

Period 5 - TD

Föreläsning 1 av 3

Jonas Petersson

Ursprungligen av: Jeanette Eriksson

nullable

Referenser

DLL

RenderTarget / Renderingsyta

Projekt 5: Tower Defence

- Demo

Projekt 5: Examinationsuppgift

- PDF

Blandat

- Deadline 11 januari 2015 23:59
- Övriga extrauppgifter ska lämnas in och redovisas muntligt innan kursslut (innan tentamen).
Gör det så fort som möjligt så att jag kan hitta tider då ni kan redovisa på (gör det helst under ett labbtillfälle).

Mer blandat

Innan vi tittar på biblioteket...

- int? Betyder att värdet kan vara null

- int? i = 10;
 - double? d1 = 3.14;
 - bool? flag = null;
 - char? letter = 'a';
 - int?[] arr = new int?[10];

```
int? x = 10;
if (x.HasValue){
    System.Console.WriteLine(x.Value);
}
```

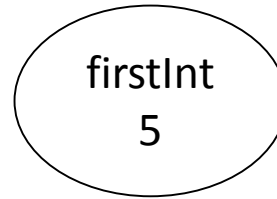
- Typomvandling

- int? n = null;
 - //int m1 = n; // Kompilerar inte.
 - int m2 = (int)n;
 - int m3 = n.Value;
 - int? n2;
 - n2 = 10; // Implicit typomvandling.

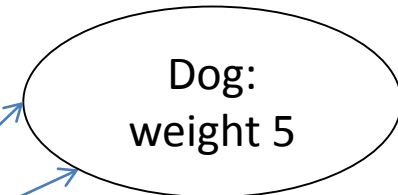
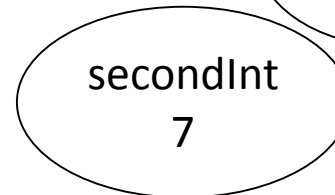
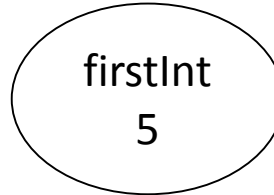
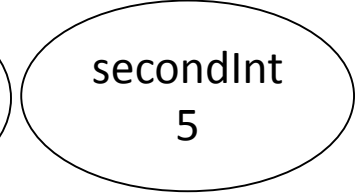
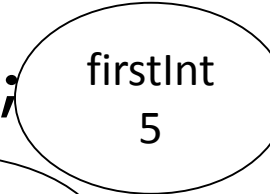
- [http://msdn.microsoft.com/en-us/library/2cf62fcy\(v=vs.120\).aspx](http://msdn.microsoft.com/en-us/library/2cf62fcy(v=vs.120).aspx)

värdetyp <-> referenstyp

- `int firstInt=5`
- `int secondInt=firstInt;`
- `secondInt=7;`
- `Dog milo=new Dog();`
- `milo.weight=5;`
- `Dog fido=milo;`



Värdetyp



Referenstyp



Referenstyp

```
class Class1{
    public static int nrOfClass1Objects =0;
    public int nr = 0;

    public Class1(){
        nrOfClass1Objects++;
    }
}

class Example {
    static void Method(Class1 c) {
        c.nr=100;
    }

    static void Main() {
        Class1 c5 = new Class1();
        Method(c5);
        // c5 är nu 100;
    }
}
```


ref och out

- Reserverat ord för att ange att en parameter skall skickas som referens istället för värde.
- ref parameter måste initieras innan användning
- out parameter behöver inte initieras.
- Värde typer är ex. egendefinierade structar, bool, numeriska typer som heltal, float m.m.
- Referenstyper är ex. klasser och listor m.m

Ref exempel

```
class RefExample {  
    static void Method(ref int i) {  
        i = 44;  
    }  
  
    static void Main() {  
        int val = 0;  
        Method(ref val);  
        // val is now 44  
    }  
}
```

Out exempel

```
class OutExample {  
    static void Method(out int i) {  
        i = 44;  
    }  
    static void Main() {  
        int value;  
        Method(out value);  
        // value is now 44  
    }  
}
```

Spline-biblioteket

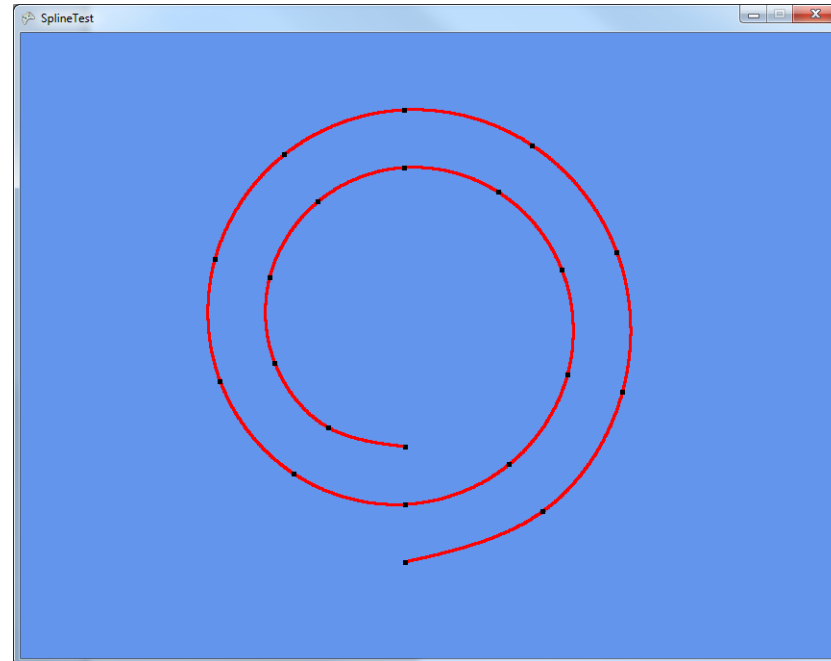
- Man kan inkludera färdigkompilerade bibliotek i sina projekt, tex en dll-fil.
- Ett sådant exempel är spline.dll som ni bör använda i projektet för att skapa den väg (kurva) där fienden ska gå.
- Finns bifogad till Examinationsuppgift 5.

Så här gör man

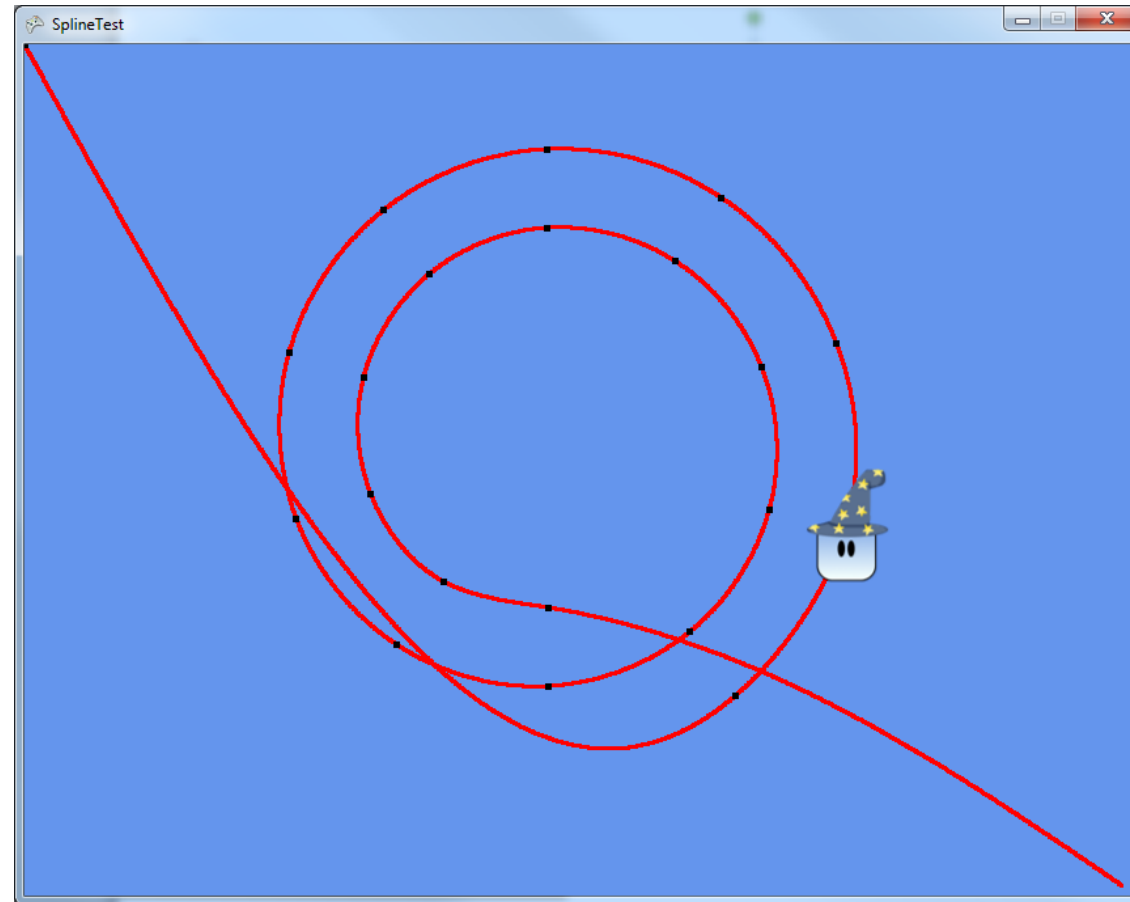
- Kopiera dll-filen till projektkatalogen
- Högerklicka på projektet i Solution Explorer
- Välj *Add Reference* och fliken *Browse*
- Välj `Spline.dll`
- Lägg till `using Spline;` där det behövs

Gränssnittet till Spline.dll (Spiral)

```
75         #region DEFAULT CURVE, USED FOR TESTING
76             public void generateDefaultPath();
77         #endregion
78     }
79 }
```



Andvändning (1) DEMO



Gränssnittet till Spline.dll (Play)

```
9 namespace Spline {
10     public class SimplePath {
11         private Path path;
12
13         //avstånden mellan de olika punkterna
14         public float[] Toffsets;
15
16         //KONSTRUKTOR
17         public SimplePath(GraphicsDevice graphics);
18
19         #region USED WHEN PLAYING
20             //Banans startpunkt (brukar vara 0) och slutpunkt
21             public float beginT;
22             public float endT;
23
24             //Det går bra att anropa GetPos även med värden utanför området.
25             //Returnerar positionen på banan på t avstånd från början av kurvan
26             public Vector2 GetPos(float t);
27         #endregion
28     }
29 }
```


Gränssnittet till Spline.dll (Edit)

```
29         #region USED WHEN EDITING
30             //ta bort alla gamla punkter
31             public void Clean();
32
33             //lägger till en punkt på kurvan
34             public void AddPoint(Vector2 p);
35
36             //ger exception om 'i' inte är giltigt
37             public void InsertPoint(Vector2 p, int i);
38             public void RemovePoint(int i);
39
40             public int AntalPunkter;
41
42             //ger exception om 'i' inte är giltigt
43             //Returnerar en punkt på kurvan 'i' enheter från kurvans början
44             public Vector2 GetPos(int i);
45
46             //ger exception om 'i' inte är giltigt
47             //Lägger till en punkt på kurvan ex. i=0 och pos=Vector.Zero gör att
kurvan startar i övre vänstra hörnet.
48             public void SetPos(int i, Vector2 pos);
49
50             //Räknar om Spline, anropas automatiskt vid förändringar av spline
51             //svarar alltid true!
52             public bool Compute();
53
54             //Om det finns minst en punkt inom maxDist så får man index på den
närmaste, annars null.
55             public int? GetNearestPoint(Vector2 pos, float maxDist);
56
57             //Om det finns en punkt inom maxDist så får man index på den i point1.
58             //Om det finns ytterligare en punkt så ger point2 index på den
59             //true om det fanns två punkter
60             public bool GetNearestPoints(Vector2 pos, float maxDist, out int? point1,
out int? point2);
61         #endregion
```

Gränssnittet till Spline.dll (Draw)

```
63         #region USED FOR DRAWING
64             //ritar ut linjen som ett rött streck
65             public void Draw(SpriteBatch spriteBatch);
66
67             //ritat ut punkterna på kurvan
68             public void DrawPoints(SpriteBatch spriteBatch);
69
70             //rita ut en punkt extra markerad
71             //ger exception om 'i' inte är giltigt
72             public void DrawPointMarked(SpriteBatch spriteBatch, int i);
73         #endregion
```

Användning (2)

```
using Spline;

namespace SplineTest {

    public class Game1 : Microsoft.Xna.Framework.Game {
        GraphicsDeviceManager graphics;
        SpriteBatch spriteBatch;

        Texture2D texture;
        SimplePath path;

        int width = 850;
        int height = 650;

        float texPos;

        public Game1() {
            graphics = new GraphicsDeviceManager(this);
            Content.RootDirectory = "Content";
        }
    }
}
```

Användning (3)

```
protected override void LoadContent() {  
  
    path = new SimplePath(graphics.GraphicsDevice);  
    path.generateDefaultPath(); //behövs inte, finns redan en från början  
  
    //path.Clean(); // tar bort alla punkter  
  
    graphics.PreferredBackBufferHeight = height;  
    graphics.PreferredBackBufferWidth = width;  
    graphics.ApplyChanges();  
  
    spriteBatch = new SpriteBatch(GraphicsDevice);  
    texture = Content.Load<Texture2D>("Trollkarl_bild");  
  
    //sätter bildens startpunkt till början av kurvan  
    texPos = path.beginT;  
    path.SetPos(0, Vector2.Zero);  
    path.AddPoint(new Vector2(width, height));  
}
```

Användning (4)

```
protected override void Update(GameTime gameTime) {  
  
    if (GamePad.GetState(PlayerIndex.One).Buttons.Back == ButtonState.Pressed)  
        this.Exit();  
  
    //förflyttar positionen längs kurvan  
    texPos++; //bestämmer hastigheten  
  
    base.Update(gameTime);  
}
```

Användning (5)

```
protected override void Draw(GameTime gameTime) {  
    GraphicsDevice.Clear(Color.CornflowerBlue);  
  
    spriteBatch.Begin();  
    //ritar ut kurvan  
    path.Draw(spriteBatch);  
    //ritar ut punkterna på kurvan  
    path.DrawPoints(spriteBatch);  
  
    //ritar ut trollkarlen på kurvan  
    if(texPos < path.endT)  
        spriteBatch.Draw(texture, path.GetPos(texPos), new Rectangle(0,0,texture.Width, texture.Height),  
            Color.White, Of, new Vector2(texture.Width / 2, texture.Height / 2),1f, SpriteEffects.None, Of);  
    spriteBatch.End();  
    base.Draw(gameTime);  
}
```

Renderingsytor/RenderTarget

- Pixelperfekta kollisioner –har Daniel gått igenom
 - Kollar pixel för pixel vilka texturer som överlappar
- Samma princip för att placera ut torn

```
public bool CanPlace(GameObject g)
{
    Color[] pixels = new Color[g.tex.Width * g.tex.Height];
    Color[] pixels2 = new Color[g.tex.Width * g.tex.Height];
    g.tex.GetData<Color>(pixels2);
    backgroundLayer.GetData(0, g.bb, pixels, 0, pixels.Length);

    for (int i = 0; i < pixels.Length; ++i)
    {
        if (pixels[i].A > 0.0f && pixels2[i].A > 0.0f)
            return false;
    }

    return true;
}
```

RenderTarget

```
backgroundLayer = new RenderTarget2D(GraphicsDevice, 1280, 720);  
go = new List<GameObject>();
```

```
private void DrawRenderTargetLayer(GraphicsDevice device)  
{  
    // skapa ny spriteBatch  
    SpriteBatch sb = new SpriteBatch(device);  
  
    // sätt till annat renderTarget  
    device.SetRenderTarget(backgroundLayer);  
    device.Clear(Color.Transparent);  
    sb.Begin();  
  
    foreach(GameObject o in go)  
    {  
        o.Draw(sb);  
    }  
  
    sb.End();  
  
    // Sätt tillbaka graphicsDevice till ordinarie fönster  
    device.SetRenderTarget(null);  
}
```


Texturemanager

- Beroende på hur programmet är strukturerat och hur texturerna ser ut kan det ibland vara praktiskt att ladda in samtliga texturer i en speciell klass, TextureManager.
- TextureManager läser in texturerna som static, och respektive tornklass kan nå dem genom ex. TextureManager.tornTexture1.

Texturemanager exempel.

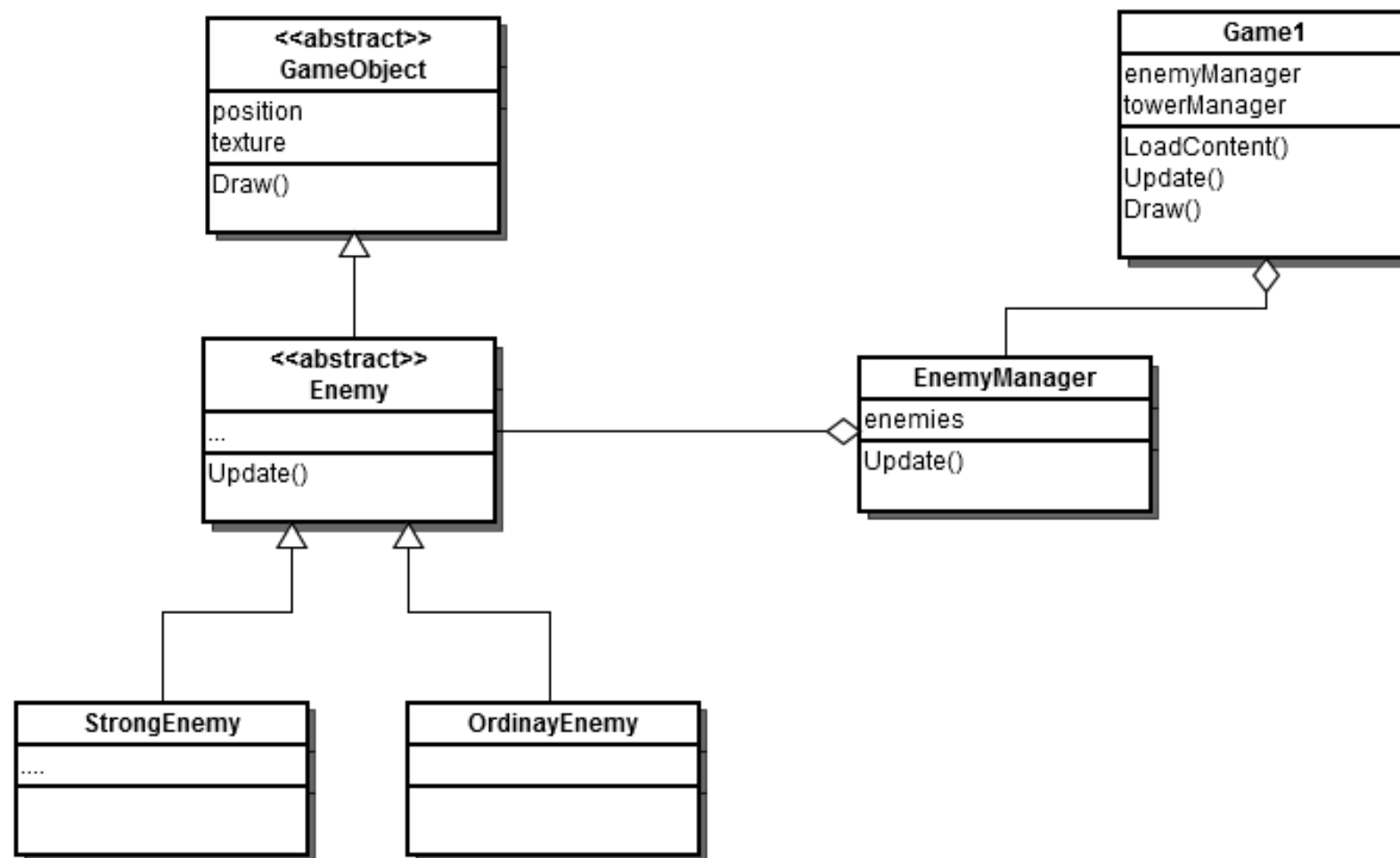
```
static class TextureManager
{
    public static Texture2D StrongTowerTex { get; private set; }
    public static Texture2D BasicTowerTex { get; private set; }
    // . . .

    public static void LoadContent(ContentManager Content)
    {
        StrongTowerTex = Content.Load<Texture2D>("tower1");
        BasicTowerTex = Content.Load<Texture2D>("tower2");
        // . . .
    }
}
```

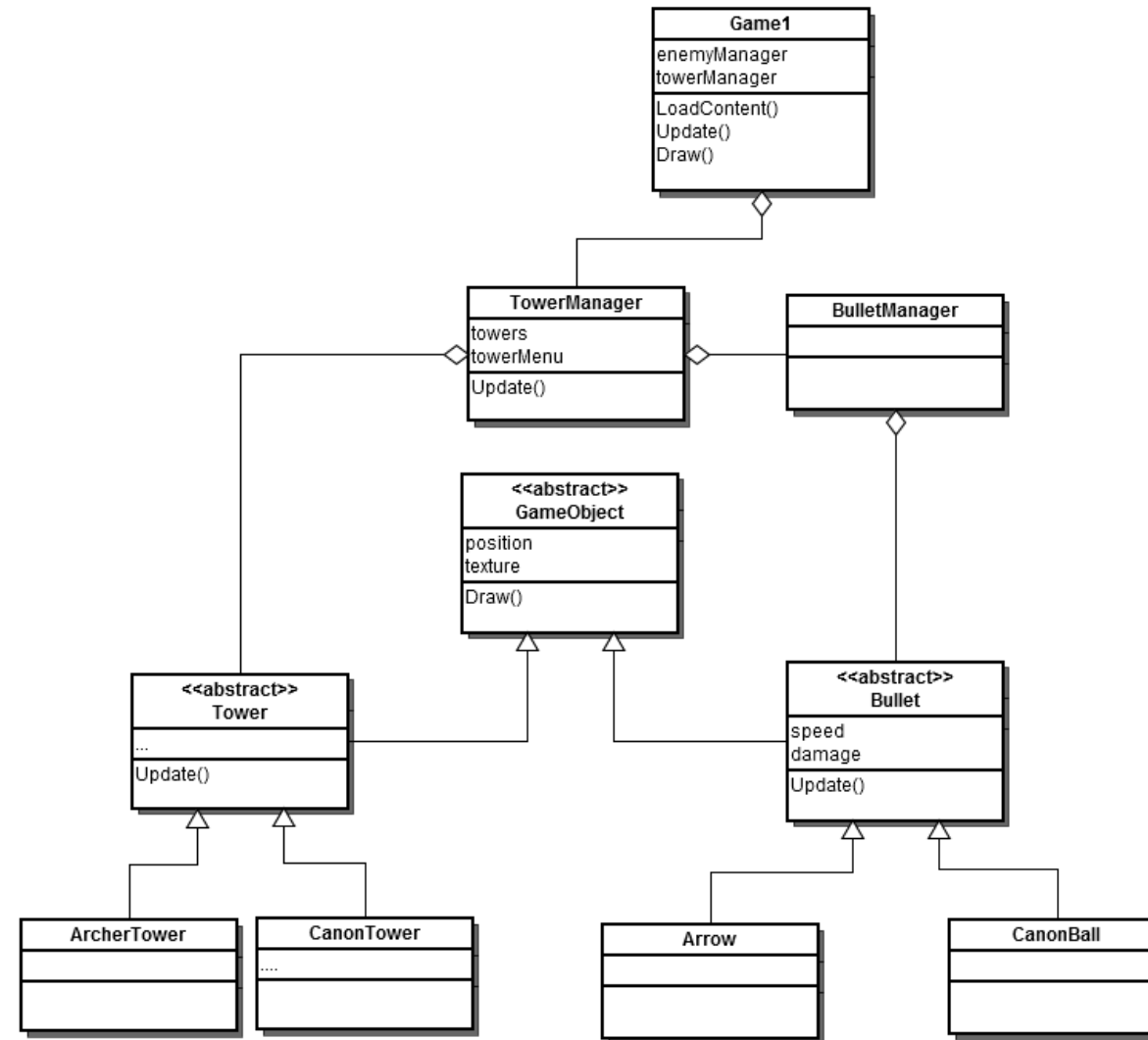
Managers

- Håller ordning på grupper av objekt
- Distribuerar anrop till Update och Draw metoder
- Hanterar interaktion mellan objekten i gruppen
- Är ett gränssnitt mellan objektgruppen och andra objekt och managers (dvs. objektgrupper)

Exempelstruktur fiender



Exemplestruktur torn och skott



Exempelstruktur, översikt

