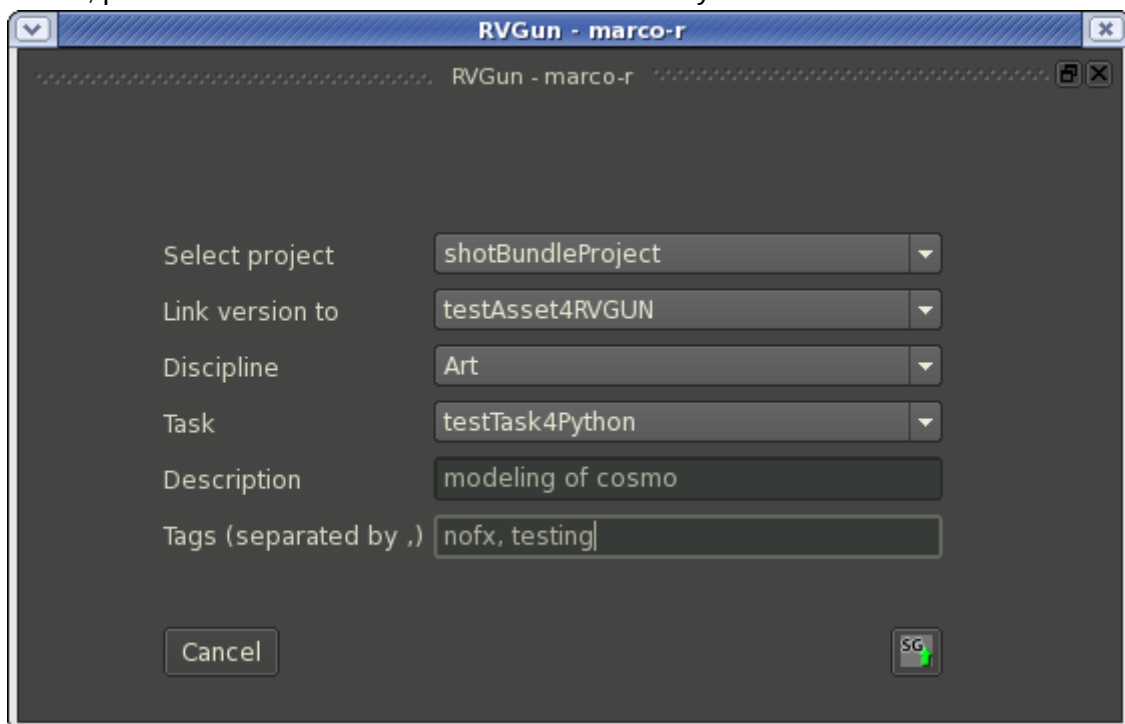
The screenshot shows the same dialog box as before, but now the "Description" text input field is filled with the text "modeling of cosmo". The other fields remain the same: "Select project" (shotBundleProject), "Link version to" (testAsset4RVGUN), "Discipline" (Art), "Task" (testTask4Python), and "Tags (separated by ,)" (empty). There is a "Cancel" button at the bottom left and a small "SG" icon with a green arrow at the bottom right.


Select project	shotBundleProject
Link version to	testAsset4RVGUN
Discipline	Art
Task	testTask4Python
Description	modeling of cosmo
Tags (separated by ,)	

Cancel

SG

> Write some tags (*Tags*) separated by commas. Tags are good for searching data into the database, please take one minute to think about some keywords.



> Before clicking the bottom-right button  to start the publishing process, use the RV main timeline to navigate through frames and choose the frame that you want to use as thumbnail.

> Now click on the bottom-right button. You will see a dialog box (Figure 5). If you are 100% sure of publishing the version to shotgun then type yes and click *OK*.

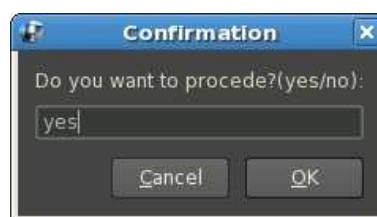


Figure 5. Confirmation dialog

> Now the computer will freeze for some time, seconds or minutes depending on the length of the frame sequence. When the system is ready you will be able to continue working. You will also see a message in the RV console telling you that everything is fine (Figure 6).

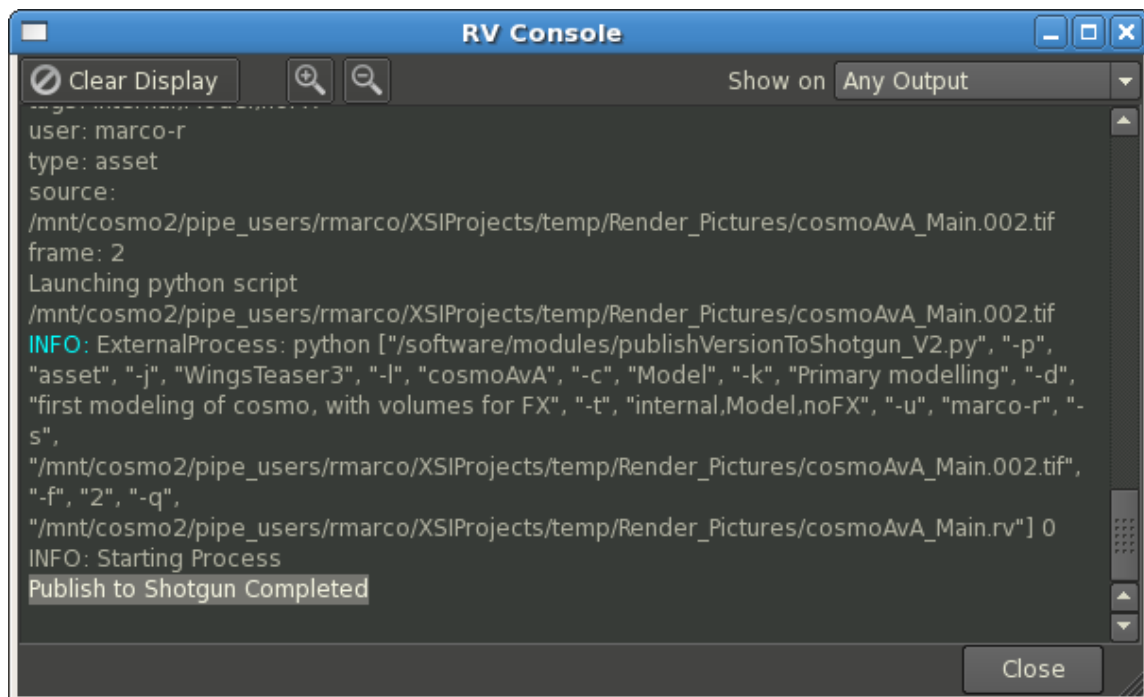


Figure 6. RV Console with success message

Now the version will be visible in Shotgun with a link (*rvlink*) back to the original contents in RV also.