

# Validating XPU Renders with RIS

XPU produces renders that are predictive of RIS, however, there are features in RIS that don't exist in XPU yet.

Some of these features are the reason why your XPU images may differ slightly from RIS.

If you find yourself in the situation where you need to validate the results you see in XPU using RIS, use the hints below to make your RIS renders achieve that comparison. However, if you tweak these knobs, be aware you may be artificially slowing down RIS, so proceed with caution.

- [Sample count](#) - XPU does not yet support adaptive sampling, so you must be sure to disable it
  - minsamples = maxsamples - To ensure that XPU and RIS are sampling at similar rates, you need to make sure that you have disabled adaptive sampling via the easiest mechanism: setting the minsamples parameter value equal to the maxsamples parameter value on PxrPathTracer.
- [Other Integrator controls](#).
  - numLightSamples, numBxdfSamples, numIndirectSamples (and their companions when you're in "manual" mode). XPU does not support these. XPU takes only one sample for each of numBxdfSamples and numIndirectSamples. If your settings are higher than 1 for any of these, you will be giving an unfair advantage to XPU because RIS will be doing extra work. XPU does not have support for numLightSamples either, but has an internal heuristic about how many samples to take. So if you have more than one light in your scene, RIS and XPU will not be doing the same amount of work.
  - clampDepth, clampLuminance. XPU does not yet support illumination clamping, so be sure to set clampDepth and clampLuminance to a sufficiently high value so that RIS does not clamp any values that XPU would be splatting to the screen.
- Eliminate any usage of "shot" lighting features, such as light linking, light filters, or the light controls that give more nuance.
- Be aware of the other [XPU limitations](#) and eliminate as many of them as possible from your test.