

Система частиц.  
Простейшие эффекты.

# Система частиц (партиклы). Что это?

- Визуальные эффекты, создающиеся при помощи большого количества мелких частиц, не несущие в себе никакой физической составляющей и предназначенные только для создания оптической красоты.
- Типичные примеры: дым из дула огнестрелов, дождь, снег, бетонная крошка.

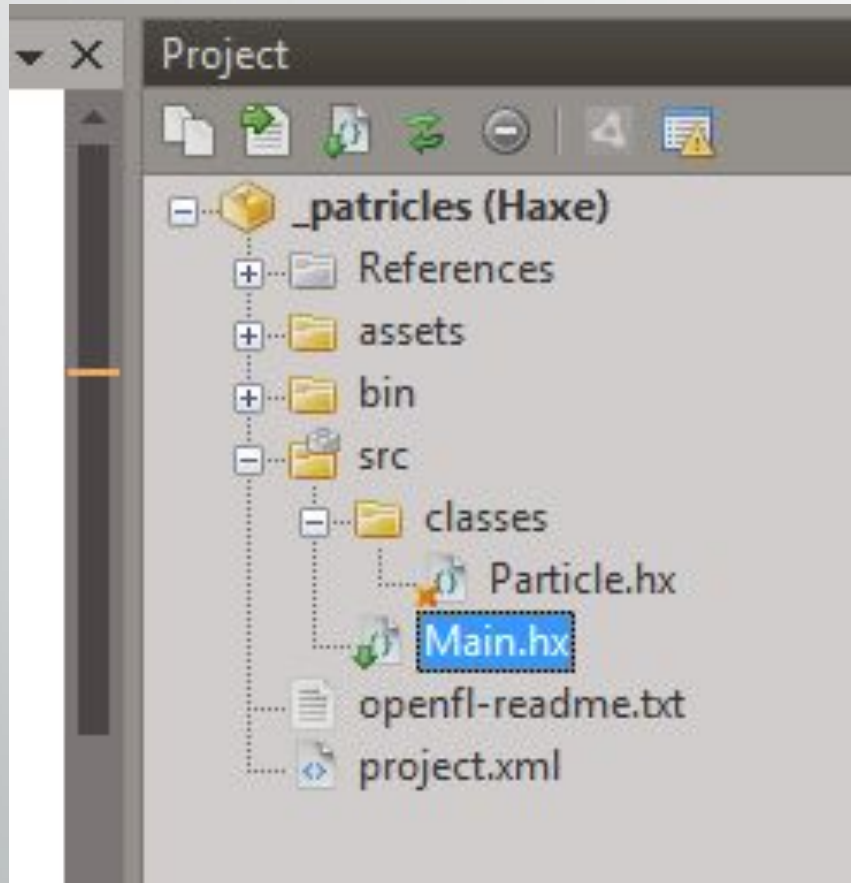
# Делаем партиклы дома, без регистрации и смс.

- Достаточно сделать класс Particle, который будет определять поведение ОДНОЙ частички.
- Внутри этого класса можно описать несколько вариантов поведения и некоторый способ выбора одного из них для каждого вида эффектов.
- Внутри класса партикла можно описать static функции, которые будут одновременно генерировать какое-то количество партиклов для создания эффекта (взрыв, например, можно сделать именно так)

# Несколько примеров эффектов на партиклах

- Данный пример притянут за уши
- Это будет единый проект с возможностью переключения между эффектами, которые тут же будут рисоваться при помощи партиклов.
- В примере показан не весь набор эффектов, которые можно нарисовать при помощи партиклов.

# Создадим класс Particle



```
Main.hx | Particle.hx*
1  package classes;
2  import openfl.display.Sprite;
3  import openfl.events.Event;
4
5  /**
6   * ...
7   * @author Zergling
8   */
9  class Particle extends Sprite
10 {
11     var TTL: Int;
12     var TTL_Start: Int;
13     var SpeedX: Float;
14     var SpeedY: Float;
15     public static var Count: Int = 0;
16
17     public function new()
18     {
19         super();
20
21         Count++;
22     }
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54 }
55
56
57
215 }
```

# Сделаем несколько штук в классе Main

```
Main.hx* Particle.hx*
13 * ...
14 * @author Zergling
15 */
16 class Main extends Sprite
17 {
18     public static var Chooser: Int;
19     public static var Keys: Array<Int> = [ 48, 49, 50, 51, 52, 53, 54, 55, 56, 57 ];
20     public static var ParticleLayer: Sprite;
21     var field: TextField;
22     var field2: TextField;
23     var format: TextFormat;
24
25     public function new() // init в старом HaXe
26     {
27         super();
28         ParticleLayer = new Sprite();
29
30         // Assets:
31         // openfl.Assets.getBitmapData("img/assetname.jpg");
32         format = new TextFormat();
33         format.font = "Arial";
34         format.color = 0xffffffff;
35         format.size = 20;
36
37         field = new TextField();
38         field.defaultTextFormat = format;
39         field.autoSize = LEFT;
40         field.y = 50;
41         field.x = 50;
42         addChild(field);
43
44         field2 = new TextField();
45         field2.defaultTextFormat = format;
46         field2.autoSize = LEFT;
47         field2.y = 100;
48         field2.x = 50;
49         addChild(field2);
50
51         addChild(ParticleLayer);
52         stage.addEventListener(KeyboardEvent.KEY_UP, onKey);
53         stage.addEventListener(Event.ENTER_FRAME, onframe_global);
54     }
55 }
```

```
Main.hx* Particle.hx*
55
56     function onframe_global(e: Event)
57     {
58         field2.text = Std.string(Particle.Count);
59     }
60
61
62
63     public function onKey(e: KeyboardEvent)
64     {
65         field.text = Std.string(e.keyCode);
66         Chooser = e.keyCode;
67         RemoveAllListeners();
68         ChooseAction();
69         // 0 = 48; 9 = 57;
70     }
71
72 }
```

# Эффект №1. Партиклы под курсором.

```
Main.hx* Particle.hx*
118
119
120 function onframe1(e: Event)
121 {
122     var particle = new Particle();
123     particle.x = this.mouseX;
124     particle.y = this.mouseY;
125     addChild (particle);
126 }
127
```

```
Main.hx* Particle.hx*
56
57
58 function start1()
59 {
60     TTL_Start = 120;
61     TTL = TTL_Start;
62
63     SpeedX = (Math.random() - 0.5);
64     SpeedY = (Math.random() - 0.5);
65     graphics.beginFill(0xffff00);
66     graphics.drawCircle(0, 0, 2);
67
68     addEventListener (Event.ENTER_FRAME, Particle_OnFrame1);
69 }
70
71 function Particle_OnFrame1(e: Event)
72 {
73     x = x + SpeedX;
74     y = y + SpeedY;
75     alpha *= 0.95;
76     TTL = TTL - 1;
77     if (TTL <= 0 || alpha < 0.005)
78     {
79         graphics.clear();
80         removeEventListener(Event.ENTER_FRAME, Particle_OnFrame1);
81         parent.removeChild(this);
82         Count--;
83     }
84 }
85
```

# Эффект №2. Взрыв по клику мыши

```
Main.hx* Particle.hx*  
87
```

```
File Edit View Search Debug Project Insert Refactor Tools Macros Synt  
Main.hx* Particle.hx*  
128  
129  
130 function onclick2(e: MouseEvent)  
131 {  
132     Particle.Blow2(this.mouseX, this.mouseY, ParticleLayer);  
133 }  
134
```

```
100  
101  
102  
103  
104  
105  
106  
107  
108
```

```
118  
119 public static function Blow2(x: Float, y: Float, layer: Sprite)  
120 {  
121     for (i in 0 ... 100)  
122     {  
123         var pr = new Particle();  
124         pr.x = x;  
125         pr.y = y;  
126         layer.addChild(pr);  
127     }  
128 }  
129
```

```
function start2()  
{  
    TTL_Start = 90;  
    TTL = TTL_Start;  
  
    var angle = Math.random() * 2 * Math.PI;  
  
    SpeedX = 3 * Math.cos(angle);  
    SpeedY = 3 * Math.sin(angle);  
  
    graphics.beginFill(0xffff00);  
    graphics.drawCircle(0, 0, 15);  
  
    addEventListener (Event.ENTER_FRAME, Particle_OnFrame2);  
}  
  
function Particle_OnFrame2(e: Event)  
{  
    x = x + SpeedX;  
    y = y + SpeedY;  
    alpha *= 0.85;  
    TTL = TTL - 1;  
    if (TTL <= 0 || alpha < 0.005)  
    {  
        graphics.clear();  
        removeEventListener(Event.ENTER_FRAME, Particle_OnFrame2);  
        parent.removeChild(this);  
        Count--;  
    }  
}
```



# Эффект №3. Мерцание звезд

```
File Edit View Search Debug Project Insert Refactor Tools Macros
Main.hx* Particle.hx*
136
137 function onframe3(e: Event)
138 {
139     var i: Int = 0;
140     for (i in 0 ... 10)
141     {
142         var particle = new Particle();
143         particle.x = Math.random() * stage.stageWidth;
144         particle.y = Math.random() * stage.stageHeight;
145         addChild(particle);
146     }
147 }
148
```

```
Main.hx* Particle.hx*
131
132
function start3()
{
    TTL_Start = 60;
    TTL = TTL_Start;
    SpeedX = (Math.random() - 0.5) / 4;
    SpeedY = (Math.random() - 0.5) / 4;

    graphics.beginFill(Std.int(Math.random() * 0xffffffff));
    graphics.drawCircle(0, 0, 2);

    addEventListener(Event.ENTER_FRAME, Particle_OnFrame3);
}

function Particle_OnFrame3(e: Event)
{
    x = x + SpeedX;
    y = y + SpeedY;
    alpha *= 0.95;
    TTL = TTL - 1;
    if (TTL <= 0 || alpha < 0.005)
    {
        graphics.clear();
        removeEventListener(Event.ENTER_FRAME, Particle_OnFrame3);
        parent.removeChild(this);
        Count--;
    }
}
150
151
152
153
154
155
156
157
158
159
```

# Эффект №4. Дождь

```
Main.hx* Particle.hx*
150
151 function onframe4(e: Event)
152 {
153     var i: Int = 0;
154     for (i in 0 ... 5)
155     {
156         var particle = new Particle();
157         particle.x = Math.random() * (stage.stageWidth + 400);
158         particle.y = - 10;
159         addChild(particle);
160     }
161 }
162

Main.hx* Particle.hx*
161
162
163

function start4()
{
    SpeedX = -4;
    SpeedY = 12;

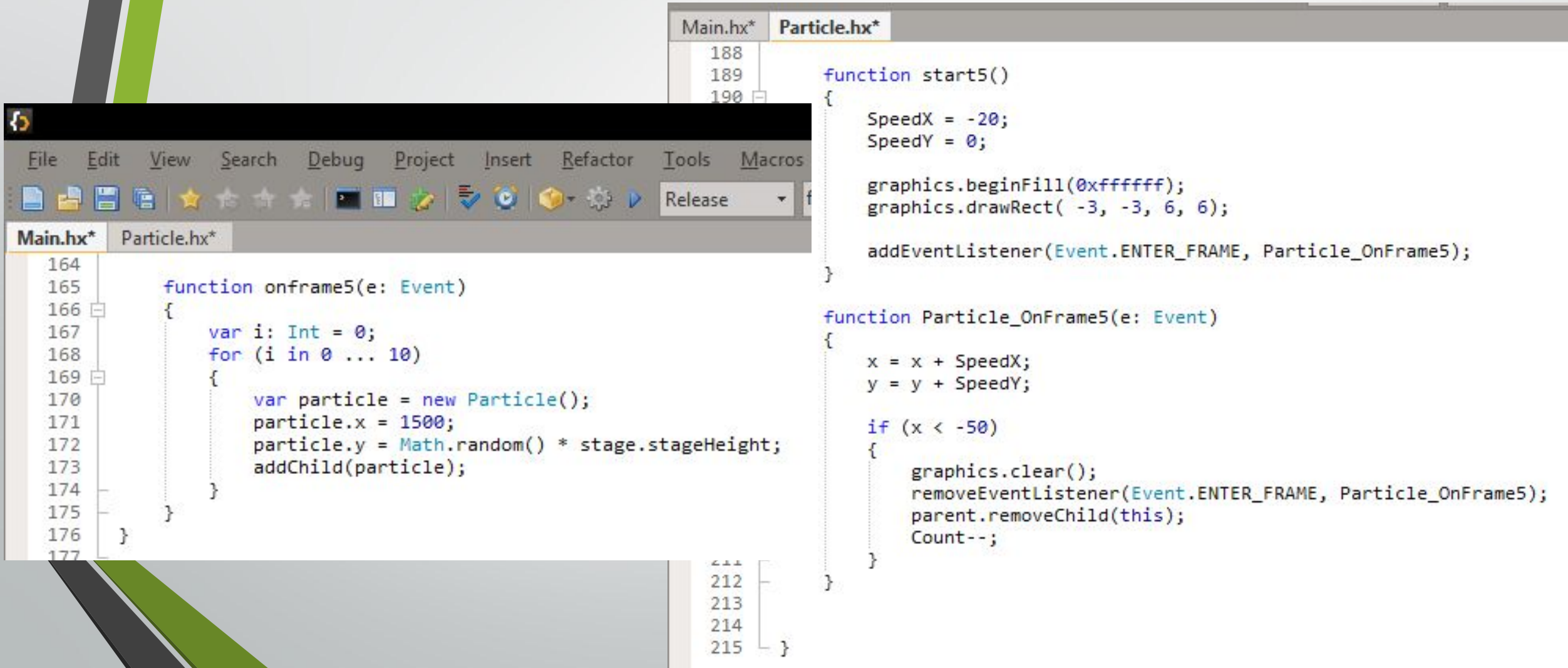
    graphics.beginFill(0x0000CC);
    graphics.drawRect( 0, -3, 2, 6);

    addEventListener(Event.ENTER_FRAME, Particle_OnFrame4);
}

function Particle_OnFrame4(e: Event)
{
    x = x + SpeedX;
    y = y + SpeedY;

    if (y > 1000)
    {
        graphics.clear();
        removeEventListener(Event.ENTER_FRAME, Particle_OnFrame4);
        parent.removeChild(this);
        Count--;
    }
}
```

# Эффект №5. Метель.



```
164  
165 function onframe5(e: Event)  
166 {  
167     var i: Int = 0;  
168     for (i in 0 ... 10)  
169     {  
170         var particle = new Particle();  
171         particle.x = 1500;  
172         particle.y = Math.random() * stage.stageHeight;  
173         addChild(particle);  
174     }  
175 }  
176 }  
177
```

```
188  
189  
190  
function start5()  
{  
    SpeedX = -20;  
    SpeedY = 0;  
  
    graphics.beginFill(0xffffffff);  
    graphics.drawRect( -3, -3, 6, 6);  
  
    addEventListener(Event.ENTER_FRAME, Particle_OnFrame5);  
}  
  
function Particle_OnFrame5(e: Event)  
{  
    x = x + SpeedX;  
    y = y + SpeedY;  
  
    if (x < -50)  
    {  
        graphics.clear();  
        removeEventListener(Event.ENTER_FRAME, Particle_OnFrame5);  
        parent.removeChild(this);  
        Count--;  
    }  
}
```

```
212 }  
213 }  
214 }  
215 }
```

# Переключение между эффектами

Main.hx\* Particle.hx\*

```
73
74 function RemoveAllListeners()
75 {
76     stage.removeEventListener(Event.ENTER_FRAME, onframe1);
77     stage.removeEventListener(MouseEvent.CLICK, onclick2);
78     stage.removeEventListener(Event.ENTER_FRAME, onframe3);
79     stage.removeEventListener(Event.ENTER_FRAME, onframe4);
80     stage.removeEventListener(Event.ENTER_FRAME, onframe5);
81 }
82
```

```
function ChooseAction()
{
    if (Chooser == Keys[1])
    {
        stage.addEventListener(Event.ENTER_FRAME, onframe1);
        return;
    }

    if (Chooser == Keys[2])
    {
        stage.addEventListener(MouseEvent.CLICK, onclick2);
        return;
    }

    if (Chooser == Keys[3])
    {
        stage.addEventListener(Event.ENTER_FRAME, onframe3);
        return;
    }

    if (Chooser == Keys[4])
    {
        stage.addEventListener(Event.ENTER_FRAME, onframe4);
        return;
    }

    if (Chooser == Keys[5])
    {
        stage.addEventListener(Event.ENTER_FRAME, onframe5);
        return;
    }
}
```

95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117

# Переключение между эффектами

```
Main.hx* Particle.hx*
8  /*
9  class Particle extends Sprite
10 {
11     var TTL: Int;
12     var TTL_Start: Int;
13     var SpeedX: Float;
14     var SpeedY: Float;
15     public static var Count: Int = 0;
```

```
16
17     public function new()
18     {
19         super();
20
21         Count++;
22
23
24         if (Main.Chooser == Main.Keys[1])
25         {
26             start1();
27             return;
28         }
29
30         if (Main.Chooser == Main.Keys[2])
31         {
32             start2();
33             return;
34         }
35
36         if (Main.Chooser == Main.Keys[3])
37         {
38             start3();
39             return;
40         }
41
42         if (Main.Chooser == Main.Keys[4])
43         {
44             start4();
45             return;
46         }
47
48         if (Main.Chooser == Main.Keys[5])
49         {
50             start5();
51             return;
52         }
53
54     }
55 }
```

# Задание

- Реализовать пример. Проверить, что все работает.
- На следующем занятии: ООП, часть 2. Наследование.