

Primitive Variables



These may also be referred to as "prototype attributes" for clarity versus [Instance Attributes](#) which may be varied per instance.

Common

Name	Type	Default	Description
identifier:object	string	empty	Object shape name.
stats:prototypeIdentifier	string	empty	Identifier string used by the stats system when referring to this master.

Shading

Name	Type	Default	Description
derivatives:extrapolate	integer	1	Extrapolated smooth normals across grid boundaries.
displacement:ignorereferenceinstance	int	0	Indicates if displacement shading should ignore properties of the reference instance.
displacementbound:CoordinateSystem	string	object	The name of the coordinate system that the displacement bound is measured in.
displacementbound:offscreen	int	0	Apply displacementbound to offscreen geometry when dicing. Geometry that is entirely offscreen, but displaces into view may be prone to undertessellation because of the offscreen dicing strategy. This can be corrected by enabling this control and setting a reasonable displacementbound. Doing so will incur a performance penalty commensurate with the size of the displacement bound.
displacementbound:sphere	float	0	Maximum displacement distance used to compute object bounds. This value should be as tight as possible.
Ri:Orientation	string	outside	Geometry winding order that determines surface normal. This attribute effects lighting and displacement. Supported values: inside, outside.
trace:autobias	integer	1	Enable automatic raytrace bias.
trace:bias	float	0.01	Manual raytrace bias in object space. Small values should be used to avoid artifacts such as shadow acne.
trace:displacements	integer	1	Enable displacement shading.
trace:sssautobias	integer	1	Enable automatic raytrace bias for subsurface scattering rays.
trace:sssbias	float	1e-4f	Manual raytrace bias (in object space) for subsurface scattering rays. Small values should be used to avoid artifacts such as surface acne.

Dicing

Name	Type	Default	Description
dice:multipolygonlength	float	1	Micropolygon distance in raster space for "instanceprojection" dicing. Values are expressed in pixel size.
dice:offscreenstrategy	string	viewfrustumdistance	Dicing method of objects outside the viewing frustum. Supported values: viewfrustumdistance, worlddistance, objectdistance, planarprojection.
dice:rasterorient	integer	1	Turning this off enables non-oriented dicing, a mode of dicing that computes micropolygon sizes using non-oriented raster space rather than screen-aligned raster space. Non-oriented dicing is useful for displacement around edges where projected micropolygon distance can be arbitrarily small.
dice:referencecamera	string	empty	Specify the camera used for dicing. If no reference camera is specified, RenderMan will use the primary camera.

dice:referenceinstance	<i>string</i>	<i>empty</i>	Specify the reference instance used for dicing and displacement shading. The reference instance is used to drive dicing and displacement based on its position, scale, user attributes, and scoped coordinate systems. Reference instances are specified by its identifier:name attribute (instance name). If no reference instance is specified, RenderMan will automatically pick the nearest instance inside the primary camera's view frustum.
dice:strategy	<i>string</i>	instanceprojection	Dicing method of objects within the viewing frustum. Supported values: instanceprojection, worlddistance, objectdistance.
dice:worlddistance	<i>float</i>	-1	Micropolygon distance in world space for "worlddistance" dicing or object space for "objectdistance" dicing.
Ri:GeometricApproximationFocusFactor	<i>float</i>	0	Allows the renderer to use more coarse dicing for blurry objects due to depth of field.
Ri:GeometricApproximationMotionFactor	<i>float</i>	0	Allows the renderer to use more coarse dicing for blurry objects due to motion blur.

Points

Name	Type	Default	Description
falloffpower	<i>float</i>	0	For use with points, if not supplied, or set to zero, the points will have a hard edge. A value of 1 is a "reasonable" value that emulates the usual cosine based falloff; this will likely be the goto value for most people doing volumetric particle effects. Values between 0 and 1 makes the falloff faster, eroding the point faster - point has "less presence". Values higher than 1 (up to infinity) makes the falloff slower to the point of being non-existent.

Curves

Name	Type	Default	Description
curve:opacitysamples	<i>int</i>	0	number of opacity/presence shading samples along a curve for opacity caching, if equal to zero then the value is determined using the dicing oracle and the length of each curve.
curve:widthaffectscurvature	<i>int</i>	1	When true the curve width of round curves is taken into account in the computation of the tube curvature, otherwise only the curvature along the curve is. This control does not affect curve ribbons. Value range: true/false.

Volume

Name	Type	Default	Description
dice:minlength	<i>float</i>	-1	Volume minimum dice length. Negative indicates to automatically compute this value.
dice:minlengthspace	<i>string</i>	<i>empty</i>	Coordinate space of dice:minlength.
Ri:Bound	<i>float[6]</i>	0 0 0 0 0 0	Volume bounds.
volume:temporalmethod	<i>integer</i>	0	Method of generating temporal data for volume rendering. Supported values: Eulerian (0), Reves (1).

SubdivisionMesh

Name	Type	Default	Description
dice:pretessellate	<i>integer</i>	1	Pre-tessellate subdivision geometry to polygons.
dice:watertight	<i>integer</i>	0	Tessellate geometry with no holes. Watertight geometry requires less raytrace bias.
shade:faceset	<i>integer[n]</i>	<i>empty</i>	Active geometry face indexes.
stitchbound:CoordinateSystem	<i>string</i>	<i>empty</i>	
stitchbound:sphere	<i>float</i>	0	

NuPatch

Name	Type	Default	Description
------	------	---------	-------------

trimcurve:sense	<i>string</i>	inside	Supported values: inside, outside.
-----------------	---------------	--------	------------------------------------

PolygonMesh

Name	Type	Default	Description
polygon:concave	<i>integer</i>	1	Allow concave polygons.
polygon:smoothdisplacement	<i>integer</i>	0	Output smoothed (per-vertex) normals as Ndsp primvar, if polygon:smoothnormals hasn't already inserted smooth normals.
polygon:smoothnormals	<i>integer</i>	0	Smooth (per-vertex) normals if not provided.

Procedural

Name	Type	Default	Description
procedural:immediatesubdivide	<i>integer</i>	0	
procedural:reentrant	<i>integer</i>	0	