Bitmap Data

Cosmin Varlan

Bitmap Data

- Avem nevoie si de Pixel Power…
- Motive: atunci cand vrem sa transformam o poza – nu o sa o transformam vectorial si apoi sa o modificam
- Cand avem de generat multe imagini care suprapuse ar forma o animatie este mai bine sa folosim BitmapData: de exemplu la MP3 player

Bitmap Data

- Obiectele BitmapData pot fi afisate pe ecran prin intrmediul obiectelor de tip Bitmap care le contin – de ce ? [sau mai bine zis ce conditii trebuie sa indeplineasca un obiect pentru a putea fi afisat pe ecran ?]

Bitmap Data

- Conditia este… sa extinda o anumita clasa….

Bitmap Data

- DisplayObject

Inheritance DisplayObject → EventDispatcher → Object
Implements IBitmapDrawable
Subclasses AVM1Movie, Bitmap, InteractiveObject, MorphShape, Shape, StaticText, Video

- Dupa cum se observa, BitmapData nu se afla in lista subclaselor… (dar Bitmap se afla)

Bitmap Data

Cum il construiesc si afisez pe ecran:

```
private function createData():void
{
    var bd:BitmapData = new BitmapData(320,240,false,0xcccccc);
    bitmap = new Bitmap(bd);

    bitmap.bitmapData = drawInBitmapData(bitmap.bitmapData);
}
```

Bitmap Data

Cum il construiesc s

Dimensiuni, transparenta si culoare de umplere…

```
private function createData():void
{
    var bd:BitmapData = new BitmapData(320,240,false,0xcccccc);
    bitmap = new Bitmap(bd);

    bitmap.bitmapData = drawInBitmapData(bitmap.bitmapData);
}
```

Bitmap Data

Cum il construiesc s

Trebuie sa il asociez cu un obiect Bitmap

```
private function createData() : void
{
    var bd:BitmapData = new BitmapData(320,240,false,0xcccccc);
    bitmap = new Bitmap(bd);

    bitmap.bitmapData = drawInBitmapData(bitmap.bitmapData);
}
```

Bitmap Data

Cum il construiesc s

Il pot modifica printr-o functie

```
private function createData(
{
    var bd:BitmapData = new
    bitmap = new Bitmap(bd);

    bitmap.bitmapData = drawInBitmapData(bitmap.bitmapData);
}
```

Bitmap Data

Cum il construiesc si afisez pe e

Functia asta deseneaza o linie oblica… (din pixeli)

```
private function drawInBitmapData(b:BitmapData):BitmapData
{
    var bd:BitmapData = b;
    for(var i:uint = 50; i<250; i++)
    {
        bd.setPixel(i,i,0xff0000);
    }
    return(bd);
}
```

Bitmap Data

As putea sa construiesc un obiect BitmapData dintr-o poza (JPG):

```
private function drawInBitmapData2(b:Bitmap):void
{
    var url:URLRequest = new URLRequest("img.jpg");
    var ld:Loader = new Loader();
    ld.load(url);
    ld.contentLoaderInfo.addEventListener( Event.COMPLETE, copieBytes);

    function copieBytes(e:Event):void
    {
        b.bitmapData = (ld.content as Bitmap).bitmapData.clone() ;
    }
}
```

Bitmap Data

Trimit ca parametru tot obiectul de tip Bitmap

As putea sa construie Data dintr-o poza (JPG):

```
private function drawInBitmapData2(b:Bitmap):void
{
    var url:URLRequest = new URLRequest("img.jpg");
    var ld:Loader = new Loader();
    ld.load(url);
    ld.contentLoaderInfo.addEventListener( Event.COMPLETE, copieBytes);

    function copieBytes(e:Event):void
    {
        b.bitmapData = (ld.content as Bitmap).bitmapData.clone() ;
    }
}
```

Bitmap Data

Si ii modific proprietatea bitmapData (dupa ce am incarcat poza)

As putea sa construie Data dintr-o poza (JPG):

```
private function drawInBitmapDat  Bitmap):void
{
    var url:URLRequest = ne  Request("img.jpg");
    var ld:Loader = new Lo  ();
    ld.load(url);
    ld.contentLoaderInf  addEventListener( Event.COMPLETE, copieBytes);

    function copieBytes(e:Event):void
    {
        b.bitmapData = (ld.content as Bitmap).bitmapData.clone() ;
    }
}
```

Bitmap Data

Eventual fac ceva cu obiectul Bitmap

As putea sa construie Data dintr-o poza (JPG):

```
    function copieBytes(e:Even void
    {
        b.bitmapData = (ld.content as Bitmap).bitmapData.clone() ;
        doSomethingWithPic(b);
    }
}
private function doSomethingWithPic(b:Bitmap):void
{

}
```

Bitmap Data

As putea sa construie Data dintr-o poza (JPG):

Eventual fac ceva cu obiectul Bitmap

```
    function copieBytes(e:Even oid
    {
        b.bitmapData = (ld. cent as Bitmap).bitmapData.clone() ;
        doSomethingWithPic );
    }
}
private function doSomethingWithPic(b:Bitmap):void
{

}
```

Bitmap Data

Dupa incarcare:

Bitmap Data

Sa modificam metoda doSomethingWithPic(b:Bitmap):

```
private function doSomethingWithPic(b:Bitmap):void
{
    var dreptunghi:Rectangle = new Rectangle ( 80,200,30,30 );
    b.bitmapData.fillRect( dreptunghi, 0x00ffffff );
}
```

Bitmap Data

Dupa modificare:

Bitmap Data

Sa modificam metoda doSomethingWithPic(b:Bitmap):

```
private function doSomethingWithPic(b:Bitmap):void
{
    var dreptunghi:Rectangle = new Rectangle ( 80,200,30,30 );
    b.bitmapData.fillRect( dreptunghi, 0x00ffffff );
    b.bitmapData.floodFill( 82, 202, 0xff0000 );
}
```

Bitmap Data

Dupa modificare:

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
      var temp : Bitmap = new Bitmap(new BitmapData( b.height, b.width, true, 0 ));
      for( var i : uint = 0; i < b.width; i++ )
          for( var j : uint = 0; j < b.height; j++ )
            {
              var culoare : uint = b.bitmapData.getPixel32( i, j );
              temp.bitmapData.setPixel32( j, i, culoare );
            }
      b.bitmapData = temp.bitmapData;
}
```

Bitmap Data

Dupa modificare:

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
    var s : Shape = new Shape();
    s.graphics.beginFill( 0xffbb00 );
    s.graphics.drawEllipse( 50, 50, 80, 40 );
    b.bitmapData.draw( s );
}
```

Bitmap Data

Dupa modificare:

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
    var dreptunghi : Rectangle = new Rectangle( 100, 100, 90, 50 );
    var punctDestinatie : Point = new Point( 180, 180 );
    b.bitmapData.copyPixels( b.bitmapData, dreptunghi, punctDestinatie );
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
        b.bitmapData.noise(123, 0, 255, 7, false );
}
```

Bitmap Data

Bitmap Data

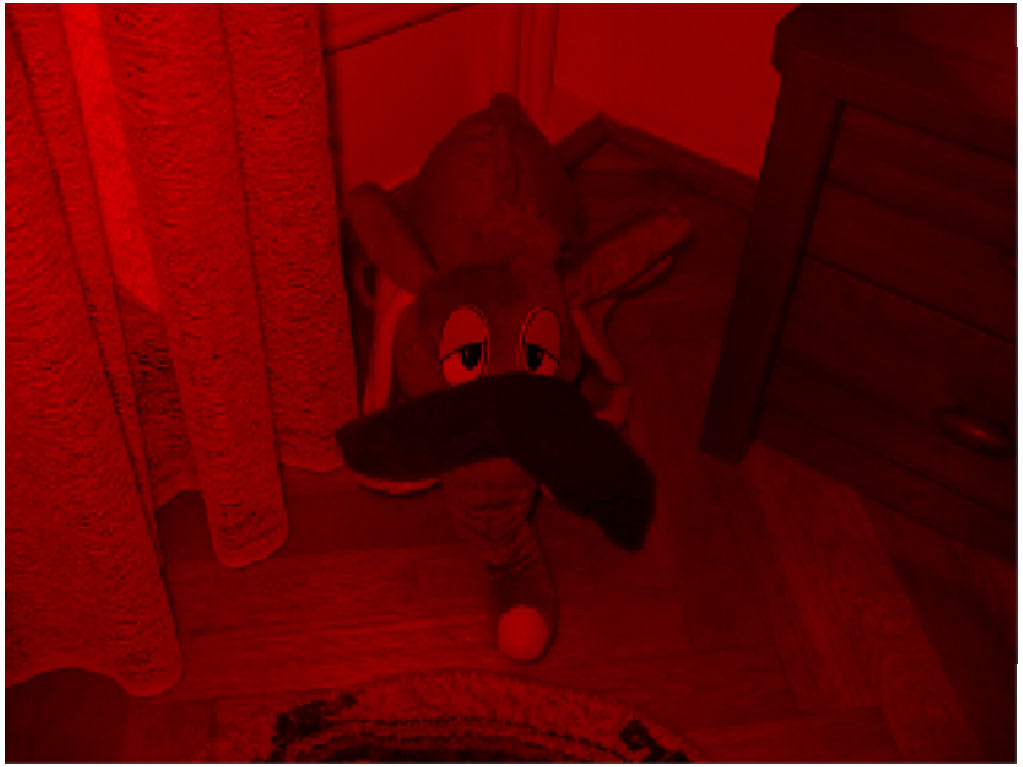
```
private function doSomethingWithPic(b:Bitmap):void
{
      b.bitmapData.perlinNoise( b.width, b.height, 15,
                    5, false, true, 7, false );
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
    var c:BitmapData = b.bitmapData.clone();
    var d:BitmapData = new BitmapData(c.width, c.height, false, 0);
    d.copyChannel( c,
          new Rectangle( 0, 0, c.width, c.height ),
          new Point( 0, 0 ),
          BitmapDataChannel.RED,
          BitmapDataChannel.RED );
    b.bitmapData = d.clone();
}
```

Bitmap Data



Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
     b.bitmapData.threshold( b.bitmapData,
         b.bitmapData.rect, new Point( 0, 0 ),
         ">", 0x00900000, 0xffffff00, 0x00ff0000, true );
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
     var filtru : BlurFilter  = new BlurFilter( 4, 4, 1 );
     b.bitmapData.applyFilter( b.bitmapData,
             b.bitmapData.rect, new Point( 0, 0 ), filtru );
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
     var filtru : BevelFilter   = new BevelFilter( );
     b.filters = [filtru] ;
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
    var matrix:Array = new Array();
    // Fiecare linie va encoda câte o culoare
    // respectiv transparența unui punct din imagine.
    matrix = matrix.concat([1,   0, 0, 0, 0]);
    matrix = matrix.concat([0, 0.5, 0, 0, 0]);
    matrix = matrix.concat([0,   0, 0, 0, 0]);
    matrix = matrix.concat([0,   0, 0, 1, 0]);
    b.filters = [new ColorMatrixFilter(matrix)];
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
      var matrix:Array = new Array();
      // Fiecare linie va encoda câte o culoare
      // respectiv transparența unui punct din imagine.
     matrix = matrix.concat([-1, 0,  0, 0, 255]);
     matrix = matrix.concat([0, -1,  0, 0, 255]);
     matrix = matrix.concat([0,  0, -1, 0, 255]);
     matrix = matrix.concat([0,  0,  0, 1,   0]);
      b.filters = [new ColorMatrixFilter(matrix)];
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
      var matrix:Array = new Array();
     matrix = matrix.concat([.3086, .6094, .0820, 0, 0]);
     matrix = matrix.concat([.3086, .6094, .0820, 0, 0]);
     matrix = matrix.concat([.3086, .6094, .0820, 0, 0]);
     matrix = matrix.concat([0,          0,      0, 1, 0]);
      b.filters = [new ColorMatrixFilter(matrix)];
}
```

Bitmap Data

Bitmap Data

```
private function doSomethingWithPic(b:Bitmap):void
{
    var cf:ConvolutionFilter =  new ConvolutionFilter();
    cf.matrixX = 7;
    cf.matrixY = 3;
    cf.matrix = [0.5, 0, 0, 0, 0.5, 0, 0, 0, 0];
    b.filters = [cf];
}
```

Bitmap Data

Bitmap Data

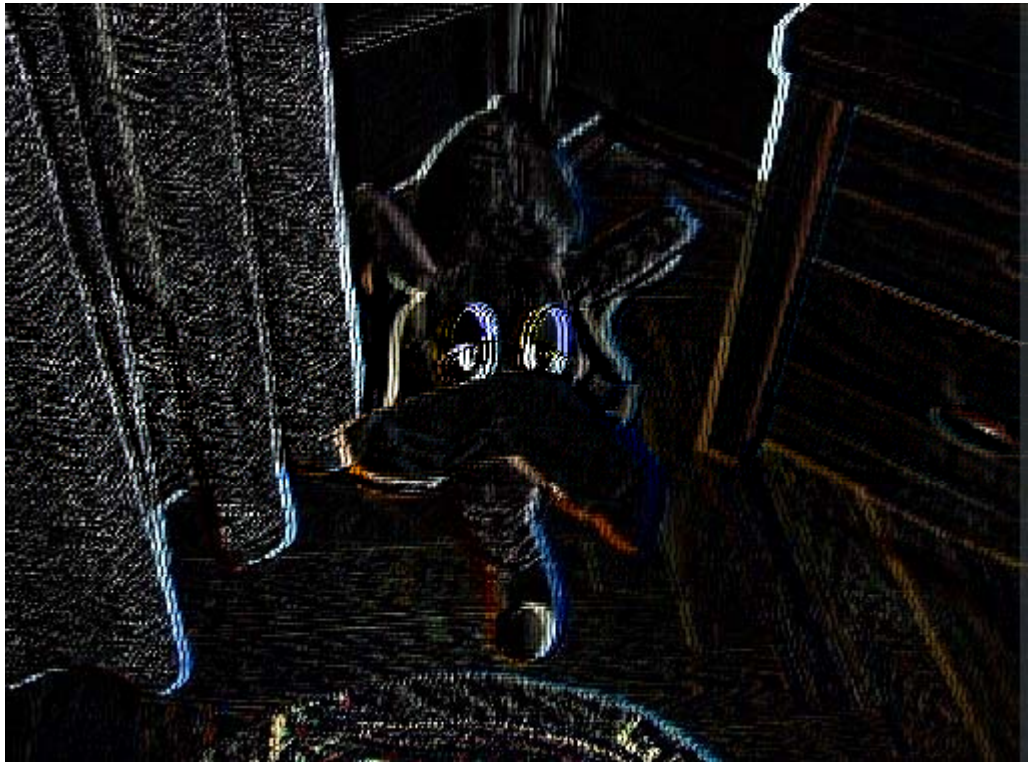
```
private function doSomethingWithPic(b:Bitmap):void
{
    var cf:ConvolutionFilter =  new ConvolutionFilter();
    cf.matrixX = 7;
    cf.matrixY = 3;

    cf.matrix = [0.5, 0, 0, 0, 0.5, 0, 0, 0, 0]; // moire
    cf.divisor = 9; cf.matrix = [5,5,5, 5,0,5, 5,5,5] ; // brightness
    cf.divisor = 1; cf.matrix = [0,-1,0, -1,5,-1, 0,-1,0]; // sharpness;
    cf.divisor = 9; cf.matrix = [-30,30,0, -30,30,0, -30,30,0]; // outline;
    cf.divisor = 1; cf.matrix = [-1, 1, 1, -1, 1, 1, -1, -1, 1]; // emboss;
    cf.divisor = 1; cf.matrix = [0, 1, 0, 1, -3, 1, 0, 1, 0]; // edges;

    b.filters = [cf];
}
```

Numai una din liniile de aici…

Bitmap Data - brightness

Bitmap Data - sharpness

Bitmap Data - outline



Bitmap Data - edges

Bitmap Data – Displacement Map

```
private function doSomethingWithPic(b:Bitmap):void
{
        var dm:DisplacementMapFilter =
                                        new DisplacementMapFilter();
        dm.mapBitmap = createBitmapData( b.width, b.height);
        dm.componentX = BitmapDataChannel.RED;
        dm.componentY = BitmapDataChannel.GREEN;
        dm.mapPoint = new Point(0,0);
        dm.scaleX = 250;
        dm.scaleY = 250;
        dm.mode = DisplacementMapFilterMode.IGNORE ;
        dm.color = 0xff0000;
        dm.alpha = 0;
        b.filters = [dm];
}
```

Bitmap Data – undeee….

```
private function createBitmapData(lungime:uint, inaltime:uint):BitmapData
{
  var bd:BitmapData = new BitmapData(lungime, inaltime);
  bd.perlinNoise(lungime, inaltime, 15, 5, false, true, 7, false);
  var bitmap:Bitmap = new Bitmap(bd);
  this.addChild(bitmap);
  bitmap.x = lungime+3;
  bitmap.y = 0;
  return (bd);
}
```

Bitmap Data – rezultat:

Bitmap Data – rezultat:

Bitmap Data – sau folosind celalat tip de noise:

Bitmap Data – sau folosind celalat tip de noise:

Gata