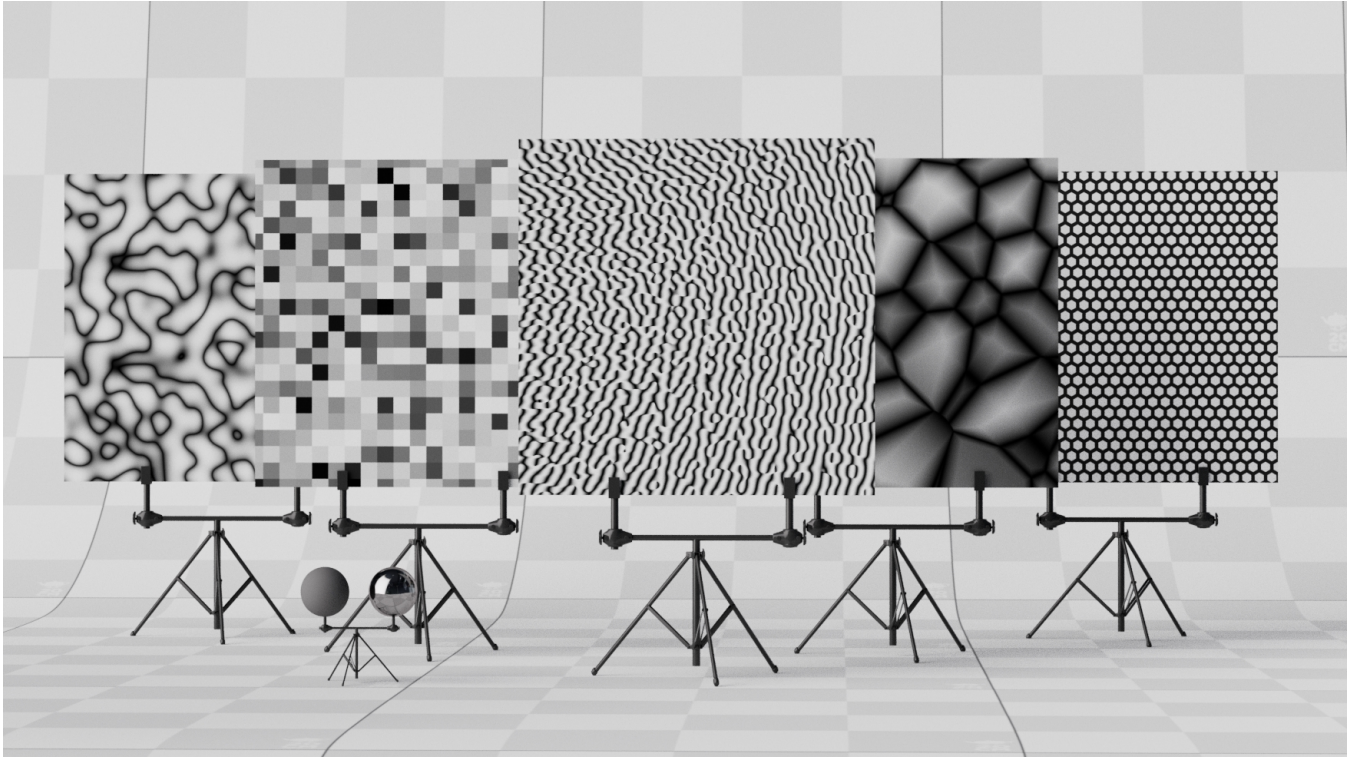


# Patterns



While Materials control the appearance of an object, patterns control the detail by varying the parameters of the material's Bxdf across a surface. This is similar to the way texture artists use textures to control the look of a material. RenderMan ships with a set of patterns that allow texturing, color correction, bumping, and various utility shading systems.

The patterns can be divided into the following categories:

---

## Image Texture

Drive patterns with texture images.

- [PxrMultiTexture](#)
- [PxrProjectionLayer](#)
- [PxrPtexture](#)
- [PxrTexture](#)

## Procedural Texture

Procedurally create noise and patterns.

- [PxrChecker](#)
- [PxrCurvature](#)
- [PxrFractal](#)
- [PxrVoronoise](#)
- [PxrWorley](#)
- [PxrPhasorNoise](#)

## Manifolds

Manifolds tell the renderer how to apply patterns on a surface. This controls features like repetition, placement, and size.

- [PxrManifold2D](#)
- [PxrBumpManifold2D](#)
- [PxrManifold3D](#)
- [PxrProjector](#)
- [PxrRandomTextureManifold](#)

- [PxrRoundCube](#)
- [PxrTileManifold](#)
- [PxrHexTileManifold](#)

## Color

Create or modify colors in the shading network.

- [PxrBlackBody](#)
- [PxrBlend](#)
- [PxrClamp](#)
- [PxrColorCorrect](#)
- [PxrColorSpace](#)
- [PxrExposure](#)
- [PxrGamma](#)
- [PxrHairColor](#)
- [PxrHSL](#)
- [PxrInvert](#)
- [PxrLayeredBlend](#)
- [PxrMix](#)
- [PxrProjectionStack](#)
- [PxrRamp](#)
- [PxrRemap](#)
- [PxrThinFilm](#)
- [PxrThreshold](#)
- [PxrVary](#)

## Bump

Create low amplitude embossing effects.

- [PxrBump](#)
- [PxrBumpMixer](#)
- [PxrNormalMap](#)
- [PxrFlakes](#)
- [aaOceanPrmanShader](#)

## Geometry Nodes

You can use these nodes to calculate geometric properties and vectors.

- [PxDirt](#)
- [PxrDispScalarLayer](#)
- [PxrDispTransform](#)
- [PxrDispVectorLayer](#)
- [PxrDot](#)
- [PxCross](#)
- [PxrFacingRatio](#)
- [PxrTangentField](#)

## Script

- [OSL Patterns](#)

## Utility

- [PxrAttribute](#)
- [PxrMatteID](#)
- [PxrPrimvar](#)
- [PxrVariable](#)
- [PxrToFloat](#)
- [PxrToFloat3](#)
- [PxrTee](#)
- [PxrShadedSide](#)

## Baking

These passthrough patterns are designed to allow users to bake arbitrary pattern networks.

- [PxrBakeTexture](#)
- [PxrBakePointCloud](#)

## Guidance for Writing Your Own Patterns

You do not need to be limited to the patterns that RenderMan provides. If you follow the guidance in [OSL Patterns](#) you can build your own patterns. XPU only runs OSL patterns, while RIS runs both C++ and OSL patterns.