Performance Monitor Dialog

From TouchDesigner 088 Wiki

Description

The Performance Monitor dialog allows you to monitor and diagnose which OPs require the most processing and which objects take the longest to draw, so you can optimize your networks. You can do this in a number of ways:

- **Each OP's Info Pop-up** - The Cook time and Cook counter of each OP can be displayed in the Info pop-up of each OP. Use the middle mouse on the icon of any operator.
- **Performance Dialog** - The Cook time, Cook counter, Viewport display times, User Interface compute times, and more can be logged in the Performance Monitor dialog. Click the Analyze button to get a snapshot of what is happening in your synth.

See also Optimize.
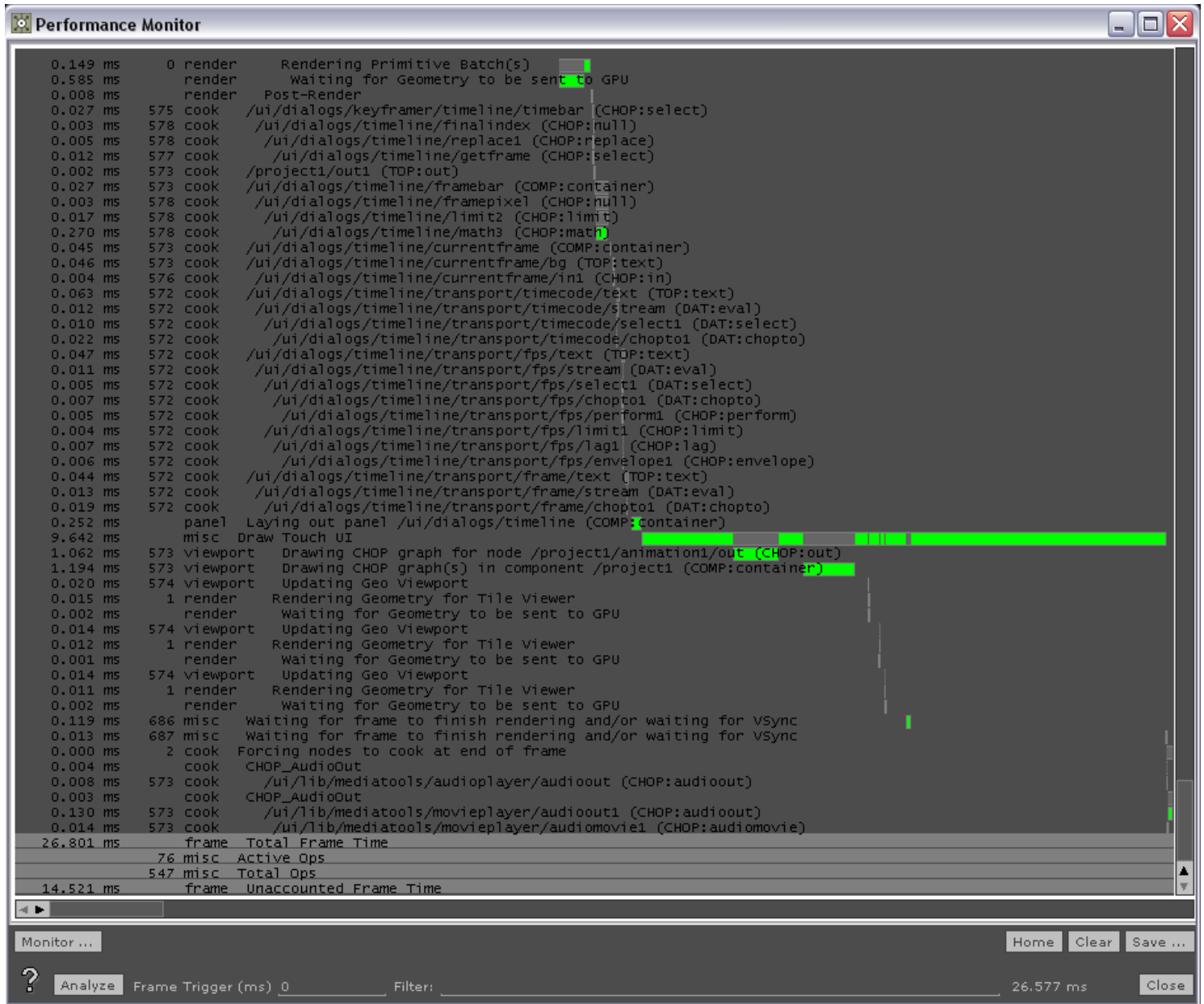
Green Overlay Bar Graph

A green bar graph per line of output gives a visual representation of the amount of time it took to process the operation. The remaining time in dark gray is the amount of time that line had to wait for other operation (Nodes waiting for their input nodes to cook for example). The bar position in X is the order in which the processing took place. For example in the above image you can see that the Lookup TOP was waiting for a bunch of other TOPs to finish cooking before it was able to cook.

Left mouse click on the Performance Monitor output window to fit the statistics graph to the performance window. Middle mouse down and drag to scale the statistics graph.

Right mouse down and drag to shift the statistics graph.

Using the Performance Monitor Dialog

To open the Performance Monitor, select Dialogs --> Performance Monitor... or use the keyboard shortcut Alt-y.

Click the Analyze button to anaylze your synth. There is a visual bar graph laid over the event info to help to identify processing time for each event. To home the graph inside the window, left-click anywhere in the output window.

Settings and Options

Analyze

Press the *Analyze* button to takes a snapshot of a few frames and outputs the event times. The messages between two consecutive frames are caught in memory first, then reported after allowing for accurate analysis of all the events pertaining to a full frame of animation. Often outputting a continuous stream with **Enable Output** will slow your synth down and effect the accuracy of the reported times. Using the Analyze method to generate a performance log as it pauses the output and therefore doesn't interfere as much with the overall performance of TouchDesigner.

Frame Trigger (ms)

Use Frame Trigger to catch events that take place over a set amount of time. A useful application of Frame Trigger is if you notice that only particular frames seem to take a long time to cook (TouchDesigner seems to freeze up at certain frames), you can use the Frame Trigger to only list cooking times for those frames that take longer than the Frame Trigger length to process. For example, while TouchDesigner is playing forward, type 100 in the Frame Trigger and then press Analyze. The next time a frame takes longer than 100 ms to cook, the performance monitor will report a list of monitored items. It is probably best to sort the list by Adjusted Length in which case the longest cook will appear at the bottom of the list.

Filter

The filter field lets you scope specific items in the performance monitor's output. For example, Filter *CHOP* will limit the output to all CHOPs.

Home

Press this button after the cook times are logged. It will resize the cook time bar graph visualization to the size of the Performance Monitor window. You can also left-click anywhere in the output window to home the graph.

Clear

Clears the display window.

NOTE: If you don't want to lose the info, save it first.

Save...

Saves the current performance log as a text (.txt) file. If you click Save, it displays a File Save dialog to save contents of the Scrolling Text Field.

Close

Closes the Performance dialog. Performance monitoring will still continue as long as the Enable button is still checked.

Performance Log Information

Monitor Items

The Monitor window allows you to select type of performance events will be monitored and therefore which information is logged.. If you want to see everything, check all the boxes. The following options allow you to control the scope of what is logged, narrowing your search if necessary:

- **OP Cook Time** - The time it takes a specific operation to cook.
- **OP Export Time** - The time it takes to export a channel.
- **Long CHOPs** - Reports CHOPs that are greater than 100 samples long.
- **Object Draw Time** - The time it takes to draw an object.
- **Custom Panel Time** - The time it takes to draw the control panel.
- **Port Draw Time** - The time it takes to draw the 3D viewport.
- **Movie Library Time** - The time it takes to draw QuickTime Image/Movie panels.
- **Draw Channels Time** - The time it takes to draw the CHOP channel viewer.
- **Midi Handling** - The time it takes to send and receive MIDI.
- **Graphics Driver Time** - The time that is spent making calls to the graphics driver (or waiting for the graphics driver to do some work).
- **Frame Time** - Reports time between frames and/or playback rate.
- **Errors** - Error messages from operators are logged.

Listed/Logged Information

Use this information to track which OPs use the most processing time, and to understand where the performance bottlenecks are.

- **Order** - The order in which OPs are cooked.
- **OP Cook Time** - Per operation cook time, and the cook time of all the OPs it calls.
- **OP Cook Count** - A separate counter for each OP. The number of times the OP has been cooked.
- **Viewport Display Time** - Count (Individual display time of each Object/SOP in the Viewport).
- **Frame Length / Frequency** - The amount of time it took to cook the entire frame.
- **Gadget Drawing** - The time it takes to draw gadget panels.
- **draw** - The time it takes to draw an object.
- **cook** - The time it takes an operator to cook.
- **port** - The time it takes a viewport or graph to draw.
- **drawlist** - Time it takes to reorder objects in the display list.
- **custom** - The time it takes to draw a control panel.
- **midi** - The time it takes to check MIDI events.
- **export** - The time it takes a CHOP channel to export to a parameter.
- **tessellate** - The amount of time it takes geometry to tessellate into polygons.

TIP: Watching networks cook is useful for understanding how to design networks in ways to minimize cooks. Always place animating operators (operators whose output wires are animating) near the leaves (ends) of the network to avoid unnecessary cooking by operators that are not changing. If an operator is changing every frame, the entire network after may also be cooking every frame.

Understanding Results

When looking at the cook times for nodes like SOPs, DATs and CHOPs, what you see is what you get. The amount of time you see on the performance monitor is the amount of time they took to execute. Reducing their workload (less points or less channels) will result in a smaller cook time.

COMPs generally have a very small cook time, so there is no way to speed up their cooking, other than avoiding the cook altogether.

TOPs are where things get tricky. The actual work for most TOPs is executed on the GPU, and is not shown in the Performance Monitor. An exception to this are TOPs that upload or download images from the GPU, like Touch In, Movie File In, Touch Out, Movie Out, etc. For the majority of TOPs, their cook time is simply the time it took the TOP to tell the GPU what work it wants it to do (e.g. it tells the GPU to execute the Displace operation on input image X and input image Y). The TOP will finish cooking long before the GPU has finished executing its task. This is called asynchronous execution. Currently there is no way to find out exactly how much time it takes the GPU to execute an operation. Consequently the Performance Monitor is currently not particularly useful to help optimize GPU usage. To optimize GPU usage refer to the Optimize article.

After reading the Cook article, you can further figure out 'why' a node is cooking by looking the graph and taking note of which OP's bars appear within which other OP's bars. In a batch of nodes whose bars are within each other, the nodes that are higher in the list are requesting the nodes lower in the list to cook. The nodes that are lower in the list are the reason a node above it cooks.

Understanding Inconsistent Results

When using the Performance Monitor on a system with a heavy load you may get some confusing results. You may get very large cooks in nodes that shouldn't have expensive cooks like Null, In and Out OPs. Other nodes that shouldn't have expensive cooks are TOPs (except for ones that upload/download data from the GPU as mentioned above). You also may get nodes that have a very large cook one frame, and then when you analyze again you get totally different values. The large cook may have moved to a different and unrelated node.

When this happens, usually it means your CPU is overworked. What is likely happening is TouchDesigner starts to cook a node, but then the CPU goes elsewhere and executes code for a totally different app. When TouchDesigner regains the CPU and finishes cooking the node it has no way of knowing that it only spent 0.1 ms of the last 2 ms actually cooking the node (for example). This is often why you get random large cooks for nodes that shouldn't take so long. Check to see if your CPU usage is at 100% (under the Windows Task Manager).

Another reason for long TOP cook times or long Draw Touch UI entries is an overworked GPU. The graphics driver may stall TouchDesigner while it waits for the GPU to catch up and finish all of the work it has been asked to execute.

Draw Touch UI

A common question is what is the 'Draw Touch UI' entry. This entry is the amount of CPU time it takes TouchDesigner to draw the UI (panels and networks, not geometry etc.). As mentioned above this entry may get very long if the GPU is overworked.

See Also

- Optimize