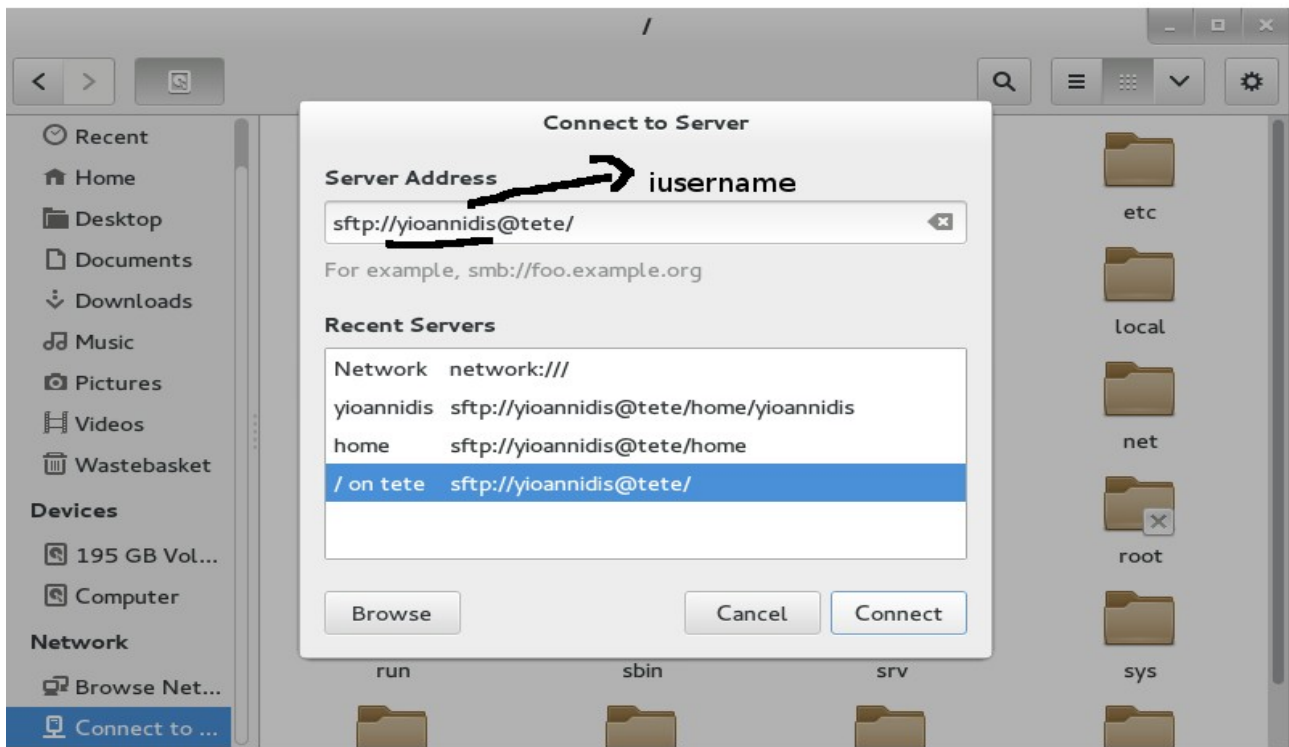


# Qube Renderfarm Guide: Houdini Mantra IFD file rendering

## Scene setup

In order to prepare your scene for submission to the renderfarm, the following steps are suggested:

- Place scene file, and associated scene assets (eg. Textures, Sims etc.) in folders, within a single directory
- Make use of relative file paths using the \$HIP or \$JOB variables when assigning all scene assets
- (If \$JOB is being used it can be set in the textport i.e.  
set -g JOB = /render/**isusername**/SimulationQubeTest)
- Copy the entire scene folder to your directory within /render on the tete server.
  - This can be done using the Connect to Server... option which can be found in the Places main linux menu, all the way down



server address: **sftp://isusername@tete/**

- It is sensible to add a bookmark to this location, so it can be easily accessed when setting up and accessing future renders.
- Copy and Paste can be used to copy your scene directory into this directory ready for use on the render farm.

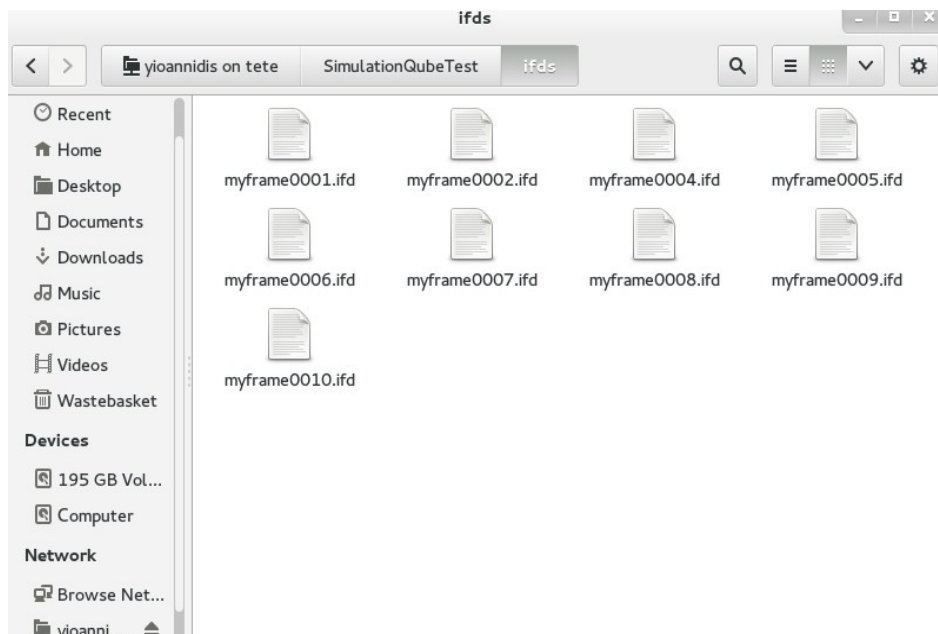
◦ Alternatively files can be copied using the sftp command line tool.

- Start Qube:
- Submit a Houdini job by clicking on: **Submit-Cmdrange Job...**

## Suggested Qube Settings

The following defaults are a good starting point for starting a Houdini job on the renderfarm.

Make sure there is a directory with all the ifd files need for mantra to read.

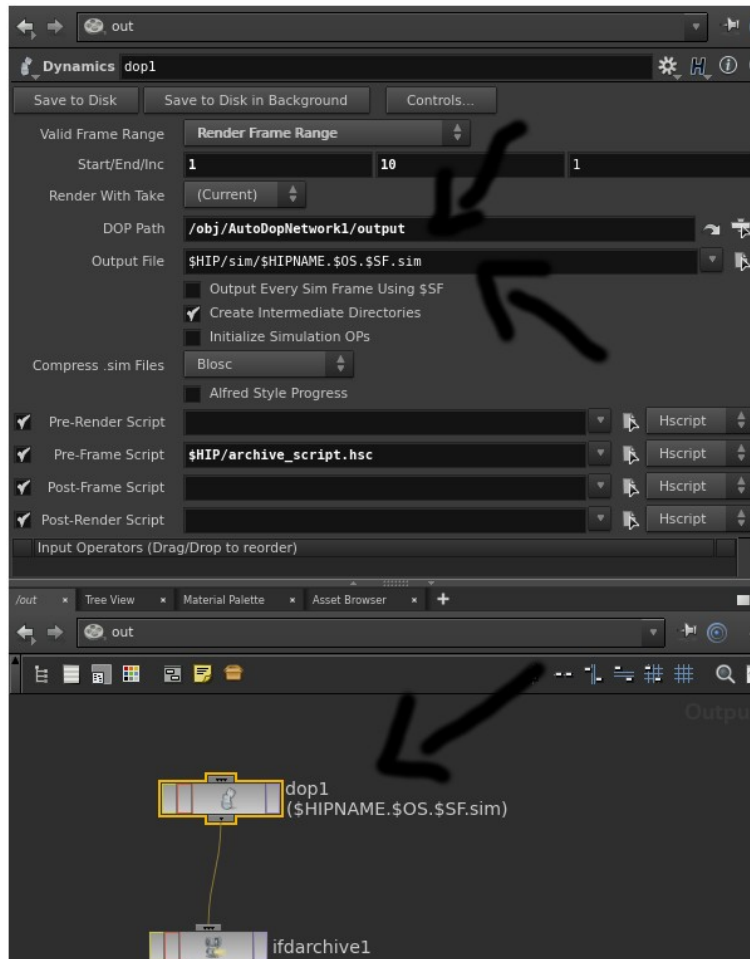


The same applies if you want to render a Simulation on the farm:

Make sure there is a directory with all the sim files needed, as well as a similar setup inside the AutoDopNetwork to export the sim files to it.

Additionally, a Dynamics dop in the Output network needs to be added to render out the sim files from the AutoDopNetwork.

(see scene setup bellow):





archive



bakedgeo



outputframes



sim

archive\_script.hsc



Hearts\_Royals.jpg



renderimages



simpleShattering.  
hipnc

backup



ifds

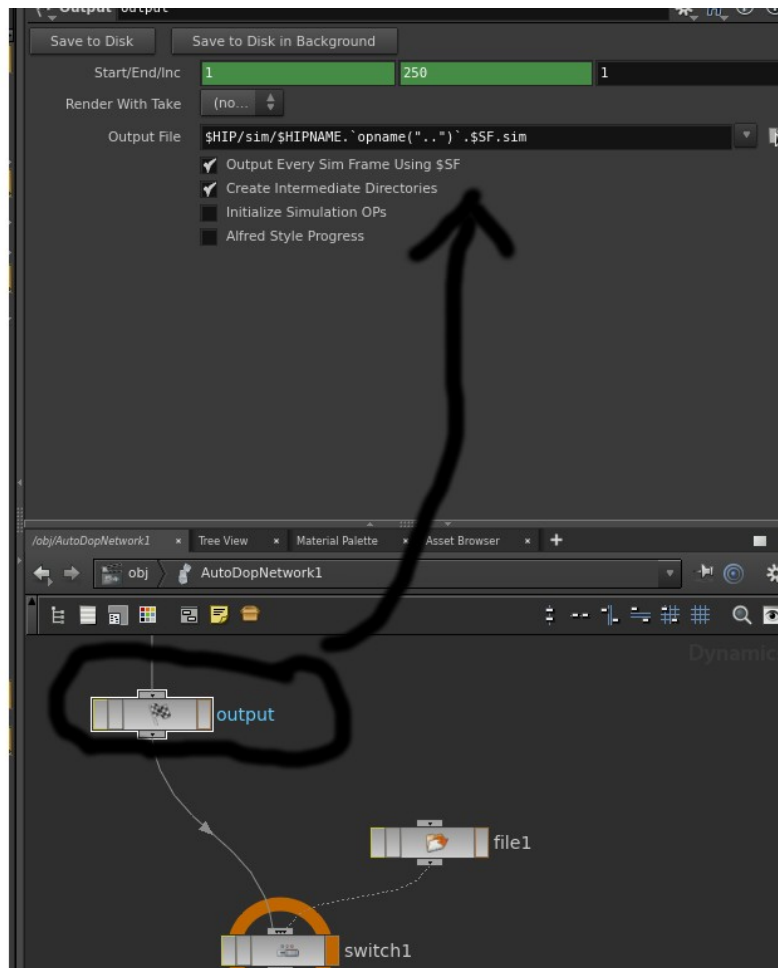


render\_script.hsc



simpleShattering15.  
hipnc





(To generate the ifds refer to the **Qube\_Renderfarm\_Guide\_Houdini\_Hrender.docx** guide)  
(basically resubmit a HRender job on Qube and remember that the .hipnc scene file you are trying to render needs to have the “Save IFD files to disk” checkbox of your mantra out driver ticked)

**Instances:** Set number of frames to render in parallel (not more than 40)

**Range:** Specify frame range in the format start-end.

**IFD file:** Enter path to the ifd file to render i.e.

*/render/**iusername**/SimulationQubeTest/ifds/myframe.iff*

**Command:** Render command to be executed. This needs to include commands to initialize the HoudiniEnvironment. We recommend starting with the entire template shown here:

**cd /opt/hfs16.0.557.;source ./houdini\_setup\_bash;**

**mantra -f /render/**iusername**/SimulationQubeTest/ifds/**myframe**QB\_FRAME\_NUMBER.iff;**

**Resubmit cmdrange**

---

**Qube Job Basics**

Name

Priority

Instances

---

**Qube Frame Range**

Range

---

**Preview Frames Submission**

Preview Priority

---

**Parameters**

Command

Frame Padding

---

**Qube Worker Selection**

Priority Cluster

Reservations


---

**Qube Advanced Job Control**

Flags


Email (job complete) ☐

---

☐ Expert Mode 

Be careful and use the same frame padding as the one in the scene you generated the ifds from

**Resubmit cmdrange**


Frame Padding  

**Qube Worker Selection**

Hosts	<input type="text"/>	<input type="button" value="Browse"/>
Groups	<input type="text"/>	<input type="button" value="Browse"/>
Omit Hosts	<input type="text"/>	<input type="button" value="Browse"/>
Omit Groups	<input type="text"/>	<input type="button" value="Browse"/>
Priority Cluster	<input type="text" value="/"/>	<input type="button" value="Browse"/>
Host Order	<input type="text" value="+host.processors.avail"/>	<input type="button" value="Browse"/>
Requirements	<input type="text"/>	<input type="button" value="Browse"/>
Reservations	<input type="text" value="host.processors=1"/>	<input type="button" value="Browse"/>
Restrictions	<input type="text"/>	<input type="button" value="Browse"/>

**Qube Advanced Job Control**

Flags	<input type="text" value="auto_mount"/>	<input type="button" value="Browse"/>
Dependency	<input type="text"/>	<input type="button" value="Add"/>
Email (job complete)	<input type="checkbox"/>	<input type="text" value="i7630130"/>
Email (failed frames)	<input type="checkbox"/>	<input type="text" value="i7630130"/>
Blocked	<input type="checkbox"/>	

☒ Expert Mode 

**Environment Variables:** Used to add environment variables.



Must have variable for license server set here

Qube Advanced Job Control

Flags

Email (job complete) ☐

Email (failed frames) ☐

FlightCheck scripts ▲

Qube Job Delayed Start ▲

Qube Job Environment ▼

Cwd

Environment Variables


Key	Value
HOUDINI_USE_HFS_PYTHON	1
SES_LMHOST	burton.bournemouth.ac.uk

Qube Job Validation & RegularExpression-based Output Parsing ▲

Qube Actions ▲

Qube Notes ▼

Notes

☐ Expert Mode 



### Resubmit cmdrange

Impersonate User

#### Qube Job Validation & RegularExpression-based Output Parsing

Min File Size

0

regex\_highlights

regex\_errors

regex\_outputPaths

regex\_progress

regex\_maxLines

20

#### Qube Actions

generateMovie

☐

#### Qube Notes

Account

Notes

Set Defaults

Clear Defaults

☒ Expert Mode



Cancel

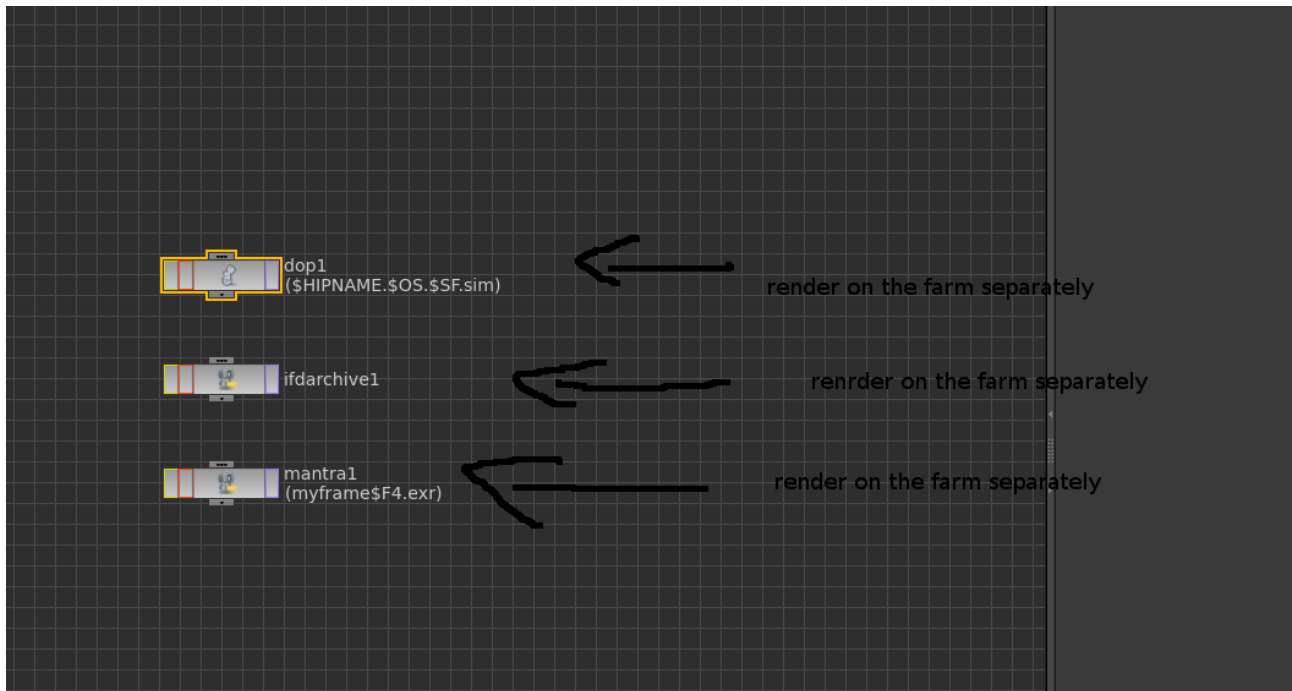
Resubmit

### Tips if you find you get dropped frames:

If you see that some of your ifd frame files are dropped for no apparent reason, you can **re-render** and **try to re-generate the IFD files**, which shouldn't take long usually cause they are much faster to generate rather than rendering from the live Houdini scene network.

**OR**

Instead of using a chain in the outputs level in Houdini, just disconnect and render separately each one of them



Resubmit Houdini (hrender)

Qube Job Basics

Name: yanHrenderSimpleSimulationQubeTest

Priority: 9999

Instances: 10

Qube Frame Range

Range: 1-10

Preview Frames Submission

Preview Priority: -1

Required Parameters

hrender: hrender [Browse]

sceneFile: /render/yioannidis/SimulationQubeTest/simpl [Browse]

outputLogfile: /render/yioannidis/SimulationQubeTest/simpl [Browse]

Driver/Cop: -d

Driver/Cop Value: /out/mantra1

Optional Parameters

R: ☒

Qube SimpleCmd and Shell Parameters

cd /opt/hfs ; source houdini\_setup\_bash;%(csh)s "%

Set Defaults Clear Defaults ☐ Expert Mode [Icon] Cancel Resubmit

Should change to either: dop1

Should change to or: ifdarchive1

Should change to or: mantra1