Bryce Lightning (Network Rendering)

Bryce Lightning is a network rendering system that permits distributing a render job to several computers on a network. What must be observed to make it work is discussed.

Introduction

Lightning is not a tool to add lightning and thunderbolts to a scene. The name Lightning was chosen to imply that a scene renders with lightning speed. This is a bit optimistic, Bryce does not render faster, but a render will finish faster when more computers share the task. Without at least two computers connected together over a home network, Lightning is useless.

History

Version 1 of Lightning was introduced with Bryce 5.0. It worked fine and the host computer automatically partook in the render. However, finding the clients took very long. Bryce 5.5 came with Lightning version 2 (later called 2a) and it worked. Bryce 6.0 featured Lightning 2b, a version that never worked reliably. Bryce 6.1 finally came with Lightning 2c which works nicely and Bryce 7.1 Pro still uses this version. Since version 2, Lightning must be started on the host machine, otherwise it does not take part in the render — though it controls it.

Making Lightning work

Setting up a network render is not complicated but not everything is intuitive. Even if a still image is to be rendered, it needs to be set up as an animation. The Lightning client must be installed on all computers that will participate in a network render. And these computers must be connected over a TCP/IP network.

Mac and PC

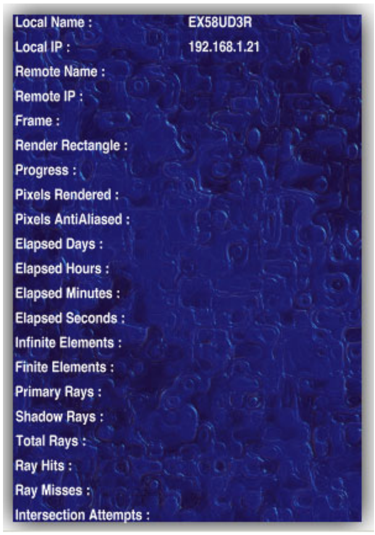
Mac and Windows PC clients cannot be mixed if the host runs on a PC. However, if the host runs on a Mac, clients are served by it on either type of machine. I could not test this myself but I was told that it works this way. If your home network consists of Macs and PCs and Lightning is to be run on both types of machines, just be aware that there might be issues. This document is PC centred but most if it also applies to the Mac.

Installing Lightning

Lightning can be installed anywhere on a computer. For the host computer (the one running Bryce), a good idea would be in a folder *Lightning* just below the *Bryce* main folder but this is not mandatory.

On the client-only computers, you may want to create a Bryce\Lightning folder structure but again, you are completely free where you install Lightning. An installed Lightning can also be copied to the host computers but this procedure is not recommended.

Once Lightning is installed, it should be tested whether it runs. On the lower right corner, the window at right should appear. The computer name and its network IP address are shown.

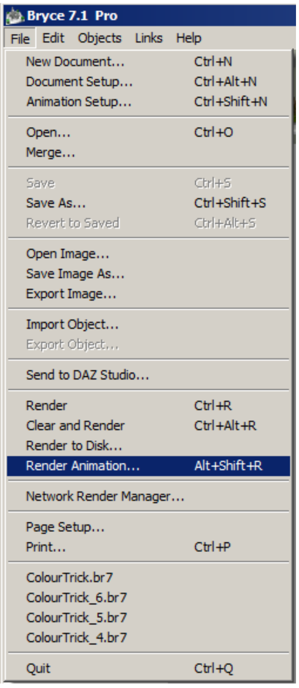


Local Name : EX58UD3R
Local IP : 192.168.1.21
Remote Name :
Remote IP :
Frame :
Render Rectangle :
Progress :
Pixels Rendered :
Pixels AntiAliased :
Elapsed Days :
Elapsed Hours :
Elapsed Minutes :
Elapsed Seconds :
Infinite Elements :
Finite Elements :
Primary Rays :
Shadow Rays :
Total Rays :
Ray Hits :
Ray Misses :
Intersection Attempts :

The Client Window

This client window shown is completely passive. When the client renders, the statistics are shown and updated. Once this client is finished, it looks again empty as shown above. This client is ready to render another scene.

Once all network rendering jobs are concluded, the question is how to get rid of the window, i.e. terminate the client. Move the mouse to the upper edge of the display and a bar to close it appears (this is a Bryce 5.0/5.5 left-over). It can also be closed from the Task Bar at the bottom of the screen where the program thumbnail can be found.



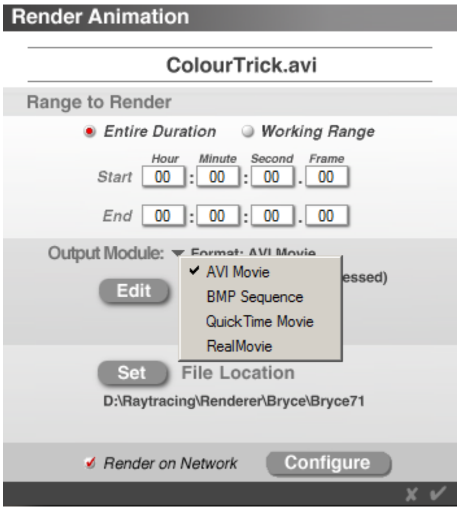
Testing Network Rendering

Before running a network render job the first time after installing, it is a good idea to just run Lightning on the host computer. Once it is confirmed that it runs, the network can be tackled with the confidence that it generally works.

With Lightning running on the host machine, launch Bryce and create a simple scene that renders within a minute or two. The default scene renders too fast to check everything and see how it works. Once the scene is ready to render, start a network render. You will have to minimise the Bryce window to see the client window and watch what is going on.

Setting up a Network Render

Click on *File* on the tool bar on top and select *Render Animation...* or hold down the [Alt] and [Shift] keys and hit [R]. This is shown at left. This is the way to access the network render host whether for an animation or a still image.


Render Animation

ColourTrick.avi

Range to Render

● Entire Duration ○ Working Range

	Hour	Minute	Second	Frame
Start	00	00	00	00
End	00	00	00	00

Output Module: Format: AVI Movie

File Location

D:\Raytracing\Renderer\Bryce\Bryce71

✓ Render on Network

The *Render Animation* dialog shown at left opens. For a still image, enable *Entire Duration*. For an animation, you may want to render only the *Working Range* and have to set *Start* and *End* Time. No need to set those for a still.

Click in the down arrow right of *Output Mode* to set the file type. For a still, select *BMP Sequence*, for an animation anyone you wish. *Edit* permits to adjust different parameters for animations.

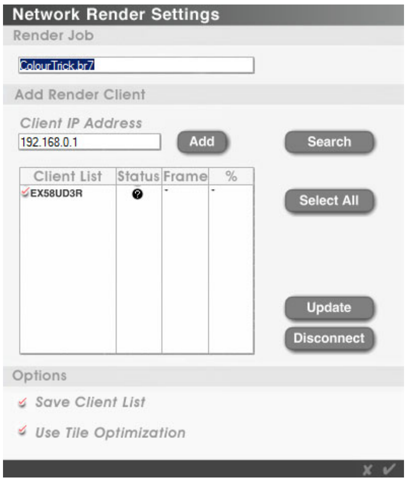
Click on *Set* to open the *Save As* dialogue of the operating system to specify location and file name for the rendered.

Finally, enable *Render on Network* and click on *Configure* to open the *Network Render Settings* dialogue.

This dialogue is shown below and needs some explanations. Consider the image at left on the next page. The Bryce file to render is shown on top. You can enter either the IP address or the name of the computer running a client, then click on *Add*. Lightning adds this client to the list below. If the IP address was entered and the client reported back the computer name, that one will be added to the list.

There is an easier way: just click on *Search* and Lightning will add each computer running a Lightning client; adds it to the list and selects it (small red tick to the left of the name).

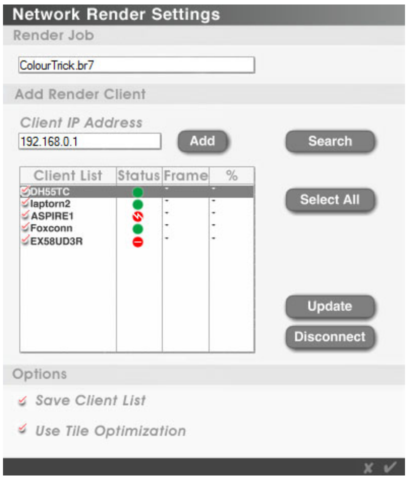
Any computer name in the list can be selected or deselected individually. With *Select All*, all in the list are selected. The line with any computer name can also be selected (below centre) and by hitting the [Delete]-key removed from the list. Clicking on *Disconnect* does the same.



ColourTrick.br7

Client IP Address: 192.168.0.1

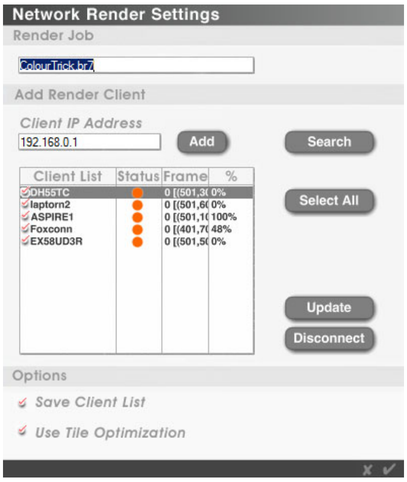
Client List	Status	Frame	%
✓ EX58UD3R	?	-	-



ColourTrick.br7

Client IP Address: 192.168.0.1

Client List	Status	Frame	%
✓ DH55TC	●	-	-
✓ laptorn2	●	-	-
✓ ASPIRE1	●	-	-
✓ Foxconn	●	-	-
✓ EX58UD3R	⊖	-	-



ColourTrick.br7

Client IP Address: 192.168.0.1

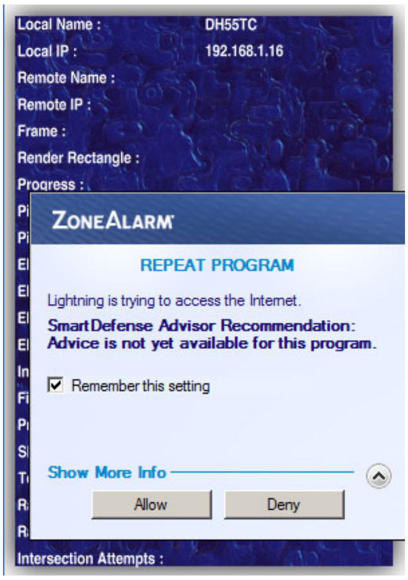
Client List	Status	Frame	%
✓ DH55TC	●	0 [(501,3(	0%
✓ laptorn2	●	0 [(501,6(	0%
✓ ASPIRE1	●	0 [(501,1(	100%
✓ Foxconn	●	0 [(401,7(	48%
✓ EX58UD3R	●	0 [(501,5(	0%

The colour dots in the *Status* column have the following meanings:

- Black with a ?: client is unknown, not found.
- Red: found but (flash) not ready, (minus) failed.
- Green: idle, ready for a render job.
- Orange: busy rendering.

The column *Frame* shows which part of the image is rendered by the respective client. For animations, this number is obvious. For stills, the column is too small to hold the line and column values.

The % column shows how the respective client does. This percentage indicates how complete the frame is or for a still, how much of the current tile has already been rendered. Sometimes there appear very funny numbers, a known bug, but luckily without any consequences. All columns update rapidly.



Local Name : DH55TC
Local IP : 192.168.1.16
Remote Name :
Remote IP :
Frame :
Render Rectangle :
Progress :
Intersection Attempts :

REPEAT PROGRAM

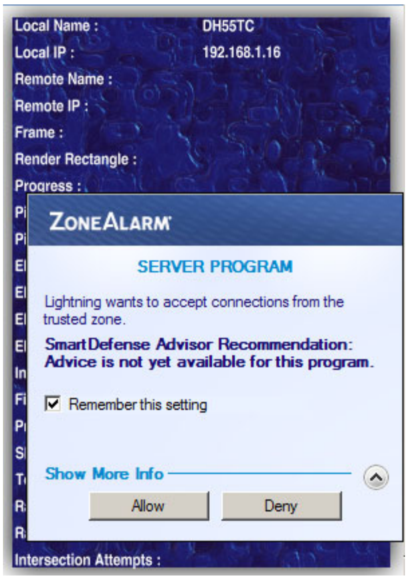
Lightning is trying to access the Internet.

SmartDefense Advisor Recommendation:
Advice is not yet available for this program.

☑ Remember this setting

What about the red status? The client was found but it is inhibited and cannot communicate. The reason is a Firewall that blocks. You have to check the settings of the firewall used. In my case, it is ZoneAlarm that blocks.

First (left) it complains that Lightning on the DH55TC wants to access the network and when permitted, the next complaint came up because Lightning wants to accept a connection from the host.



Local Name : DH55TC
Local IP : 192.168.1.16
Remote Name :
Remote IP :
Frame :
Render Rectangle :
Progress :
Intersection Attempts :

SERVER PROGRAM

Lightning wants to accept connections from the trusted zone.

SmartDefense Advisor Recommendation:
Advice is not yet available for this program.

☑ Remember this setting

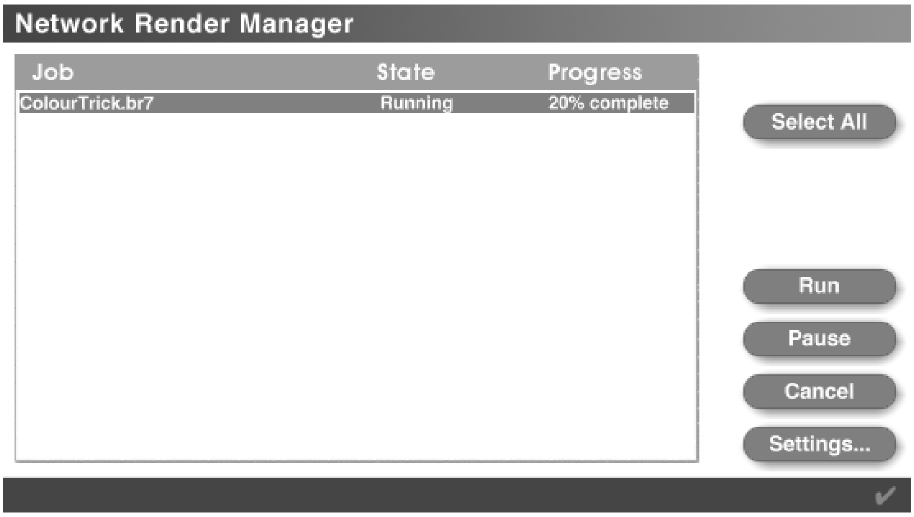
These are typical issues when Lightning is run for the first time. Therefore, the security settings of all computers running Lightning must be checked and the both-way connection to and from the host computer allowed.

Once all clients seem to work, click on *Update* to make Lightning contact all clients again. Only the clients with a green status can be used for work.

There is an option near the end of the *Network Render Settings* dialogue that permits to save the client list. Next time a network render job is started, Lightning reads the list and you have only to click on *Update*. The file containing the client list is saved in the Bryce main folder with the name `ClientList.txt` — but this is not a text file that can be opened with a text editor.

The final option at the bottom of this dialogue is *Use Tile Optimization*. Enable this option if you render a still picture. The image will be divided into tiles the size of 100 x 100 pixels and the tiles sent to each client which renders it and sends it back to the host once finished.

If you render an animation make sure this option is disabled. A frame can be considered as being a tile and tiling a frame would only make the render slower.



Network Render Manager

Job	State	Progress
ColourTrick.br7	Running	20% complete

We are not yet done. Once everything is fine in the *Network Render Settings*, accept to close it. The *Network Render Manager* dialogue pops up. It lists all jobs currently at work. *Run* starts the selected job, *Pause* interrupts it and *Cancel* abandons it. Clicking on *Settings* brings up the *Network Render Settings*.

The dialogue above can be closed by clicking on accept. It can be re-opened from the tool bar on top: *File* > *Network Render Manager...* (see page 2).

Troubleshooting

There are Bryce and network related issues that can present themselves. Here are a few of them and what can be done to isolate and fix them.

Bryce

Render priority is set on the host machine only. The way priority is set will be the setting of each client. *Low Priority* uses one single CPU core; *Normal Priority* uses half of the available CPU cores up to 8 and *High Priority* all up to 8. Note that for a multi-threaded CPU, this is also counted as a core. A computer featuring a multi-threaded 4 core processor will be considered by Bryce and Lightning as an 8 core CPU.

The Lightning host sends out the complete source file to each client in sequence. This file is called `nartemp.bry` which stands for network assisted rendering temporary bryce file. This file is stored on each client in the folder where Lightning is installed. If the render is complete, this file is deleted. A large file can take a while until all clients have it because they are not sent as broadcast or multicast together.

The client must have enough memory to load the complete source file. This seems unnecessary when rendering a still image with tiles 100 x 100 pixels, but it becomes clear that for an animation, the complete source file is needed to render a frame.

There cannot be an unlimited number of clients. More than 16 are not served. Unfortunately, I cannot test this limit because I only have five computers.

Bryce Lightning cannot be set to a dedicated IP address and it cannot NAT (network address translation). It uses the one of the computer on which it runs. It also uses ephemeral TCP and UDP ports. Depending which ports are free when Lightning is launched the ports can vary. However, once the start is set, each additional client gets the next one.

Typical TCP ports used start at around 3420 on the host to the clients and listens on the TCP ports starting at around 28250. The TCP ports are used to send the files and accept the rendered tiles. For commands, Lightning uses UDP with ports starting around 28260 (see also `https://www.horo.ch/raytracing/tests/nwr.html`).

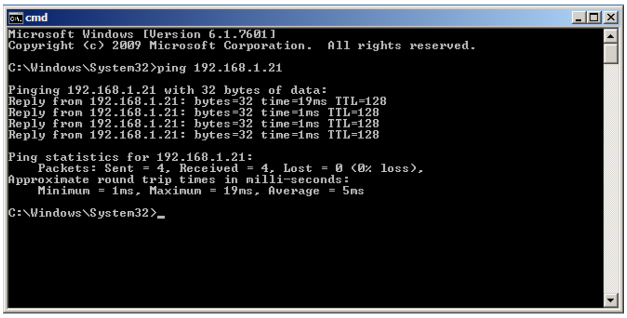
Because there is only the IP address of the computer, not a dedicated one for Lightning, it gets very difficult to render over the public Internet, though it can be done. Port address translation (PAT) must be configured in the router connecting the home or local network to the Internet. Again because of the ephemeral nature of the ports, this becomes quite complicated.

Network

It would go too far to elaborate on how to set up a local or home network. There is the simple part of connecting the machines using RJ45 category 5 (or better) network cables, preferably via a Switch or Hub — or using a WLAN. Then there is the network configuration of the computer, either give it a dedicated IP address or set it to DHCP (dynamic host control protocol) and decide which unit in the network will assume this function. If the home network sports a router it can act as DHCP server. Then a list of IP addresses can be defined and even though each computer in the network gets its address from the DHCP server, it will always be given the same address. This is a helpful option for troubleshooting.

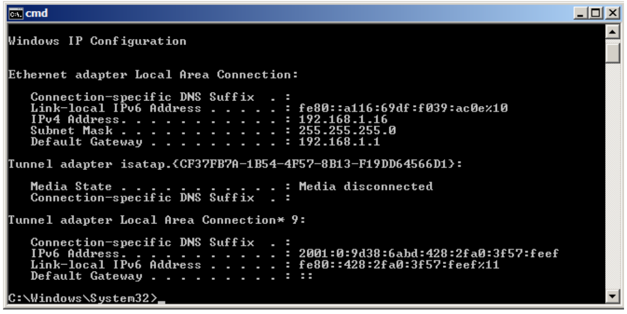
If the Lightning host cannot find a client on which Lightning runs, there are a few things that can be checked.

Start the command window, CMD on the host computer. In each Lightning window on the clients you find its IP address (in the example page 1 it is 192.168.1.21), then type `ping 192.168.1.21` (or the number you find) on the command prompt. If you get an answer as shown at right, the host can see that client on the network.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\System32>ping 192.168.1.21

Pinging 192.168.1.21 with 32 bytes of data:
Reply from 192.168.1.21: bytes=32 time=19ms TTL=128
Reply from 192.168.1.21: bytes=32 time=1ms TTL=128
Reply from 192.168.1.21: bytes=32 time=1ms TTL=128
Reply from 192.168.1.21: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.1.21:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 19ms, Average = 5ms

C:\Windows\System32>_
```

If it cannot, then the host may not be in the same network. To verify this, you have to verify the subnet masks match by typing at the prompt *ipconfig*. There is quite some information but there is only the *Subnet Mask* to check (example: `255.255.255.0`). It should be the same on both computers (well, they can be different, but calculating the subnet mask goes a bit far).



```
Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::a116:69df:f039:ac0e%10
   IPv4 Address. . . . . . . . . . . : 192.168.1.16
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1

Tunnel adapter isatap.{CF37FB7A-1B54-4F57-8B13-F19DD64566D1}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Local Area Connection* 9:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2001:0:9d38:6abd:428:2fa0:3f57:feef
   Link-local IPv6 Address . . . . . : fe80::428:2fa0:3f57:feef%11
   Default Gateway . . . . . . . . . : ::

C:\Windows\System32>_
```

If all looks fine, check the firewall and also the permissions. You may have to run things as Administrator and open TCP and UDP ports in the firewall or just the applications.

The above are quite simple tests and make up 95% of all issues encountered. Once you have Lightning running and rendering, you can pull out the network cable or stop the client from rendering and no tile or frame will be lost. The unfinished one is discarded and it will be given to another client to render. If the disconnected client is reconnected, a new tile or frame is sent to it and it will resume the work. Lightning 2c is quite stable and a pleasure to use.

Render Speed Considerations

Usually, setting the host to *High Priority* will be the logical choice. If the computers in the home network belong to family members that agree that part of their machine can be used for network rendering, using *Normal Priority* would be the better choice. The render will finish later but the computers hosting Lightning can still be used by their owners without noticing that it crunches numbers in the background as long as no CPU intensive games are played at the same time.

Once Lightning was made working, the next time it will be set up in no time. However, to start up Lightning for a 5 minutes render job does not make much sense. At a rough estimate, if the render does not need an hour or more, it is not worth the trouble, particularly if the source file is large and needs quite some time to be sent to and received by all clients.

Use the fastest machine as host; it is always the slowest machine that dictates the overall time of the network render. It can be frustrating to wait for the last machine to finish the very last tile or frame when all other machines finished already some time ago.

Bryce Memory Usage

It is well known that Bryce — still being a 32-bit application — has a 2 GB limit. This is also true for Lightning. To launch a network rendering job, Bryce takes an additional slice of 600 to 700 MB the moment *Render to Network* in the *Render Animation* dialogue is enabled. If the scene is large, this may also take quite long.

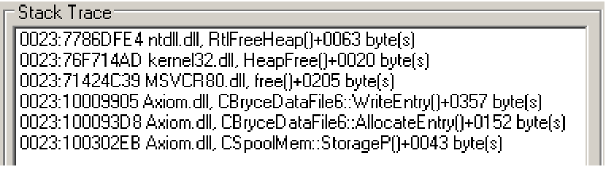
If a Lightning client cannot handle the file, an *Out of Memory* message appears. When it is acknowledged, another message tells that an object is missing and Lightning will continue without it. The final image may miss an object.

In the test setup was one object 300 MB and two 145 MB ones. The Bryce scene 1310 MB and when set to network render, 2 GB. Both Lightning clients on two computers took 1290 MB each when they complained about exhausted memory. One of the 145 MB objects was not rendered and the memory used by each Lightning was down at 987 MB.

Setting the large address aware flag on the Lightning clients as well rendered the scene correctly, maximal memory used was 1610 MB that dropped to 1280 MB when rendering. Such a file should render without setting the LAA flag.

However, since I set the flag anyway, I could successfully network render a scene with one 300 MB objects and six 145 MB objects. The scene had 2550 MB, which increased to 3050 MB when enabling network render. While rendering, it increased to 3360 MB. The Lightning clients needed 2850 MB to set them up and 2520 MB to actually render.

Adding another 145 MB object crashed Bryce. If Bryce can set up network rendering and the Lightning clients have the LAA flag set, the Bryce host sets the limit of how large a scene can be rendered over the network.



```
0023:7786DFE4 ntdll.dll, RtlFreeHeap()+0063 byte(s)
0023:76F714AD kernel32.dll, HeapFree()+0020 byte(s)
0023:71424C39 MSVCR80.dll, free()+0205 byte(s)
0023:10009905 Axiom.dll, CBryceDataFile6::WriteEntry()+0357 byte(s)
0023:100093D8 Axiom.dll, CBryceDataFile6::AllocateEntry()+0152 byte(s)
0023:100302EB Axiom.dll, CSpoolMem::StorageP()+0043 byte(s)
```

The document size also determines Bryce and Lightning memory needs. Doubling the document size from 800 x 600 to 1600 x 1200 added 200 MB to the memory. I tested in normal priority and setting to high priority took another 200 MB slice of Bryce memory.

Always save your scene with a unique name <u>before</u> you attempt a network render.

October 2014/horo