

This tutorial will show you how to use Deadline to remotely render your Blender project using DMA's computers.

NOTE: DMA currently has Blender version 4.5 LTS installed on our machines. If you are trying to render from a different version of Blender you might get errors.

---

First, if you are working from your personal computer, follow the instructions in the tutorial for setting up Deadline. If you are on a DMA machine it should already be installed.

[Preparing To Use Deadline MacOS](#)

or

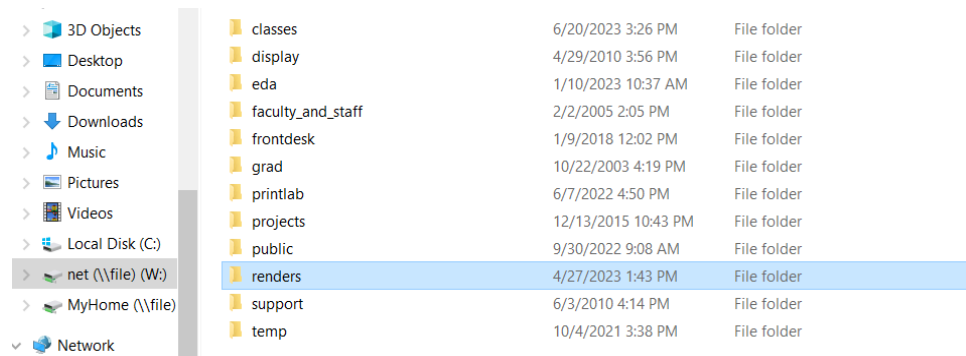
[Preparing To Use Deadline Windows](#)

---

Ensure the “Output Properties” and “Render Properties” in your Blender file are set to what you are wanting to be rendered. Your path will need to be relative, not absolute (so something like ‘//myrenders/’ instead of ‘/Users/jonathancecil/Desktop/Deadline Renders/jonathancecil/myrenders/’).

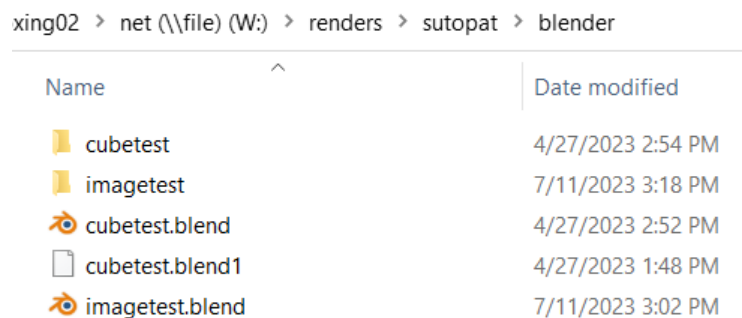
*NOTE: If you are trying to render to a video file rather than a PNG sequence, Deadline is set up to divide up the work across multiple machines. You could do this and have it send you the video in multiple video files, or you could set it so that the whole thing renders on one computer remotely (this is potentially risky as the longer one job works on one computer the more likely it is to be interrupted and fail). It is highly recommended to render frames and assemble the video later.*

Save your .blend file to your subfolder in the Renders folder, which you can access either via cloud.dma.ucla.edu, or, if you are on a DMA machine, is in the 'net W:/' drive you can find in the File Explorer.



Inside this folder you'll see other DMA users with their own folders, and if you don't already have one you should make your own.

This is where you will save your .blend file. It's a good idea to also create a subfolder within that to have your rendered images go into so that your main folder doesn't get disorganized



Now you open the Deadline Monitor program and you will see this screen.

The screenshot displays the Deadline Monitor application window. The title bar reads "Deadline Monitor - render.dma.ucla.edu:4433 (Deadline10RemoteClient.ptx)". The interface is divided into three main sections: Jobs, Tasks, and Workers.

**Jobs Panel:** Shows a list of submitted jobs. The top summary bar indicates "721 total, 492 completed, 72 suspended, 157 failed, 1 selected". The table columns are Job Name, User, Errors, Comment, Department, Task Progress, and Size. The first job, "Untitled", is selected and shows 38 errors and 100% progress (5/5).

**Tasks Panel:** Shows a list of tasks for the selected job. The top summary bar indicates "5 total, 5 completed" and "00:00:00:00 remaining". The table columns are Task ID, Frame, Status, Progress, and Worker. All tasks are completed with 100% progress.

**Workers Panel:** Shows a list of available render workers. The top summary bar indicates "90 total, 34 idle, 44 offline, 12 stalled, 1 selected" and "0% of 4.5 THz utilization". The table columns are Worker Name, Machine Name, Machine User, Description, Status, Last Status Update, Offline Message, Comment, Assigned Pools, Assigned Groups, and Dequeuing Mode. The first worker, "BLUE951", is selected and shows a status of "Offline".

The top-left window is a list of submitted jobs, which you can click on and then see how they were distributed into separate jobs to be sent to computers to be rendered in the top-right window. The bottom window shows a list of the currently available computers.

To start a Blender job, on the top toolbar click Submit > 3D > Blender, and you will get a window popup that looks similar to this:

Submit Blender Job To Deadline

Job Options Draft

**Job Description**

Job Name: Untitled

Comment:

Department: Jonathan

**Job Options**

Pool: none

Secondary Pool:

Group: blender

Priority: 50

Task Timeout: 0 ☐ Enable Auto Task Timeout

Concurrent Tasks: 1 ☒ Limit Tasks To Worker's Task Limit

Machine Limit: 0 ☐ Machine List Is A Deny List

Machine List: ...

Limits: ...

Dependencies: ...

On Job Complete: Nothing ☐ Submit Job As Suspended

**Blender Options**

Blender File: animation\restaurant\_anim\_test\rain\_restaurant.blend ...

Output File (Optional): ...

Frame List: 100-200

Frames Per Task: 1 ☐ Submit Blender Scene File With The Job

Threads: 0

Build To Force: None

Path to BlenderForceGPU Script: ...

Submit Close

(this window shows settings for Cycles CPU rendering)

The fields you want to pay attention to are:

- Job Name
  - Put something descriptive here so that you know what it is when it appears in the long list of jobs
- Group
  - Select “blender”
- Blender File
  - Select your .blend file that you put in the cloud render folder
- Output File
  - Select the subfolder in your render folder where you want the renders to be output to

- Frame List
  - Which frames in blender do you want to be rendered.
    - Using a single number, like 73, will render that single frame
    - To render frames sequentially, enter the first frame number followed by a '-' character, and then enter the last frame number.
- Frames per task
  - This is how many frames will be sent for a job to each computer.
    - If your frames will take a long time to render you ideally will keep this value low to minimize the chance for interruption and failure. If they are quicker renders you can put this higher to make it more efficient. If you don't know where to start, try 5 and see how it goes.
- Path to BlenderForceGPU Script
  - For standard CPU rendering in Cycles, you can leave this blank. We've hacked this section so that we can pass a GPU rendering flag to the Blender render command. If you want to, or need to, render with OptiX (GPU with ray tracing) or CUDA (GPU render), you can pass the appropriate command as listed below.
    - For OptiX rendering in cycles enter this as your additional argument (without the quotes)
      - "\\render\renders\extras\BlenderForceOptixScript.py"
    - For Cuda rendering in cycles enter this as your additional argument (without the quotes)
      - "\\render\renders\extras\BlenderForceGPUScript.py"

Then you can hit submit!

---

You can watch the progress of your job by clicking on it in the job list. If you start to see red errors, you can right click on your job and select "View Job Reports" which will pop up a window with logs from the processes. If you search around in these logs and find an error they will hopefully give you a clue as to what went wrong! If not, come to the helpdesk and we can help you figure it out.

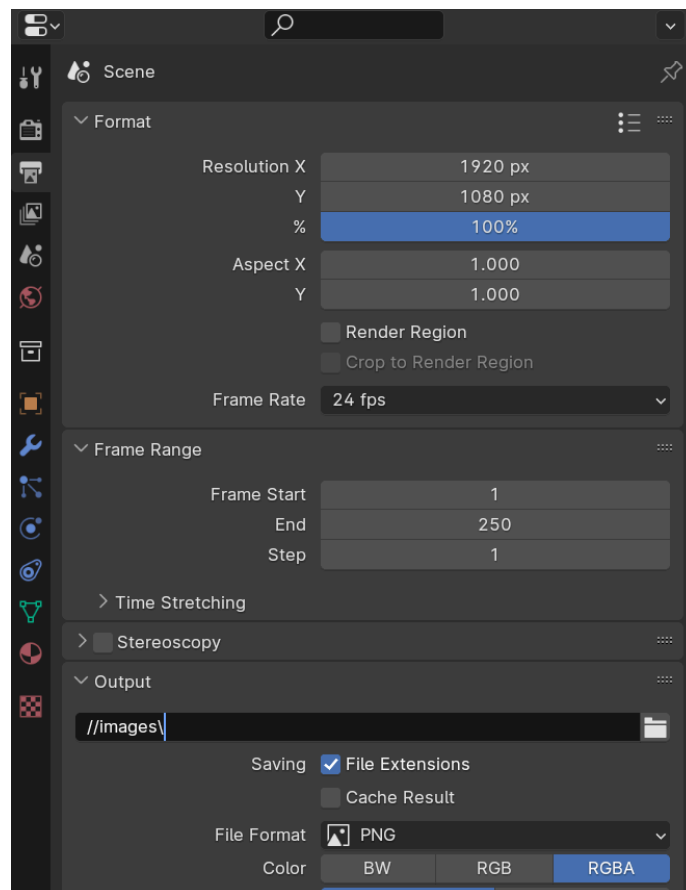
---

If all goes well your renders will appear in the folder in the cloud!

## Troubleshooting

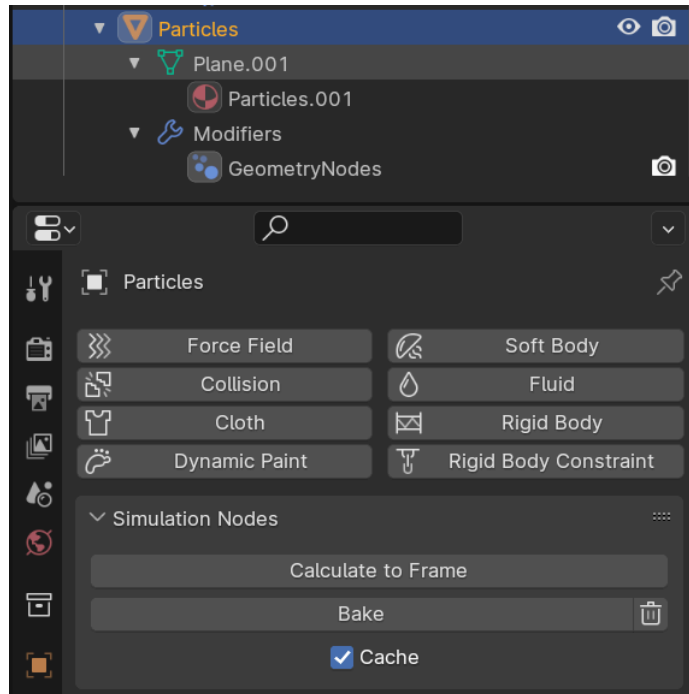
### Where are my renders?

If Deadline Monitor shows that your render is done, the frames should be on the DMA Cloud in your Deadline Renders folder. If you can't find them where you expect, there is a good chance your render output folder is set to /tmp in Blender. Make sure the output location in your Scene->Output->Output is set to a relative path using the "//" shortcut. Here's an example of how the setting should look inside Blender for a images folder that is in the same directory as the Blender file:



### Why are my particles not rendering?

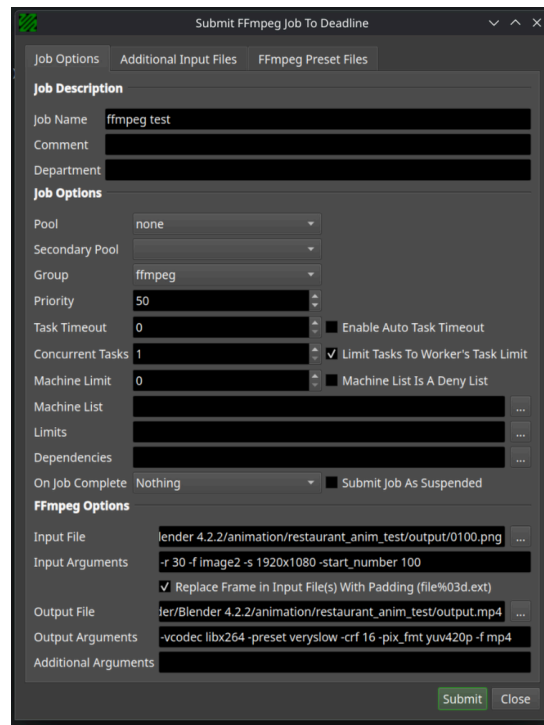
If your particle systems are rendering correctly when rendered locally in Blender, and you see that particles are not rendering as expected through Deadline, it could mean that you need to bake your particle system into the file. In Blender, set the current frame to the last frame in your animation. From the outliner, select your particle system, click on the physics properties tab, select simulation nodes, and click “Calculate to Frame”. Make sure the cache option is selected, and then “Bake”. When you render again, your particle system will be baked into the file and should render as expected.



## Bonus

### FFmpeg assembly of frames

If you are rendering out frames for assembly into a video, you need to take your frames and encode them to a video file. You can use a tool like Adobe Media Encoder on your local system, or you can submit a job to Deadline. Deadline can use a command line encoder called FFmpeg, and there is a submission window for it in the Deadline Monitor under Submit->Processing->FFmpeg. It's on you to provide the command line arguments, but here are some hints to get you started.



### Input File

- Choose the first frame in your sequence.

### Input Arguments

- `-r 30 -f image2 -s 1920x1080 -start_number 100`
- You need to include the start number of your first frame, usually 1 but in this case was 100

### Output File

- Select the output directory and select a name for your new output file.

### Output Arguments

- `-vcodec libx264 -preset veryslow -crf 16 -pix_fmt yuv420p -f mp4`