

## Lesbrief 2: AS3, Inheritance, MovieClips en meer

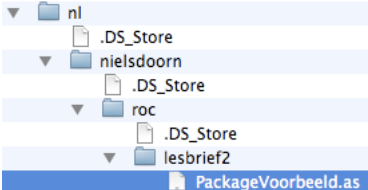
### Veranderingen in AS3

- Packages
- Imports
- Document klasse

## Packages

- Om je klassen onder te verdelen
- Moet je definiëren, maar mag wel leeg zijn
- Structuur op schijf moet overeen komen

```
1 package nl.nielsdoorn.roc.lesbrief2 {  
2     class PackageVoorbeeld {  
3         function PackageVoorbeeld() {  
4             }  
5     }  
6 }
```



## Imports

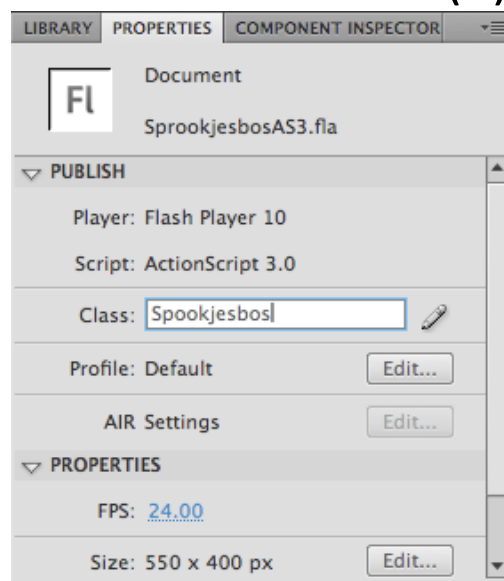
- Om aan te geven welke klassen je nodig hebt die niet in het huidige package staan

```
package {  
    import flash.display.MovieClip;  
    import flash.display.Bitmap;
```

## Document klasse

- Aangeven in .fla welke klasse meteen moet worden aangemaakt en toegevoegd aan de stage
- Moet een MovieClip of ander Display object zijn
- Aangeven in de properties van een .fla bestand

## Document klasse (2)



## Document klasse (3)

- Je kunt de constructor van de Document klasse gebruiken om je applicatie op te bouwen

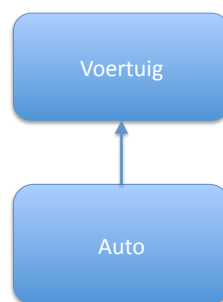
## Overerving

- Klasse hiërarchieën
- Soms wil je een bestaande klasse uitbreiden met meer eigenschappen of met meer of andere verantwoordelijkheden
- Een specifiekere klasse maken

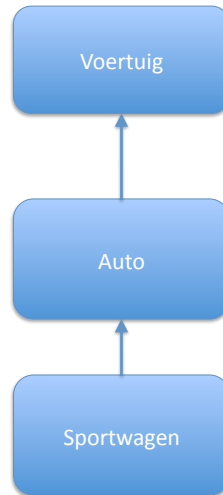
## Voertuig



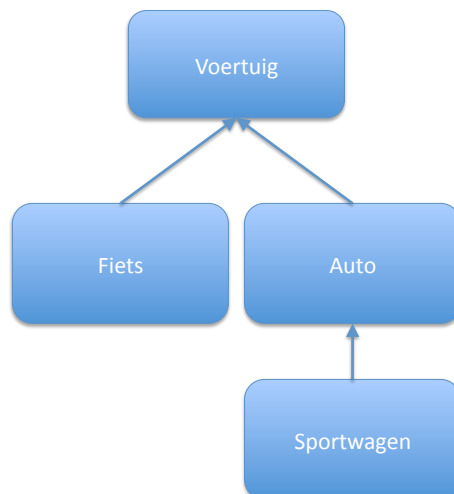
## Voertuig (2)



### Voertuig (3)

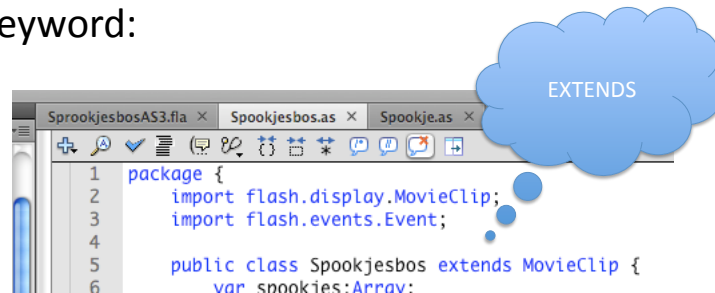


### Voertuig (4)



## Inheritance in ActionScript

- In ActionScript te gebruiken met het extends keyword:

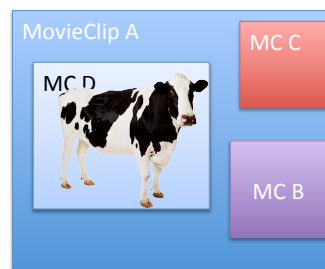


## Opbouw van een flash applicatie

- Stage
  - Bevat alle display objecten die je in je applicatie hebt
- Main timeline
  - Frames
- Symbols
  - MovieClips
  - Buttons
  - Graphics

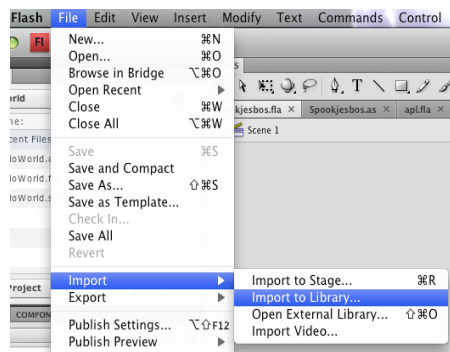
## MovieClips

- Kan ook weer MovieClips bevatten
- Kan worden voorzien van eventhandlers
- Heeft een eigen timeline

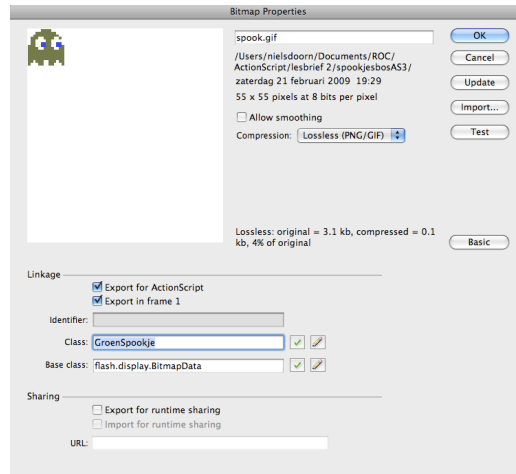


## Objecten importeren

- Afbeeldingen, geluiden, andere .swf bestanden



## Exporteren voor ActionScript



## BitmapData en Bitmap

- Een afbeelding uit de library kun je exporteren als een klasse die erft van BitmapData
- Deze BitmapData kun je gebruiken om een Bitmap op het scherm te zetten
- Een object van het type Bitmap gebruikt dus BitmapData
- Lijkt omslachtig, maar is efficiënter als je een afbeelding vaker wilt gebruiken want dan kun je de data hergebruiken

## BitmapData en Bitmap (2)

```
package {  
    import flash.display.MovieClip;  
    import flash.display.Bitmap;  
  
    class Spookje extends MovieClip {  
        var grsp = new GroenSpookje(55,55);  
        var bitmap:Bitmap;  
  
        function Spookje() {  
            bmap = new Bitmap(grsp);  
            this.addChild(bmap);  
        }  
    }  
}
```