

PST-marble Commands and Parameters

Colors

RGB colors can be specified in three formats:

```
[ 0.906 0.8 0.608 ]
```

Red, green, and blue color components between 0 and 1 in square brackets.

```
[ 231 204 155 ]
```

Red, green, and blue color components between 0 and 255 in square brackets.

```
(e7cc9b)
```

Red, green, and blue (**RrGgBb**) hexadecimal color components between 00 and FF (or ff) in parentheses.

In the command arguments [*rgb* ...] indicates a bracketed sequence of colors. For example:

```
[(c28847) [231 204 155] [0.635 0.008 0.094]]
```

```
rgb  $\gamma$  tint
```

Returns the *rgb* color as modified by γ . $0 < \gamma < 1$ darkens the color; $1 < \gamma$ lightens the color; and $\gamma = 1$ leaves it unchanged.

```
rgb  $\gamma$  shade
```

Returns the *rgb* color as modified by γ . $0 < \gamma < 1$ lightens the color; $1 < \gamma$ darkens the color; and $\gamma = 1$ leaves it unchanged.

```
rgb  $\zeta$  edgy-color
```

Returns the *rgb* color flagged so that in raster rendering the boundary of each drop of that color is lightened while its center is darkened. Where a is the point's initial distance from the drop center and r is the drop's initial radius, the effective $\gamma = \exp(\zeta a^2/r^2) (\exp(\zeta) - 1)/(\zeta \exp(\zeta))$. When $\zeta = 0$, $\gamma = 1$ and the drop has uniformly tone.

Parameters

```
\psMarble[ parameter-assignment, ..., parameter-assignment ] (width, height)
```

```
\psMarble[ parameter-assignment, ..., parameter-assignment ] (x-, y-) (x+, y+)
```

The comma separated parameter assignments are part of the `\psMarble` command. In the list below, the default value for each parameter is shown to the right of the parameter name. Note that the values assigned to `background=`, `colors=`, `seed=`, `actions=`, and `spractions=` must be enclosed in curly braces {}.

```
background= {[0 0 0]}
```

Specifies the color for regions where paint has not been dropped (or moved to).

```
bckg= true
```

When `bckg=false` the background color is not shown.

```
paper= {[1 1 1]}
```

Specifies whether tinting (brightening) or shading (darkening) occurs when shading due to paper movement occurs from the `jiggle-shade` or `wriggle-shade` commands. Default is white (brightening).

```
colors= {[0.275 0.569 0.796] [0.965 0.882 0.302] [0.176 0.353 0.129] [0.635 0.008 0.094] [0.078 0.165 0.518]  
[0.824 0.592 0.031] [0.059 0.522 0.392] [0.816 0.333 0.475] [0.365 0.153 0.435] [0.624 0.588 0.439]}
```

Specifies a color sequence accessible in paint-dropping commands as `colors`.

```
drawcontours= false
```

When `drawcontours=true` paint contours are drawn with lines; when `drawcontours=false` contours are filled;

```
oversample= 0
```

When `oversample=0` a resolution-independent image is produced using contour-rendering. When the number of drops gets too large (> 150) triangular artifacts start to appear. Changing to `oversample=1` employs raster-rendering to more quickly compute each image pixel individually. When `oversample=2` the rendering takes four times as long, but each pixel is the averaged over its four quarters, producing an image nearly as good as `oversample=0`. When `oversample` is between 0 and 1, the rendering is on a coarser grid than `oversample=1`, speeding image production.

```
overscan= 1
```

When the `overscan` value is greater than 1, proportionally more image (outside of the specified area) is shown, and the specified area is outlined with a dashed rectangular border. This is a utility for developing marblings.

```
seed= {Mathematical Marbling}
```

Specifies the random seed used for `normal-drops`, `uniform-drops`, `normal-spray`, and `uniform-spray` commands. Changing the `seed` value changes the positions of all drops from the `normal-drops`, `uniform-drops`, `normal-spray`, and `uniform-spray` commands.

```
viscosity= 1000
```

Specifies the overall kinetic viscosity of the virtual tank fluid. Its units are mm^2/s ; the default value of 1000, which is 1000 times more viscous than water, is a typical value for marbling. Increasing **viscosity** reduces the fluid movement far from the tines.

actions= {0 0 36 colors 35 concentric-rings}

Specifies the sequence of marbling commands to perform. The default is a single command dropping 35 colors in the **colors** sequence. The available commands are listed below.

shadings= {}

When **oversample**> 0, specifies the sequence of shading commands to perform after all marbling **actions** and **spractions** are performed.

spractions= {}

Specifies the sequence of spray commands to perform. Spray commands are performed after marbling and before shading.

Dropping Paint

$x y R_d \text{rgb}$ **drop**

Places a drop of color **rgb** and radius R_d centered at location x, y .

$x y R_i [\text{rgb} \dots] n$ **concentric-rings**

Places n rings in color sequence **[rgb ...]** centered at location x, y , each ring having thickness R_i .

$x y \theta [R \dots] [\text{rgb} \dots] R_d$ **line-drops**

Places drops of colors **[rgb ...]** (in sequence) of radius R_d in a line through x, y at θ degrees clockwise from upward at distances **[R ...]** from x, y .

$x y [\Omega_{\perp} \dots] [\Omega_{\parallel} \dots] \theta [\text{rgb} \dots] R_d$ **serpentine-drops**

Places drops of colors **[rgb ...]** of radius R_d in a serpentine pattern (starting lower left to right; right to left; left to right...) at offsets $\Omega_{\perp} \times \Omega_{\parallel}$ centered at location x, y and rotated by θ degrees clockwise from upward. The sequences Ω_{\perp} and Ω_{\parallel} control the serpentine sequence.

$x y R \theta S \delta [\text{rgb} \dots] n R_d$ **coil-drops**

Places n drops of colors **[rgb ...]** (in sequence) of radius R_d in an arc or spiral centered at x, y starting at radius R and θ degrees clockwise from upward, moving S along the arc and incrementing the arc radius by δ after each drop.

$x y L_{\perp} L_{\parallel} \theta [\text{rgb} \dots] n R_d$ **normal-drops**

Places n drops of colors **[rgb ...]** of radius R_d randomly in a circular or elliptical disk centered at x, y having diameters L_{\perp} and L_{\parallel} respectively perpendicular and parallel to θ degrees clockwise from upward. For a circular disk ($R = L_{\parallel}/2 = L_{\perp}/2$), 63% of drops are within radius R , 87% of drops are within $R\sqrt{2}$, and 98% of drops are within radius $2R$.

$x y L_{\perp} L_{\parallel} \theta [\text{rgb} \dots] n R_d$ **uniform-drops**

Places n drops of colors **[rgb ...]** of radius R_d randomly in a L_{\perp} by L_{\parallel} rectangle centered at location x, y and rotated by θ degrees clockwise from upward.

Deformations

$\theta [R \dots] V S D$ **rake**

Pulls tines of diameter D at θ degrees from the y-axis through the virtual tank at velocity V , moving fluid on the tine path a distance S . The tine paths are spaced **[R ...]** from the tank center at their nearest points.

$x_b y_b x_e y_e V D$ **stylus**

Pulls a single tine of diameter D from x_b, y_b to x_e, y_e at velocity V .

$x y [R \dots] \omega \theta D$ **stir**

Pulls tines of diameter D in circular tracks of radii **[R ...]** (positive R is clockwise) around location x, y at angular velocity ω . The maximum angle through which fluid is moved is θ degrees.

$x y \Gamma t$ **vortex**

Rotates fluid clockwise around location x, y as would result from an impulse of circulation Γ after time t . At small t the rotational shear is concentrated close to the center. As time passes the shear propagates outward.

$\theta \lambda \Omega A B$ **jiggle**

Consider the tank as split into parallel infinitesimal strips perpendicular to direction θ (degrees clockwise from upward). Where a is the distance in the θ direction, the strips are displaced along their lengths by $0.5B \cos(360(a + \Omega)/\lambda)$ and in the θ direction by $0.5A \sin(360(a + \Omega)/\lambda)$. Displacements in the θ direction squeeze and expand distances between the strips. To prevent tears and tunnels $|\pi A| < |\lambda|$. For area-preserving **jiggles**, use $A = 0$.

$x y \lambda A B$ **wriggle**

wriggle is to **jiggle** as **stir** is to **rake**. Consider the tank as split into concentric rings around x, y . Where r is the radial distance from x, y , rings are rotated by $0.5B \cos(360r/\lambda)$, and expanded and contracted $0.5A \sin(360r/\lambda)$. To prevent overlap $|\pi A| < |\lambda|$. There is no offset parameter. When $A/\lambda > 0$, maximum compression is at r equal to odd multiples of 0.5λ ; otherwise maximum compression is at integer multiples of λ .

θ Ω shift

Shifts tank by distance Ω in direction θ degrees clockwise from upward.

x y θ turn

Turns tank around x, y by θ degrees clockwise.

[n S Ω tines]

The **tines** command and its arguments are replaced by a sequence of n numbers. The difference between adjacent numbers is S and the center number is Ω when n is odd and $S/2 - \Omega$ when n is even.

Spray Actions

Spray actions are intended for drops small enough that they don't noticeably move paint contours. The radii of spray droplets are the cube roots of log-normal distributed values with mean R_d .

x y L_{\perp} L_{\parallel} θ [rgb ...] n R_d normal-spray

Places n drops of colors [rgb ...] of radius R_d randomly in a circular or elliptical disk centered at x, y having diameters L_{\perp} and L_{\parallel} respectively perpendicular and parallel to θ degrees clockwise from upward. For a circular disk ($R = L_{\parallel}/2 = L_{\perp}/2$), 63% of drops are within radius R , 87% of drops are within $R\sqrt{2}$, and 98% of drops are within radius $2R$.

x y L_{\perp} L_{\parallel} θ [rgb ...] n R_d uniform-spray

Places n drops of colors [rgb ...] of radius R_d randomly in a L_{\perp} by L_{\parallel} rectangle centered at location x, y and rotated by θ degrees clockwise from upward.

Shadings

Shadings commands simulate the lightening and darkening of paint transferred to paper caused by pulling the paper from the bath at uneven rates. Shading is always performed for **spractions**, but only when **oversample** > 0 for **actions**. Shading commands are placed within the braces of the **shadings**={} parameter.

θ λ Ω A_s jiggle-shade

Applies darkening and lightening resulting from the squeezing and expansion of a **jiggle** command sharing its first four arguments: " **θ λ Ω A B jiggle**". A_s does not need to equal A from the **jiggle** command. When A_s is closer to zero, shading will be softer; when A_s is further from zero, shading will be darker. As with A in the **jiggle** command, realistic shading requires $|\pi A_s| < |\lambda|$.

x y λ Ω A_s wriggle-shade

Applies darkening and lightening resulting from the squeezing and expansion of a **wriggle** command sharing its first three arguments. Unlike **wriggle**, **wriggle-shade** takes an offset argument Ω . A_s does not need to equal A from the **wriggle** command. When A_s is closer to zero, shading will be softer; when A_s is further from zero, shading will be darker. As with A in the **wriggle** command, realistic shading requires $|\pi A_s| < |\lambda|$. When $A/\lambda > 0$ and $\Omega = 0$, the darkest rings are at r equal to odd multiples of 0.5λ ; otherwise the darkest rings are at integer multiples of λ and there is a dark spot at x, y .