Table 14.2 lists each predefined global variable available to surface shaders, together with its data type, storage class and a summary of its meaning. Different sets of global variables are available to other shader types. They appear in Table 14.3.

Type	Name	Storage Class	Purpose
color	*Cs*	**varying/uniform**	Surface color (input)
color	*Os*	**varying/uniform**	Surface opacity (input)
point	*P*	**varying**	Surface position
point	*dPdu*	**varying**	Change in position with *u*
point	*dPdv*	**varying**	Change in position with *v*
point	*N*	**varying**	Surface shading normal
point	*Ng*	**varying/uniform**	Surface geometric normal
float	*u,v*	**varying**	Surface parameters
float	*du,dv*	**varying/uniform**	Change in *u,v* across element
float	*s,t*	**varying**	Surface texture coordinates
color	*L*	**varying/uniform**	Direction from surface to light source
color	*Cl*	**varying/uniform**	Light color
point	*I*	**varying**	Direction of ray impinging on surface point (often from camera)
color	*Ci*	**varying**	Color of light from surface (output)
color	*Oi*	**varying**	Opacity of surface (output)
point	*E*	**uniform**	Position of the camera

Table 14.2 *Global Variables Available to Surface Shaders*

Surface color and transparency

Cs and *Os* represent the current surface color and opacity, respectively, as declared in **RiColor**() and **RiOpacity**() and bound to the surface being shaded when it was created.

Cs and *Os* are used as filter values. The color of reflected light from a surface with surface color *Cs* under incident light with color *Cl* is often taken to be *Cl* **Cs*. In other words, each component of *Cs* scales the corresponding component of the incoming light according to the absorption of the surface. *Os* has the same effect on light passing *through* the surface. Normally, every component of *Cs* and *Os* lies in the range [0,1].

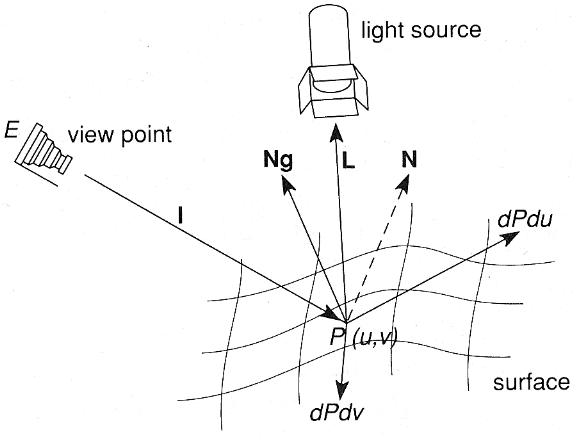

Figure 14.1 *Surface position, normal, and parametric derivatives*

Surface position and change

The **point** value P represents the position of the point being shaded in world space, and Ng is the **geometric normal vector**, perpendicular to the surface, at that point. The **shading normal vector** N is by default equal to Ng, but may be different for shading purposes. If a displacement shader changes the surface normal, it usually works on N and leaves Ng alone.

Parameter space

The floating-point values u and v give the position of the current point on the current surface in parameter space. The **points** *dPdu* and *dPdv* are parametric derivatives, giving the derivative of surface position P with respect to u and v, respectively. The surface normal Ng is defined to be the cross-product of these two vectors. u and v always range between exactly 0 and 1 on all surfaces except polygons.

Figure 14.1 illustrates P, Ng, *dPdu* and *dPdv*. The normal vector Ng is the cross product of *dPdu* and *dPdv* by definition.

Texture space

The floating-point values s and t give the texture-space coordinates of the current point on the surface. They may be used to