Looping

Loop moving points on curve using CurveU:

```
int loop_frames = chi("loop_frame");
float fps = 29.97;
float loop_time = loop_frames / fps;



float t = @Time / loop_time;
t -= floor(t);

float ping = abs(2 * t - 1);
float speed_mult = chf("speed_mult");
ping *= speed_mult;

float exponent = chf("ease_exponent");
float slow_ping = pow(ping, exponent);

float u = f@curveu + slow_ping;
u -= floor(u);

int prim = i@class;
vector pos = primuv(1, "P", prim, set(u, 0, 0));
@P = pos;
```

> Add a resample node with curveu, also to control amount of points

---

Loop points in Y:

```
int loop_frames = chi("loop_frames");
float fps = 29.97;
float loop_time = loop_frames / fps;
```

```
float t = @Time / loop_time;
t -= floor(t);


float offset = frac(t + rand(@ptnum)); // unique phase per point


float min_y = chf("min_y");
float max_y = chf("max_y");
float range = max_y - min_y;


@P.y = min_y + offset * range;
```

Loop on CurveU (nonPingPong):

1st attribute wrangle set to points after the point scatter:

```
vector uvw;
int prim;
float dist = xyzdist(1, @P, prim, uvw);
f@curveu = uvw.x;
i@class = prim;
```

2nd attrib wrangle set to points one the 1st input as the points and the 2nd to the curve with curevu from resample:

```
int loop_frames = chi("loop_frames");
float fps = 29.97;
float loop_time = loop_frames / fps;
//get time
float t = @Time / loop_time;
t -= floor(t);

float speed = chf("speed");

// offset
```

```
float base_u = f@curveu;
float offset_u = t * speed;

float u = base_u + offset_u;
u -= floor(u); // wrap it around

int prim = i@class;
vector pos = primuv(1, "P", prim, set(u, 0, 0));
@P = pos;
```

Revision #4
Created 2025-05-27 15:39:49 UTC by Anthony
Updated 2025-05-29 18:38:44 UTC by Anthony