



Applying
Dependency Injection



What views need

What's a view?



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz

Director Jean-Pierre Jeunet

Year 2001

Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal

Director Alejandro González Iñárritu

Year 2000

Category Drama



Add Movie



Search



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

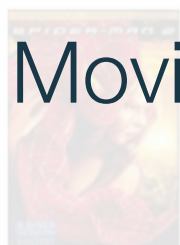
Add Review

Application

MainUI

HBox

MovieList

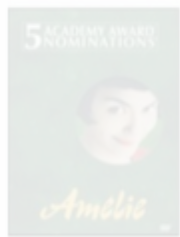


Spider Man II

Starring Tobey Maguire, Kirsten Dunst
Director Sam Raimi

Year 2004

Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz
Director Jean-Pierre Jeunet

Year 2001

Category Comedy, Foreign, Romance



Amores Perros

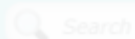
Starring Emilio Echeverria, Gael García Bernal
Director Alejandro González Iñárritu

Year 2000

Category Drama



Add Movie



Search

MovieDetail



Spider Man II

Starring Tobey Maguire, Kirsten Dunst
Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ Joe H.

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ Christine M.

For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

ReviewForm

MovieDetail View

Detail
Text

Trailer



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



Rating

data

Reviews

★★★★☆ **Joe H.**

Another great spider-man movie. I am a fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

Add Review



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

Your rating:



Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

Your rating:



Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



state

★★★★☆ **Joe H.**

Another great spider-man movie. ... an and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

Add Review



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

[Add Review](#)



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was ve
well done. The special effects and the acting is very good. I recommend this movie to everyon
not just comic book fans.

send
notification

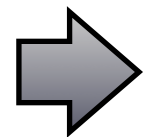
Your rating:



Sending review...



Nice movie!



Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

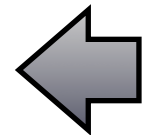
Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

Your rating:



Nice movie!

Review added



receive
notification

Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



send & receive
events

★★★★☆ **Joe H.**

Another great spider-man movie. The movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

Your rating:



Review added

Nice movie!

Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

transitions

Your rating:



Review added

Nice movie!

Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. The movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

validation

Your rating:



Review added

Nice movie!

Cancel

Submit

Data





Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz

Director Jean-Pierre Jeunet

Year 2001

Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal

Director Alejandro González Iñárritu

Year 2000

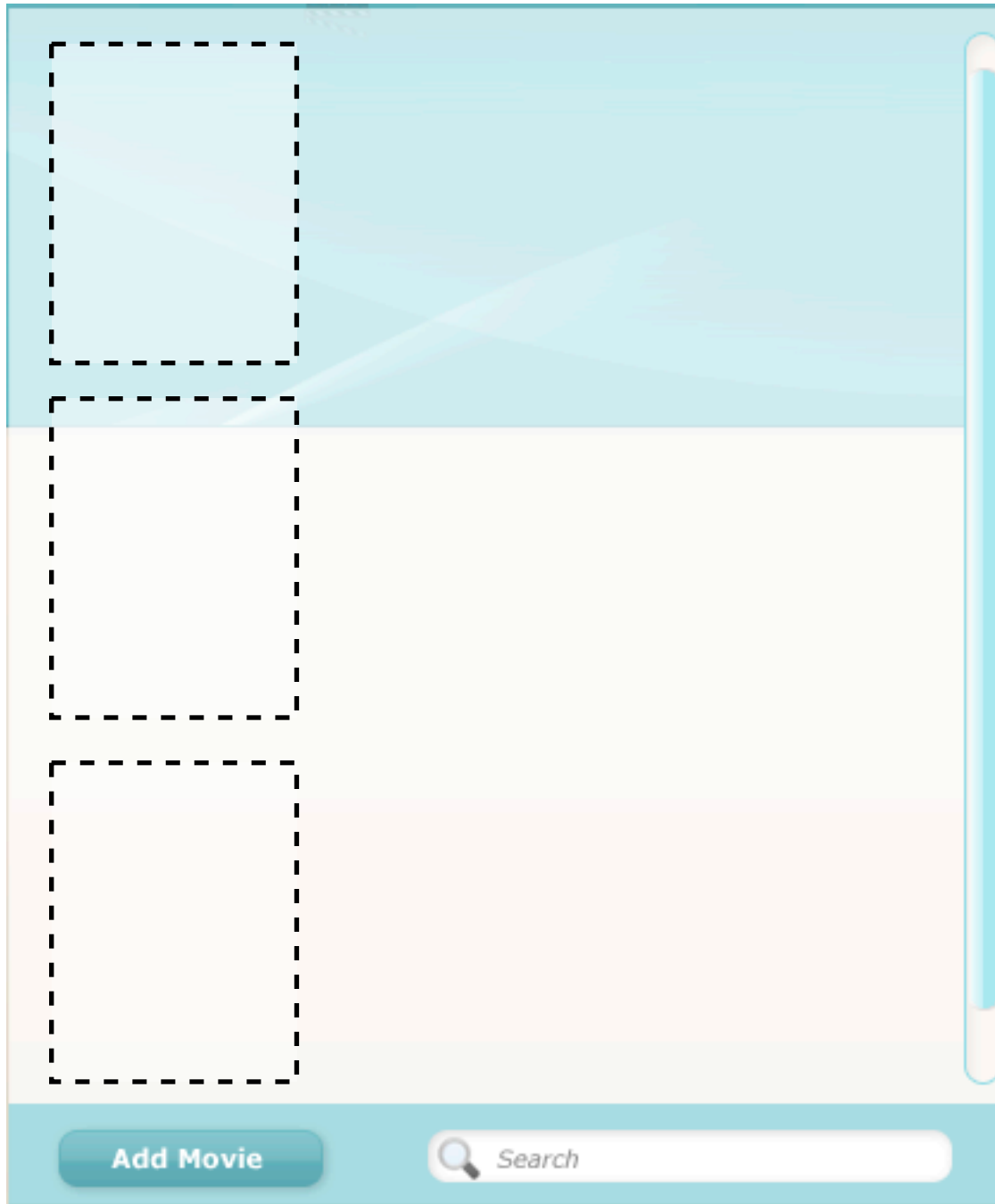
Category Drama



Add Movie

 Search

ArrayCollection
for list
dataproducer



ArrayCollection
for list
dataprovder

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

where do we get this data from?

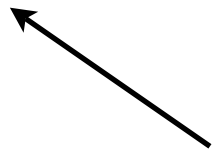
```
<mx:List id="list" dataProvider="{.....}"
```

```
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

Parent
(injection)

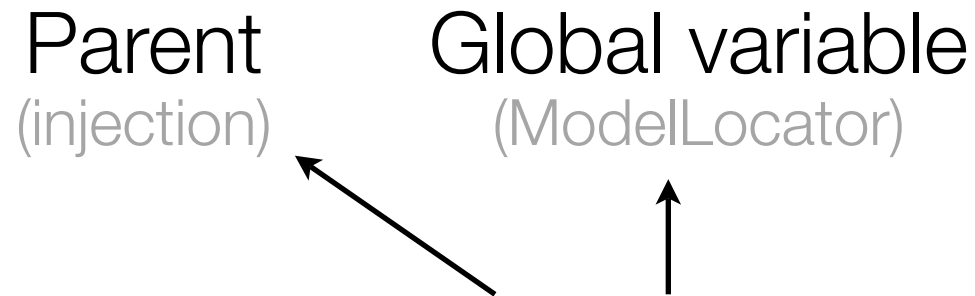


where do we get this data from?

```
<mx:List id="list" dataProvider="{.....}"  
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

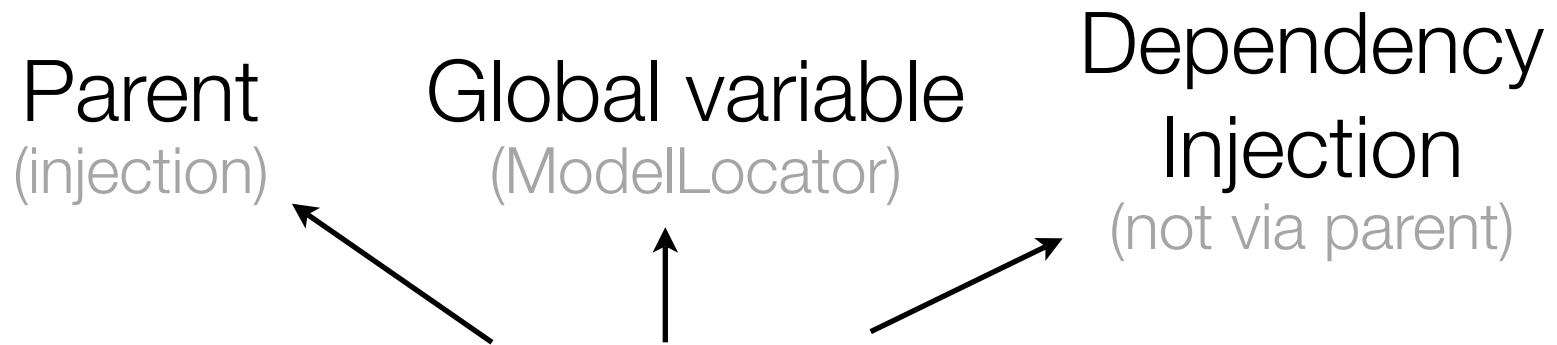


where do we get this data from?

```
<mx>List id="list" dataProvider="{.....}"  
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

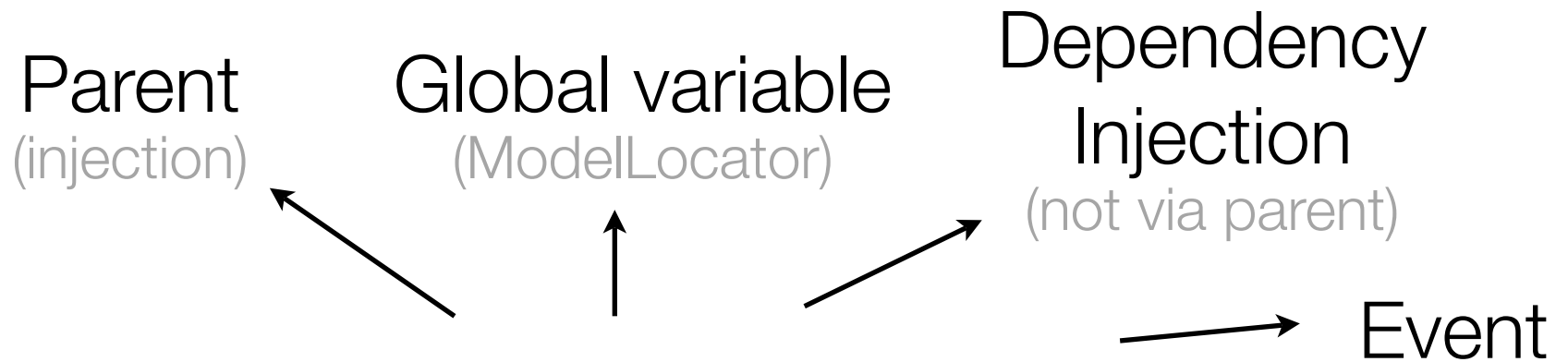


where do we get this data from?

```
<mx>List id="list" dataProvider="{.....}"  
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```



where do we get this data from?

```
<mx>List id="list" dataProvider="{.....}"  
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```


Using a Global Variable

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
<mx:Script>
```

```
<![CDATA[
```

```
import com.movix.model.MyModelLocator;
```

```
[Bindable]
```

```
private var model:MyModelLocator = MyModelLocator.getInstance();
```

```
]]>
```

```
</mx:Script>
```

```
<mx>List id="list" dataProvider="{model.movies}"
```

```
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```

Using Injection / passing data from parent

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
<mx:Script>
```

```
<![CDATA[
```

```
import mx.collections.ArrayCollection;
```

```
[Bindable]
```

```
public var movies:ArrayCollection;
```

```
]]>
```

```
</mx:Script>
```

```
<mx>List id="list" dataProvider="{movies}"
```

```
itemRenderer="myMovieRenderer" />
```

```
</mx:Panel>
```

Using Injection / passing data from parent

Parent view code:

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/08/05/mxml" dataProvider="{movies}">
  <mx:Script>
    <![CDATA[
      import mx.collections.ArrayCollection;

      [Bindable]
      public var movies:ArrayCollection;

    ]]>
  </mx:Script>

  <mx:List id="list" dataProvider="{movies}"
    itemRenderer="myMovieRenderer" />
</mx:Panel>
```

```
<views:MovieList
  movies="{movies}" />
```

Using Injection / passing data from parent

Parent view code:

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/08/06/mxml" dataProvider="{movies}">
  <mx:Script>
    <![CDATA[
      import mx.collections.ArrayCollection;

      [Bindable]
      public var movies:ArrayCollection;

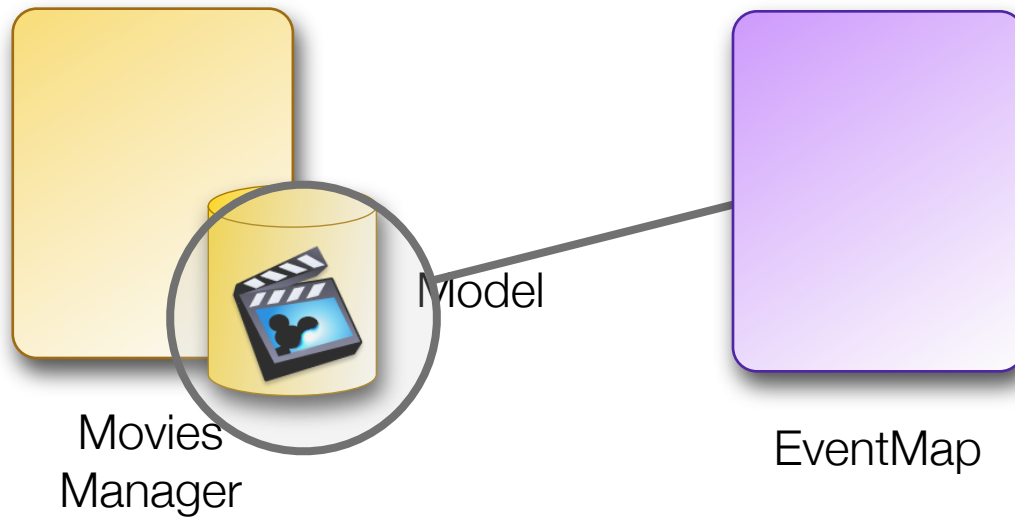
    ]]>
  </mx:Script>

  <mx:List id="list" dataProvider="{movies}"
    itemRenderer="myMovieRenderer" />
</mx:Panel>
```

```
<views:MovieList
  movies="{movies}" />
```

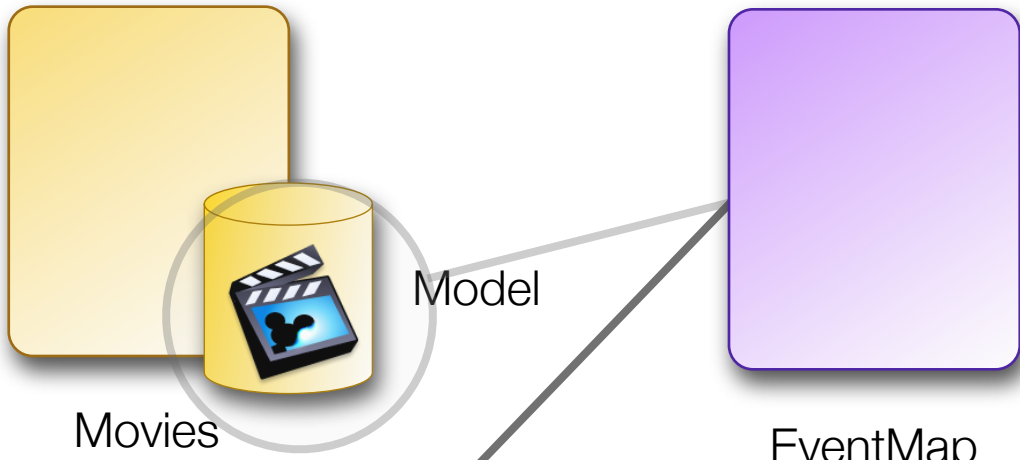
Dependency Injection



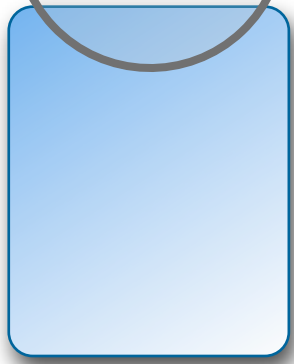


```
public var movies:ArrayCollection;
```





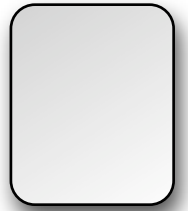
```
public var movies:ArrayCollection;
```



The Event Map

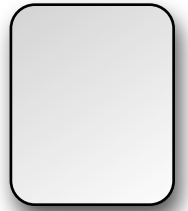


Main application



```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application ...>
    <!-- Event Maps -->
    <maps:MainEventMap />
    <!-- Views -->
    <views:MainUI />
</mx:Application>
```

Main application



```
<?xml version="1.0" encoding="utf-8"?>  
<mx:Application ...>
```

```
  <!-- Event Maps -->
```

```
  <maps:MainEventMap />
```

```
  <!-- Views -->
```

```
  <views:MainUI />
```

```
</mx:Application>
```



com/movix/



business



events



maps



ui (views, item renderers, controls)



vos

```
<?xml version="1.0" encoding="utf-8"?>  
<EventManager ...>
```

```
    // mapping for events
```

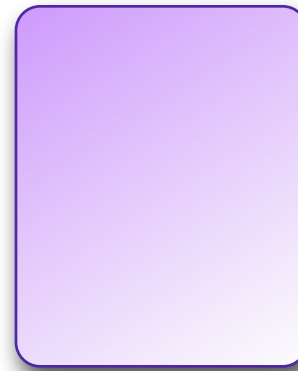
```
    // mapping for injectors
```

```
</EventManager>
```



Movies
Manager

Model



EventMap

```
<EventMap ...>
```

```
<Injectors target="{MovieList}">
```

```
public var movies:ArrayCollection;
```



MovieList

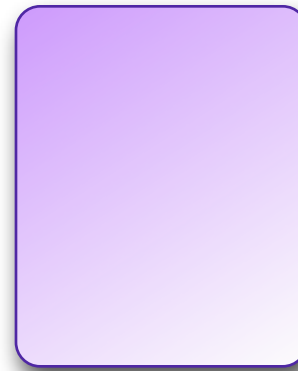
```
</Injectors>
```

```
</EventMap>
```



Movies
Manager

Model



EventMap

```
<EventMap ...>
```

```
<Injectors target="{MovieList}">
```

```
public var movies:ArrayCollection;
```



MovieList

```
</Injectors>
```

```
</EventMap>
```



Movies
Manager

Model



EventManager

```
<EventManager ...>
```

```
<Injectors target="{MovieList}">
```

```
public var movies:ArrayCollection;
```

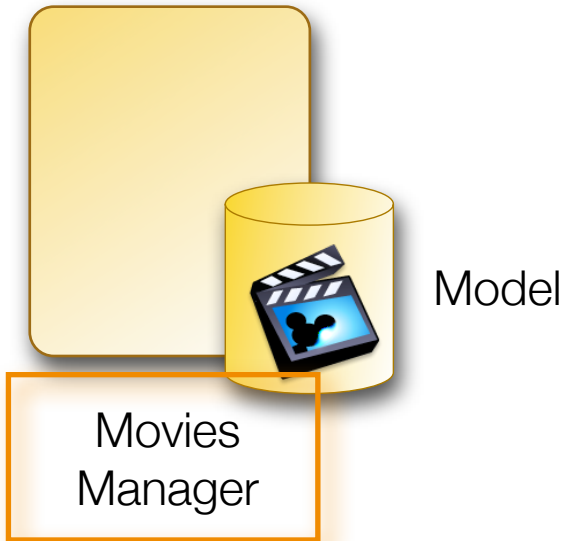
```
<PropertyInjector  
  source="{MoviesManager}"  
  sourceKey="movies"  
  targetKey="movies" />
```



MovieList

```
</Injectors>
```

```
</EventManager>
```

```
<EventMap ...>
```

```
<Injectors target="{MovieList}">
```

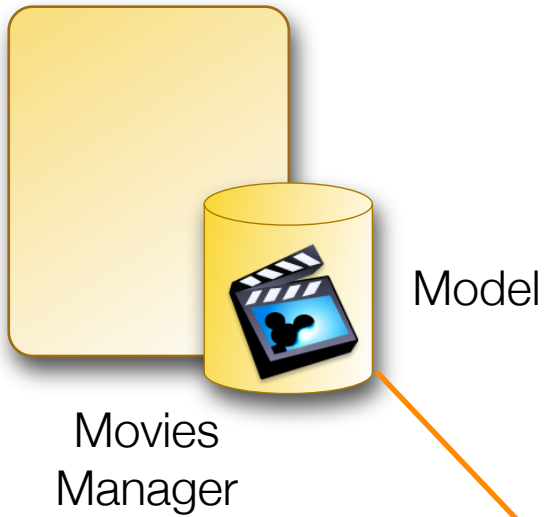
```
public var movies:ArrayCollection;
```



```
<PropertyInjector  
  source="{MoviesManager}"  
  sourcekey="movies"  
  targetkey="movies" />
```

```
</Injectors>
```

```
</EventMap>
```



```
public var movies:ArrayCollection;
```



```
<EventMap ...>
```

```
<Injectors target="{MovieList}">
```

```
<PropertyInjector  
  source="{MoviesManager}"  
  sourceKey="movies"  
  targetKey="movies" />
```

```
</Injectors>
```

```
</EventMap>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
    <PropertyInjector  
      source="{UsersManager}"  
      ...../>
```

```
  </Injectors>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
<Injectors target="{MovieList}">
```

```
<PropertyInjector  
  source="{MoviesManager}"  
  sourceKey="movies"  
  targetKey="movies" />
```

```
<PropertyInjector  
  source="{UsersManager}"  
  .... />
```

```
</Injectors>
```

```
</EventManager>
```



MovieList

MovieList
created

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```



MovieList

```
    <PropertyInjector  
      source="{UsersManager}"  
      ..... />
```

```
  </Injectors>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
    <PropertyInjector  
      source="{UsersManager}"  
      ..... />
```



MovieList

```
  </Injectors>
```

```
</EventManager>
```

Why is this good?



VS

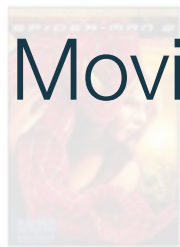


Application

MainUI

HBox

MovieList



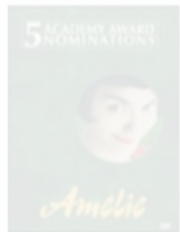
Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz

Director Jean-Pierre Jeunet

Year 2001

Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal

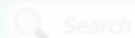
Director Alejandro González Iñárritu

Year 2000

Category Drama



Add Movie



Search

MovieDetail



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

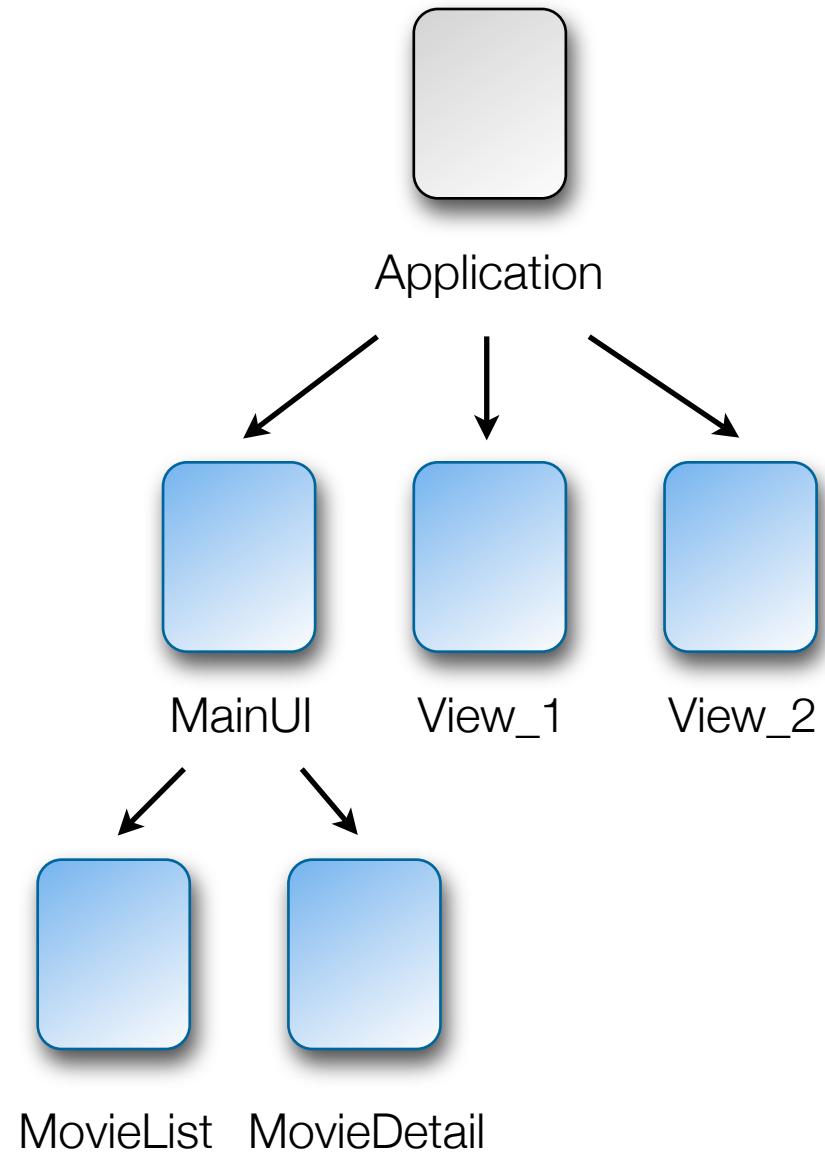
For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

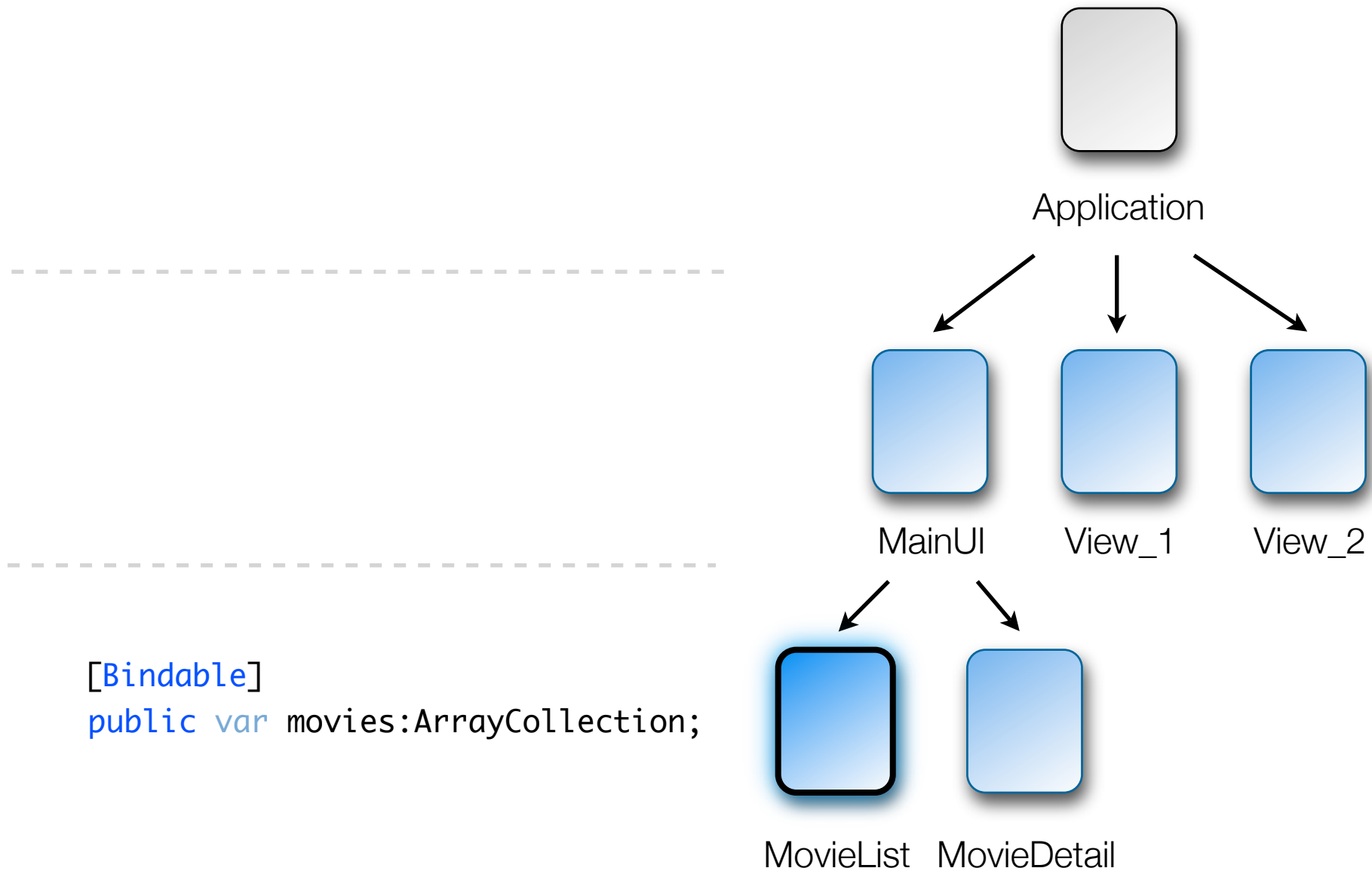
Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

Add Review

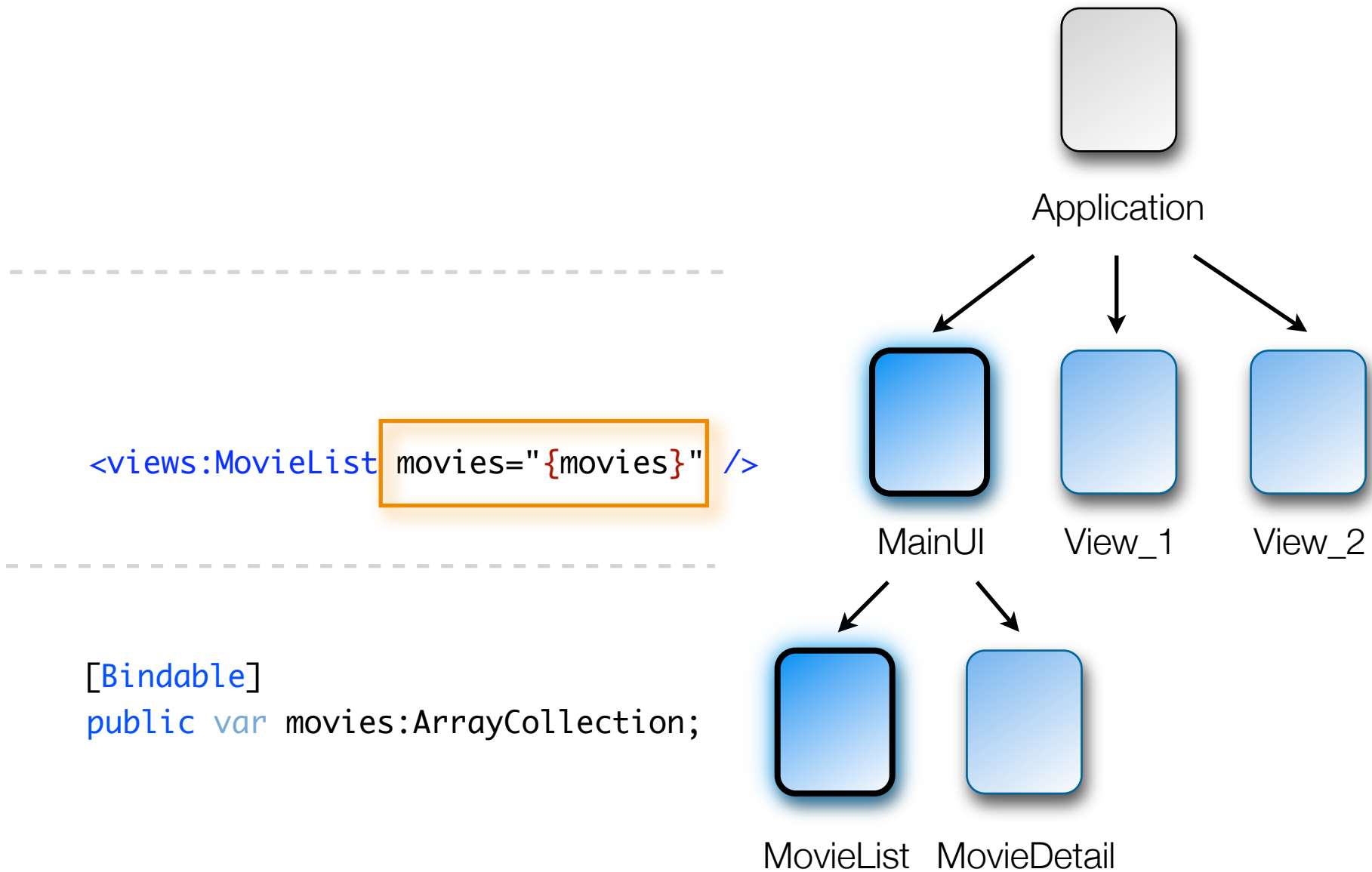
A typical display list tree



Passing data from parent to children



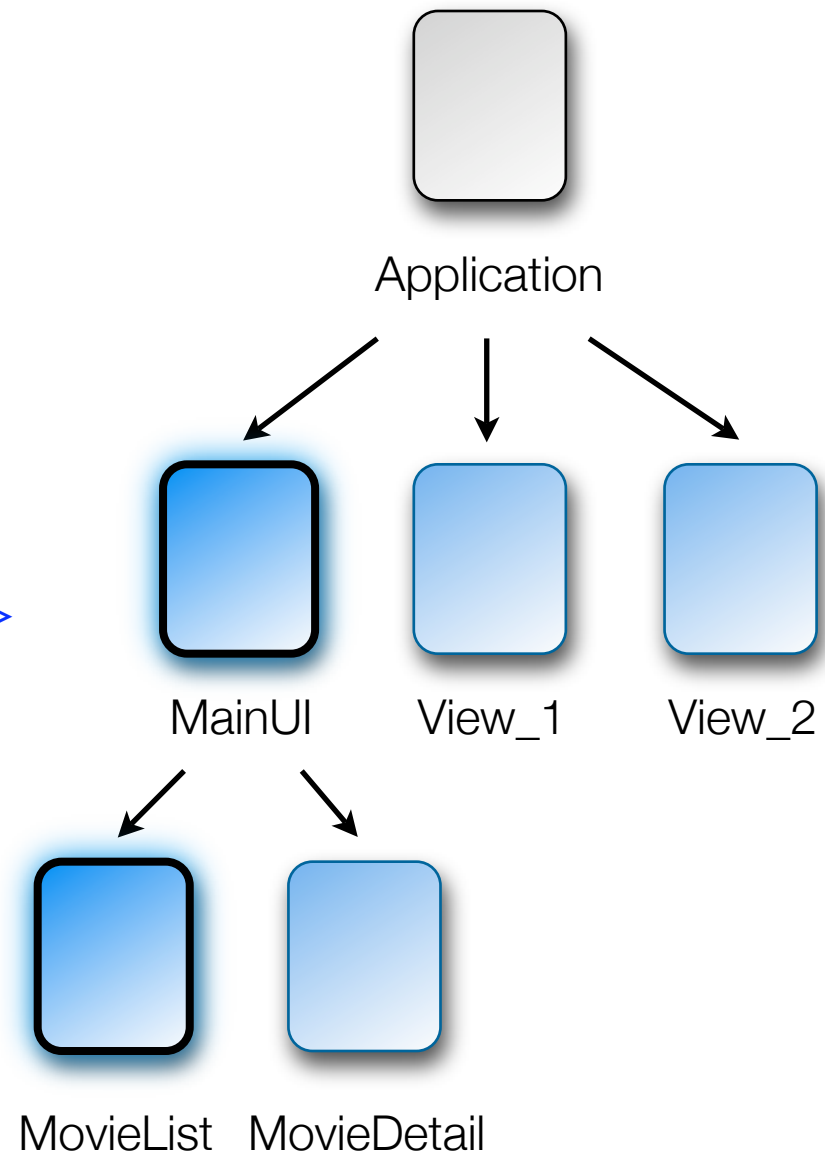
Passing data from parent to children



Passing data from parent to children

```
[Bindable]  
public var movies:ArrayCollection;  
  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

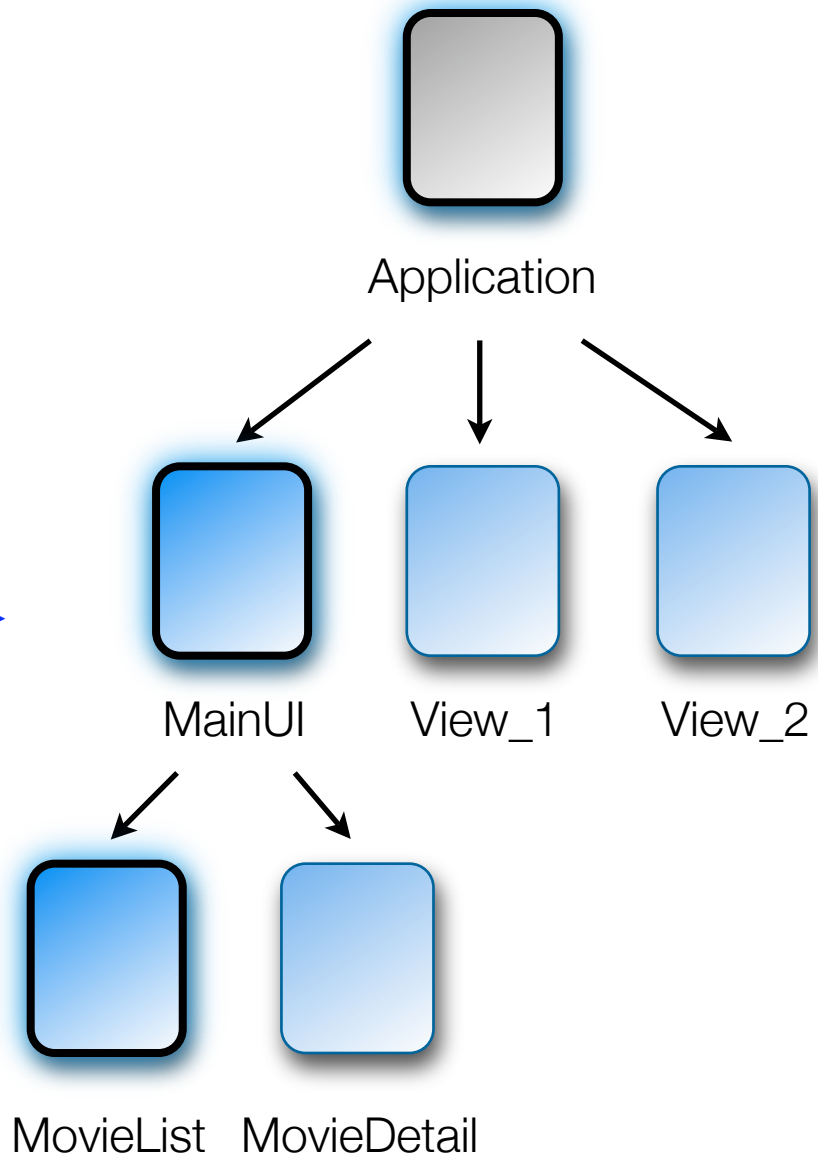


Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 />
```

```
[Bindable]  
public var movies:ArrayCollection;  
  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

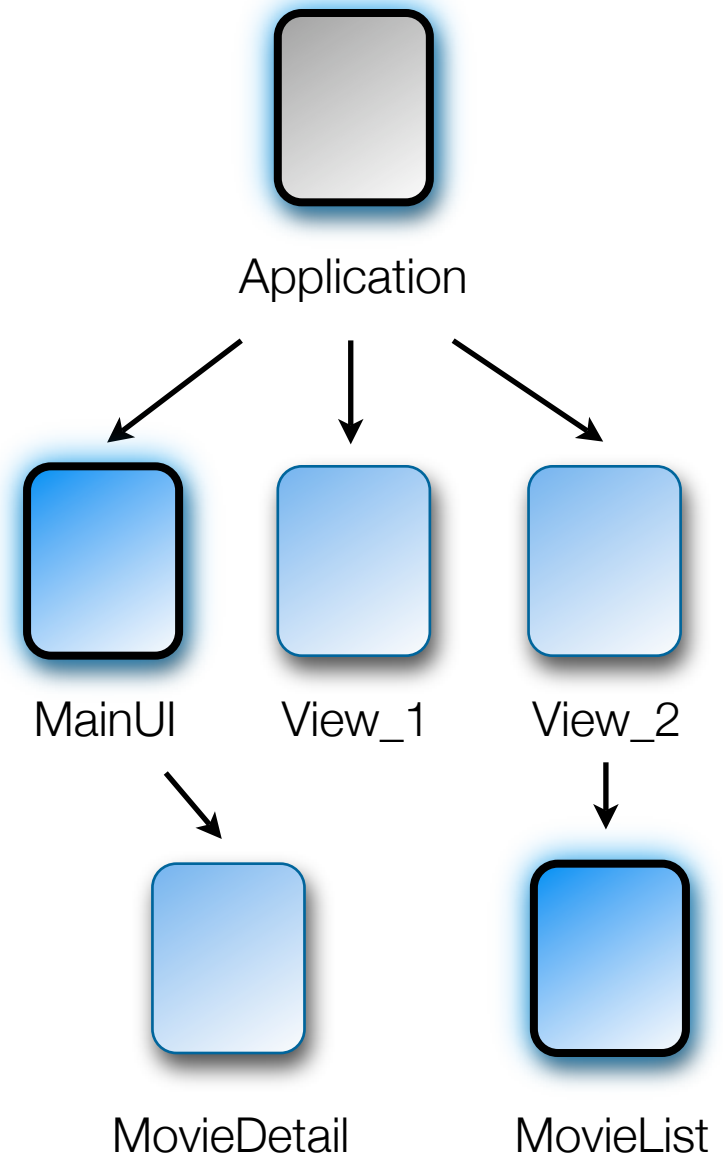


Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 />
```

```
[Bindable]  
public var movies:ArrayCollection;  
  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```



Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 />
```

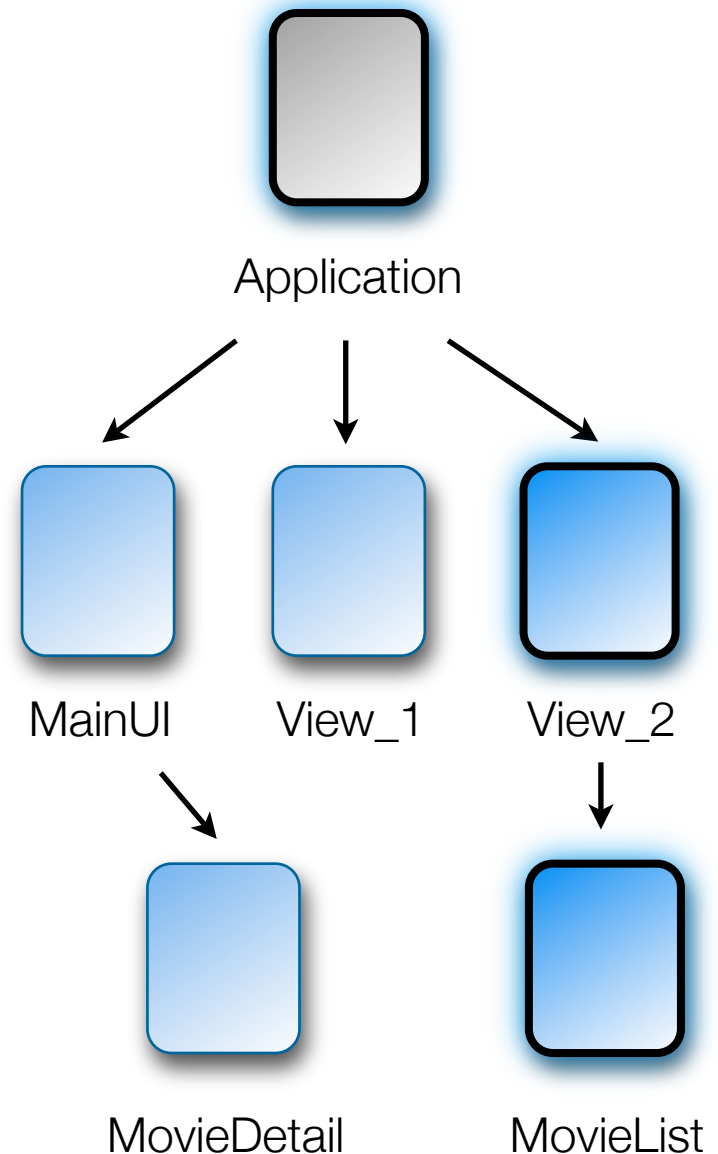
[Bindable]

```
public var movies:ArrayCollection;
```

```
<views:MovieList movies="{movies}" />
```

[Bindable]

```
public var movies:ArrayCollection;
```



Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

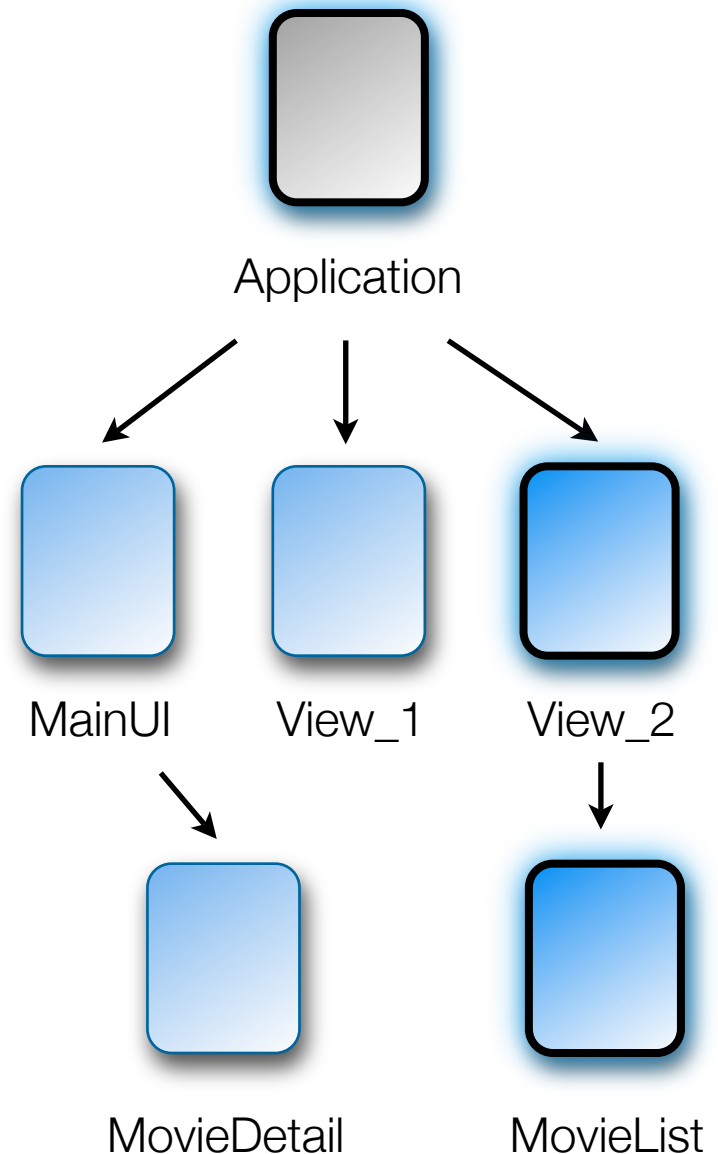
[Bindable]

```
public var movies:ArrayCollection;
```

```
<views:MovieList movies="{movies}" />
```

[Bindable]

```
public var movies:ArrayCollection;
```

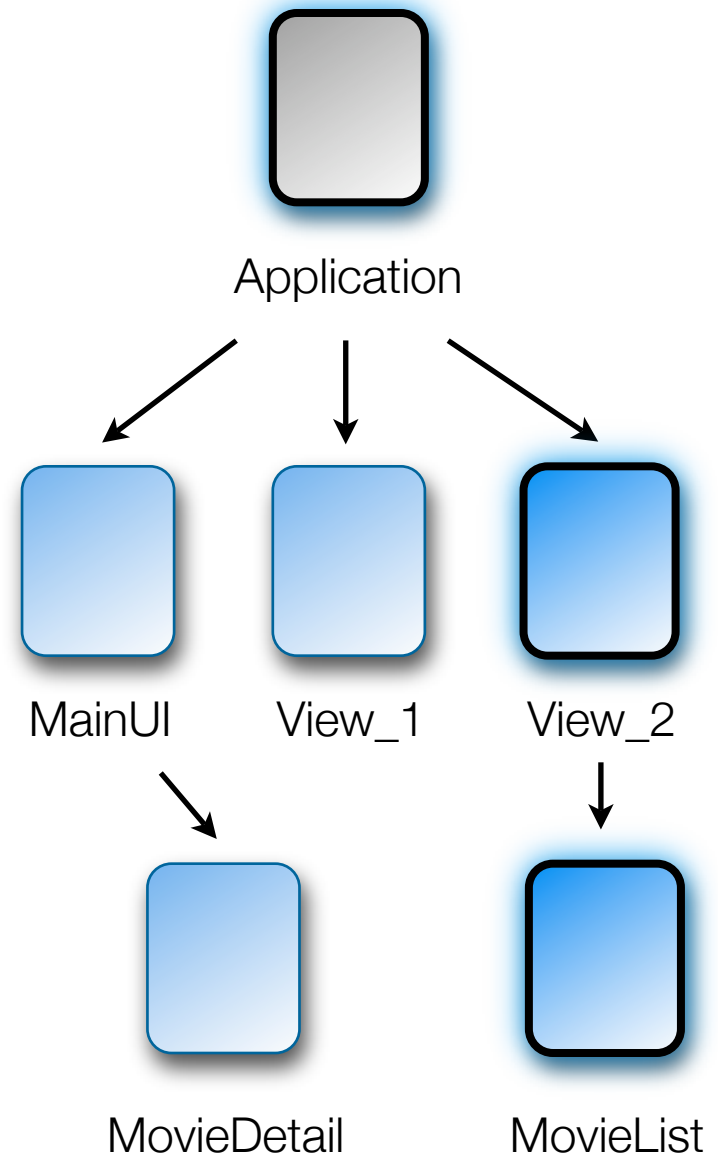


Passing data from parent to children

```
<views:MainUI movies="{movies}" />?  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;?  
  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

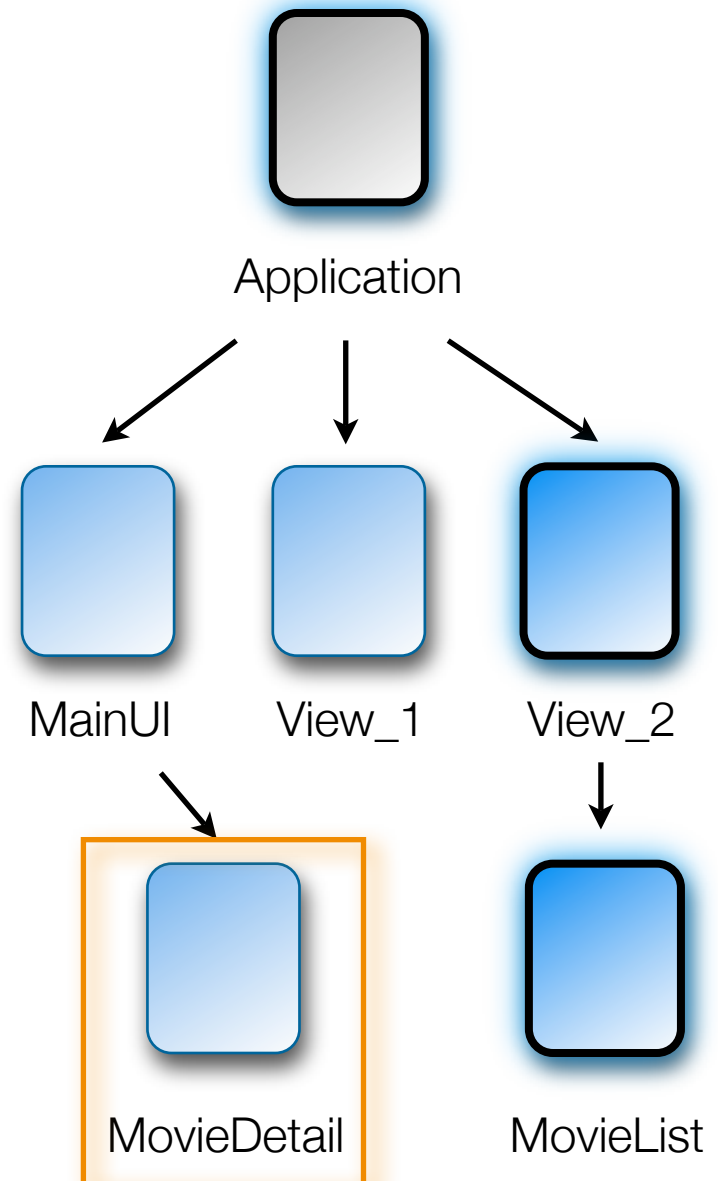


Passing data from parent to children

```
<views:MainUI movies="{movies}" />?  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;?  
  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

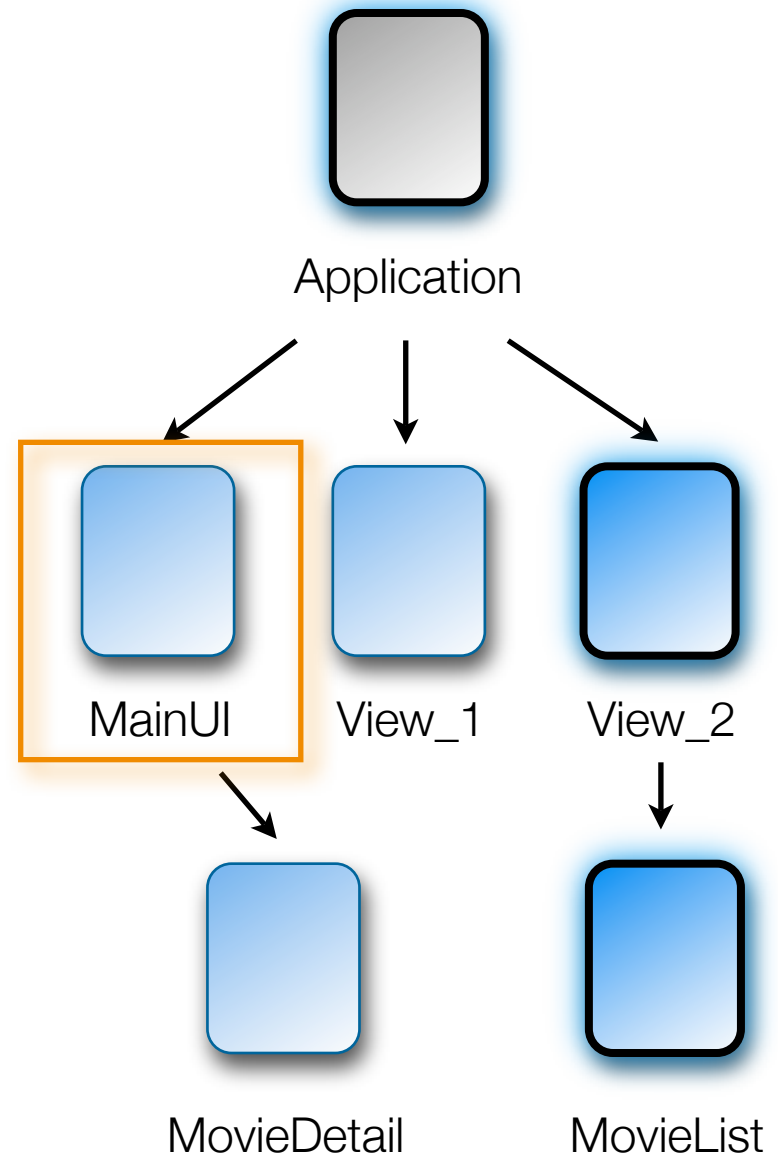


Passing data from parent to children

```
<views:MainUI movies="{movies}" />?  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection; ?  
  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

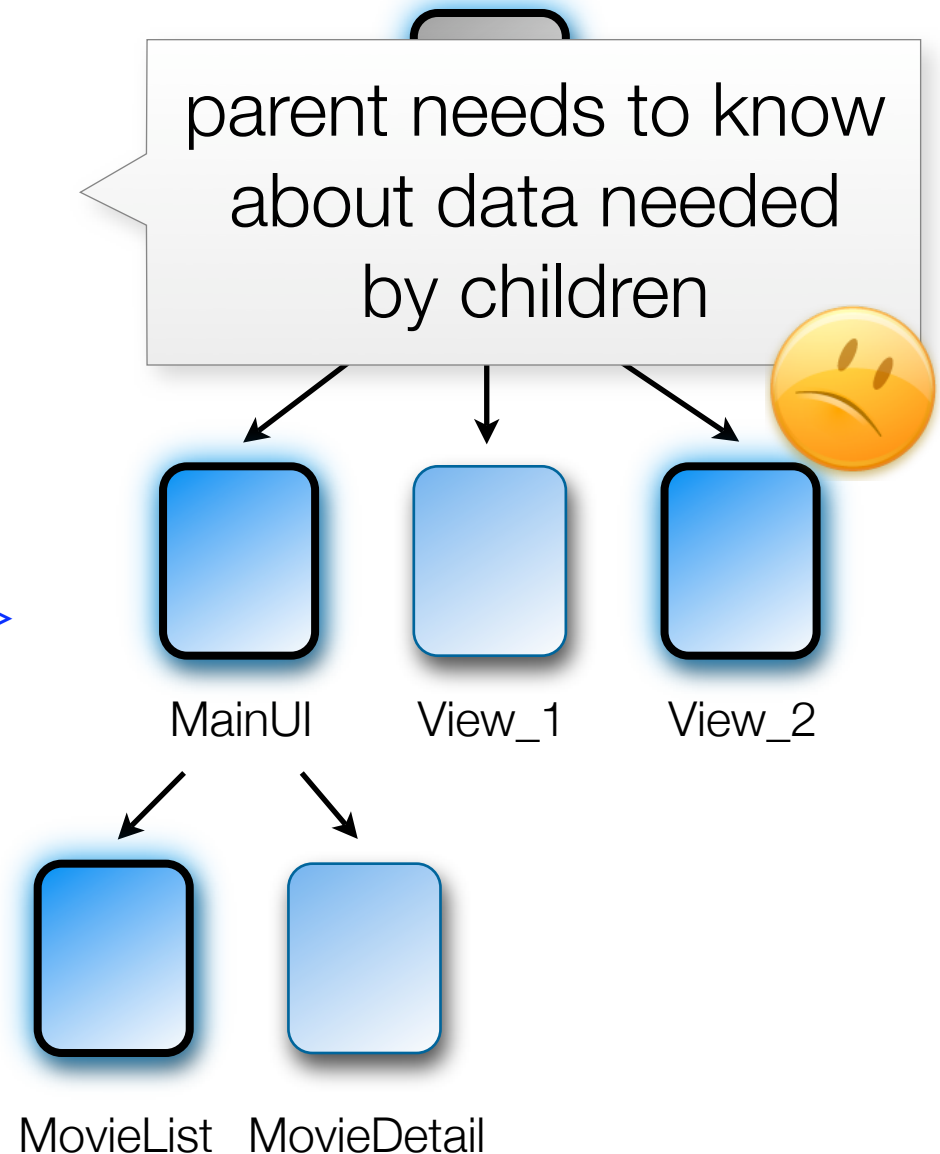


Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

```
public var movies:ArrayCollection;  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

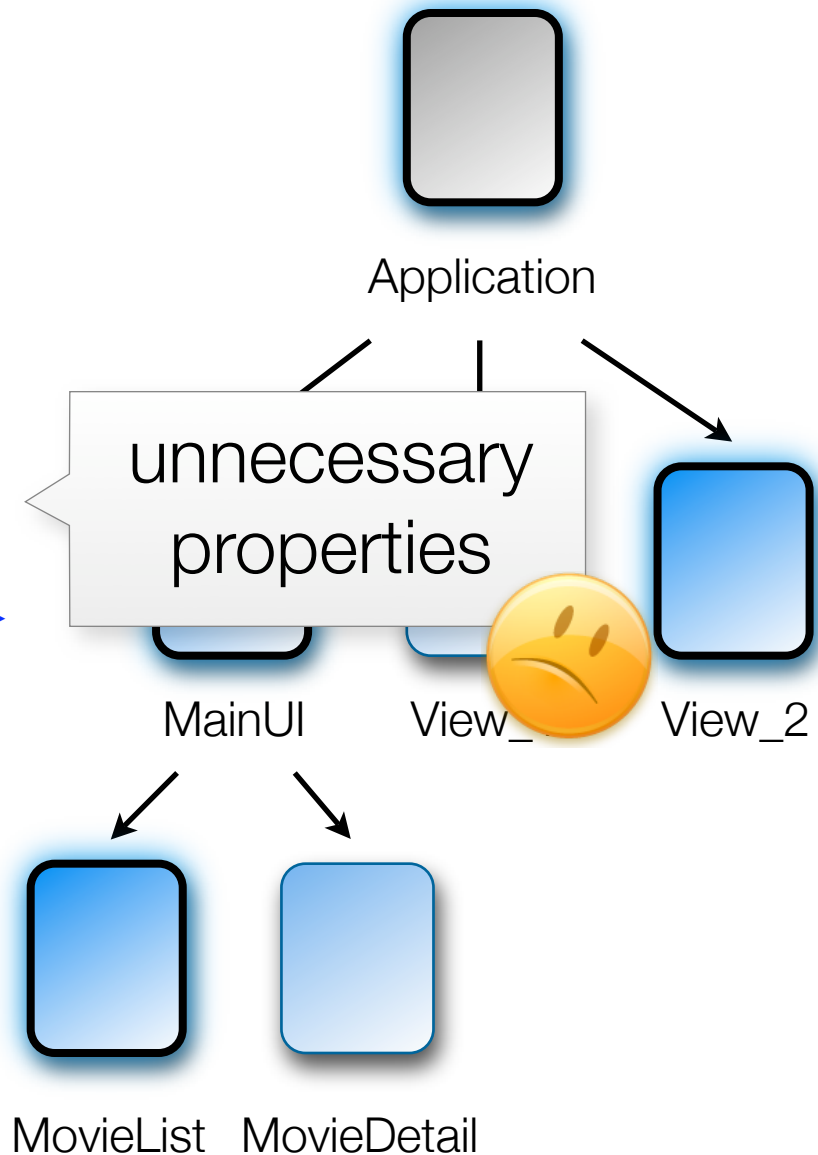


Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

```
public var movies:ArrayCollection;  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```

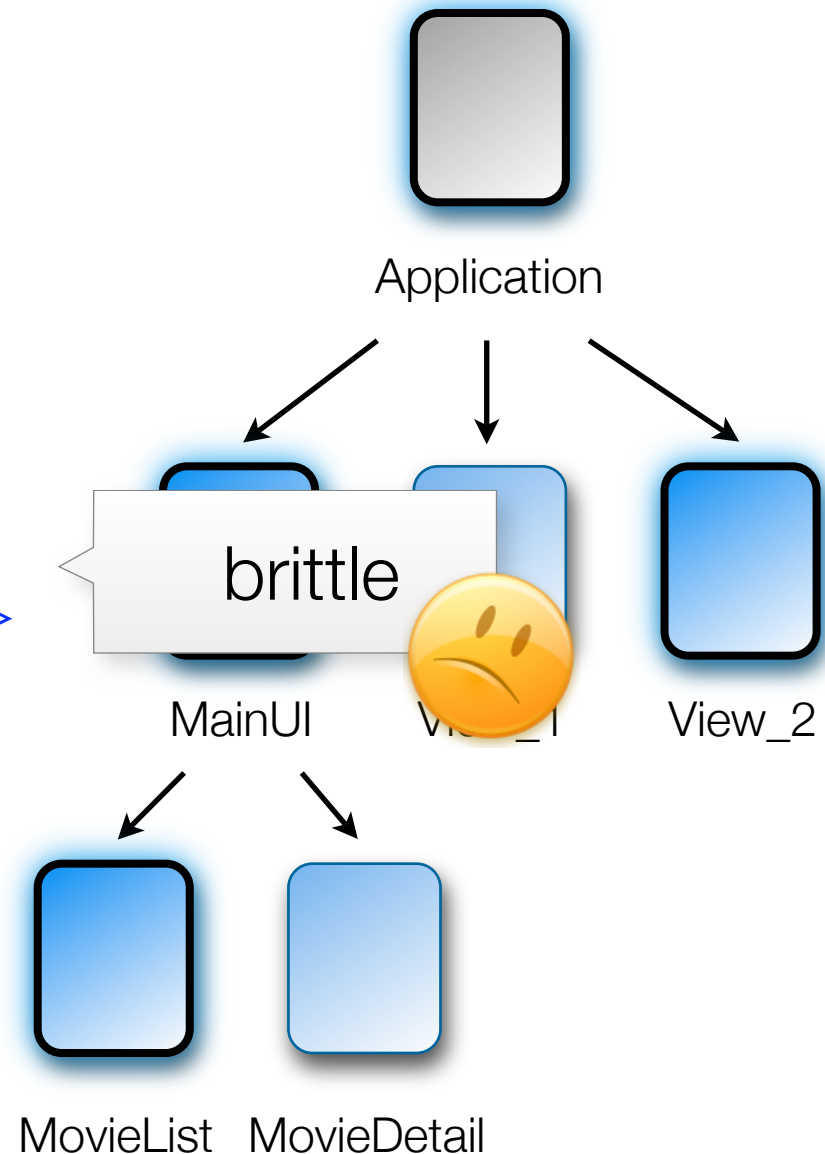


Passing data from parent to children

```
<views:MainUI movies="{movies}" />  
<views:View_1 />  
<views:View_2 movies="{movies}" />
```

```
public var movies:ArrayCollection;  
<views:MovieList movies="{movies}" />
```

```
[Bindable]  
public var movies:ArrayCollection;
```



Using Injection

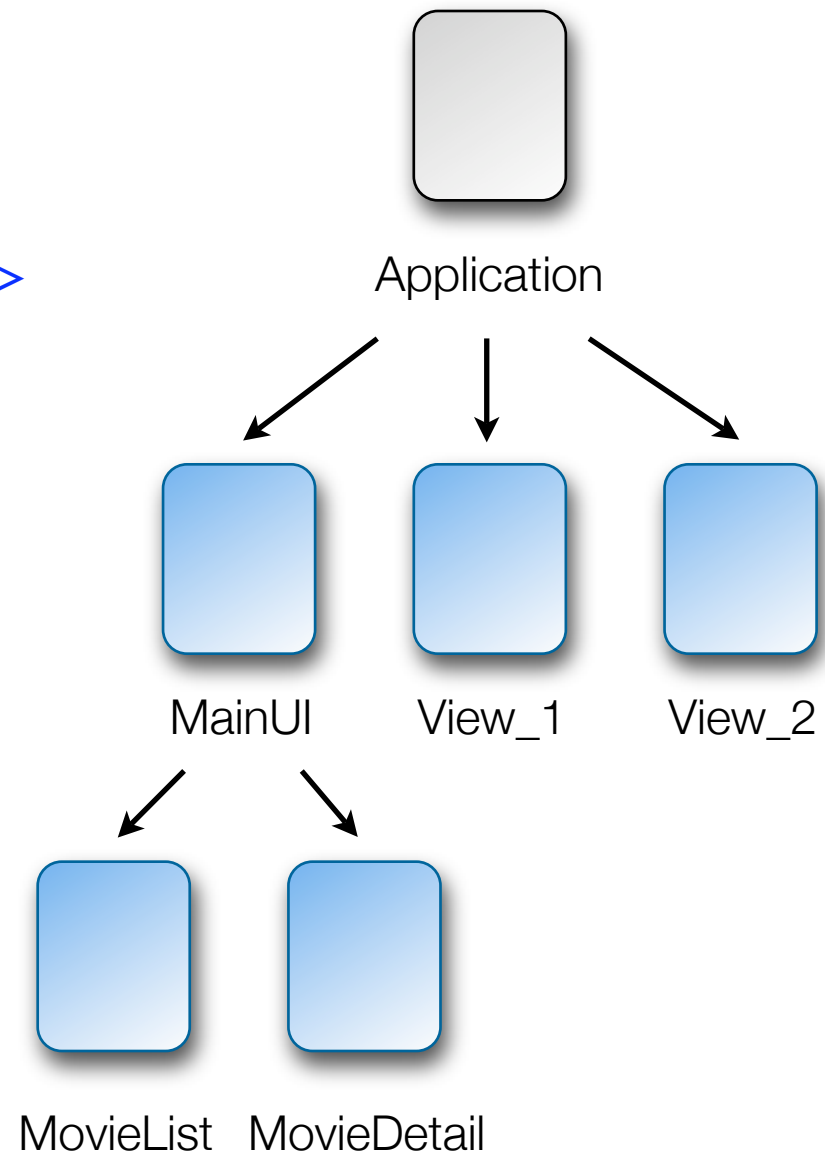
```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
  </Injectors>
```

```
</EventManager>
```



Using Injection

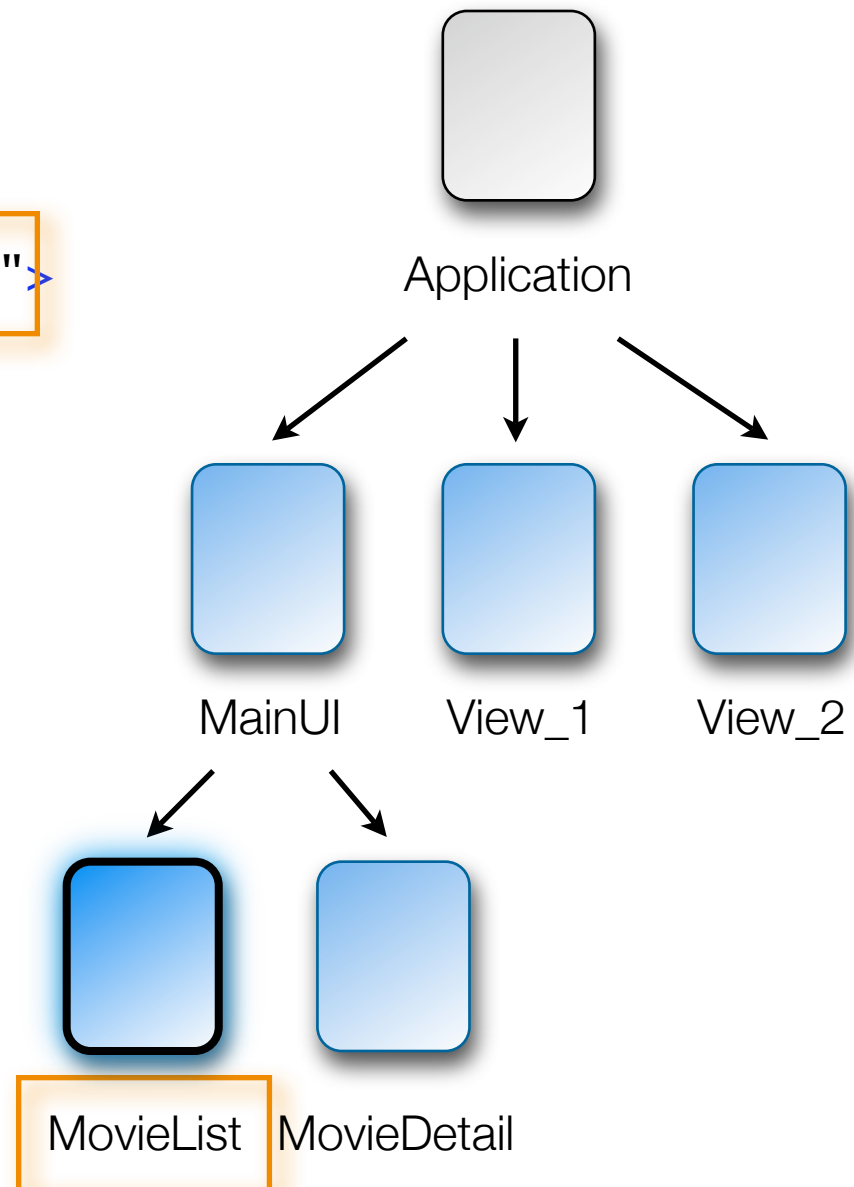
```
<EventManager ...>
```

```
<Injectors target="{MovieList}">
```

```
<PropertyInjector  
  source="{MoviesManager}"  
  sourceKey="movies"  
  targetKey="movies" />
```

```
</Injectors>
```

```
</EventManager>
```



Using Injection

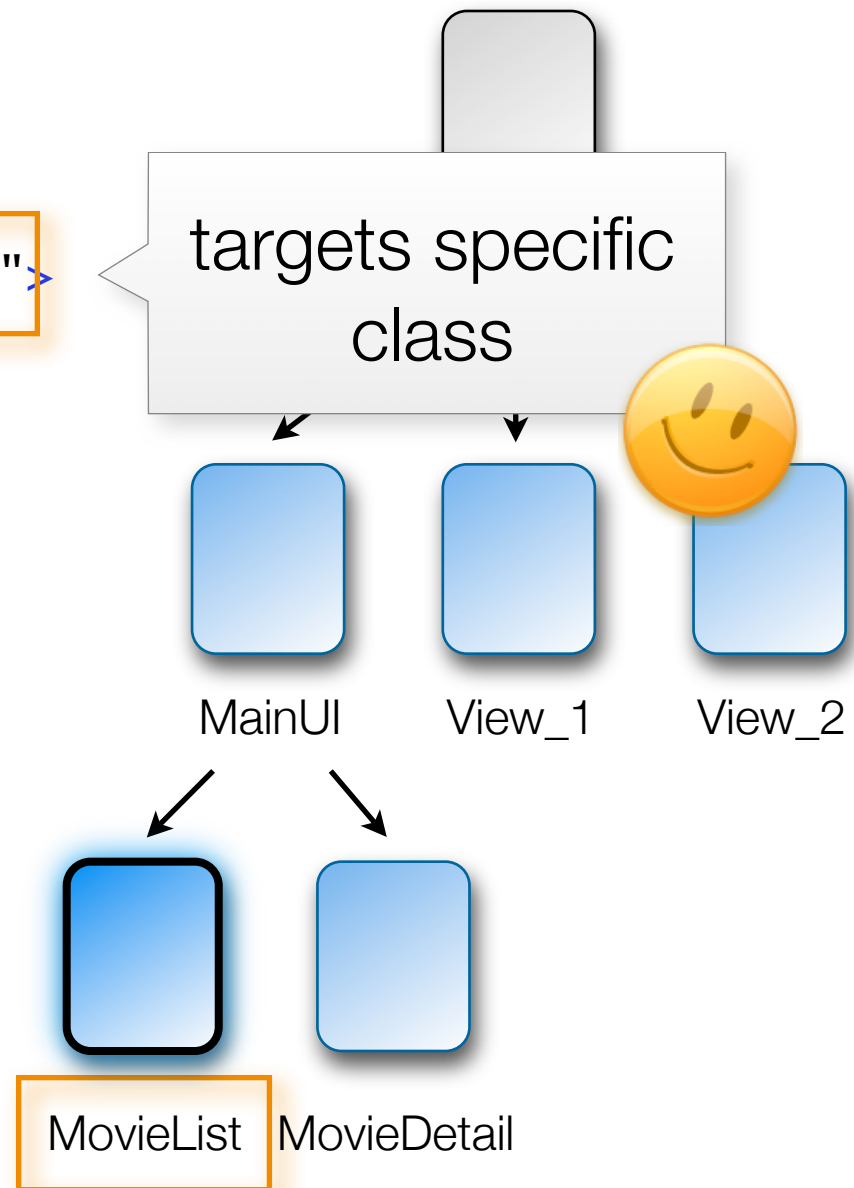
```
<EventManager ...>
```

```
<Injectors target="{MovieList}">
```

```
<PropertyInjector  
  source="{MoviesManager}"  
  sourceKey="movies"  
  targetKey="movies" />
```

```
</Injectors>
```

```
</EventManager>
```



Using Injection

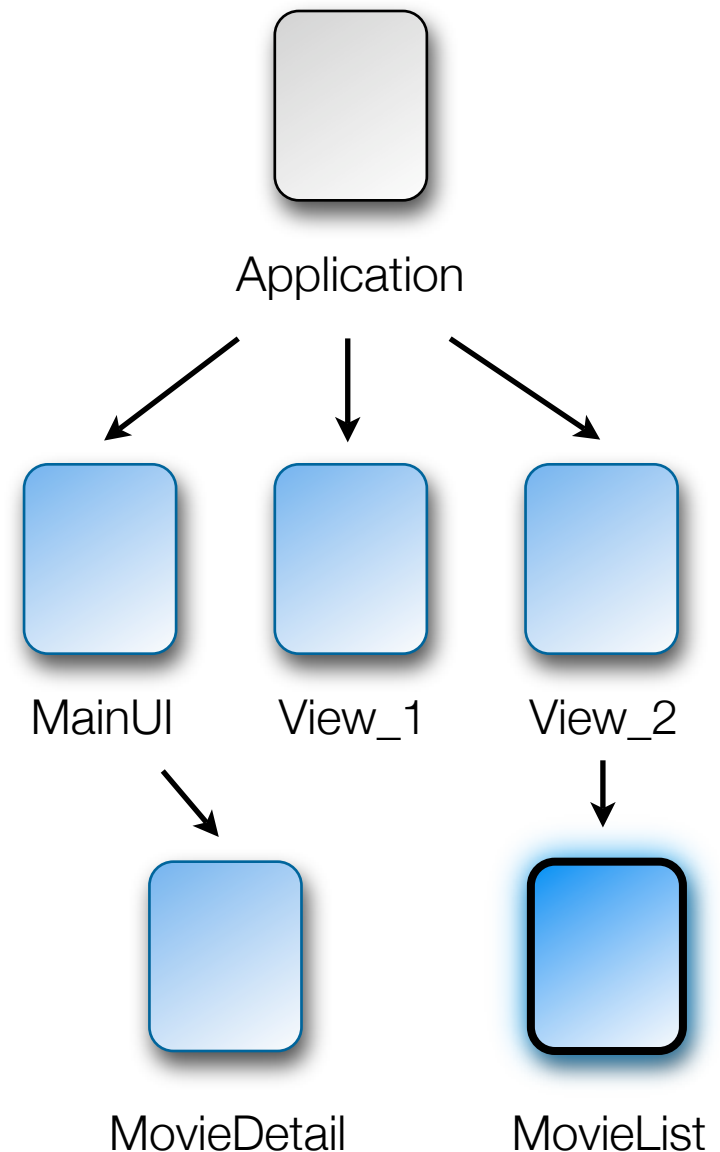
```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
  </Injectors>
```

```
</EventManager>
```



Using Injection

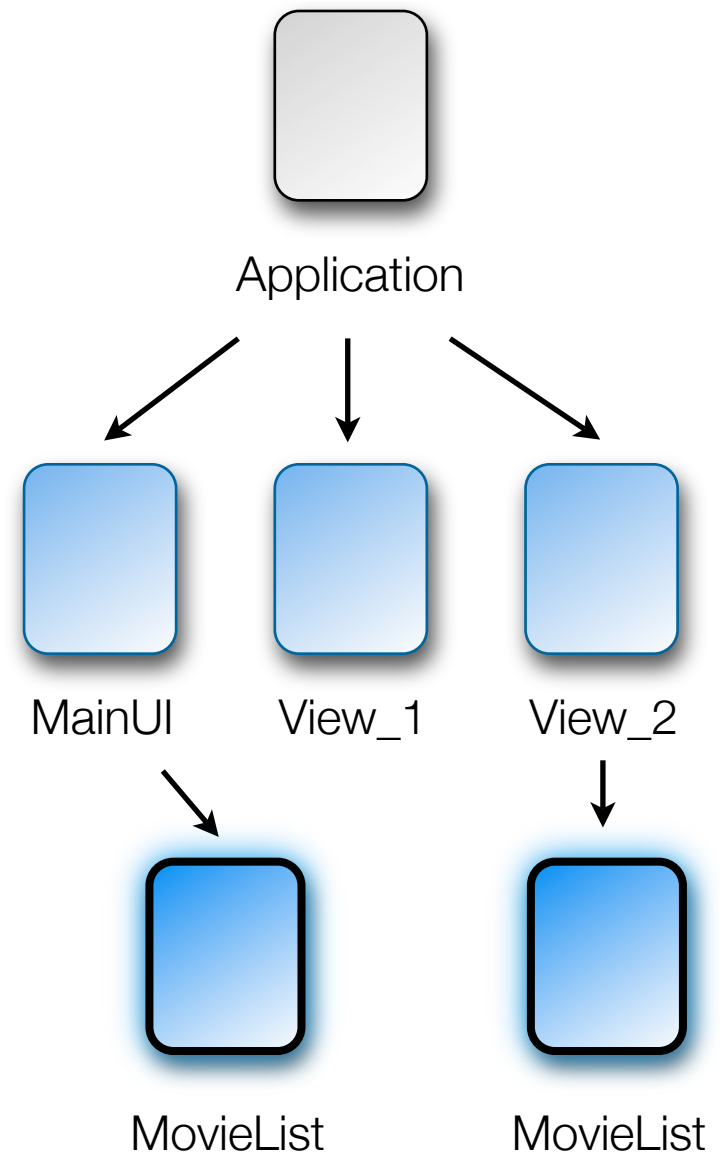
```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
  </Injectors>
```

```
</EventManager>
```



Using Injection

```
<EventManager ...>
```

```
<Injectors target="{MovieList}">
```

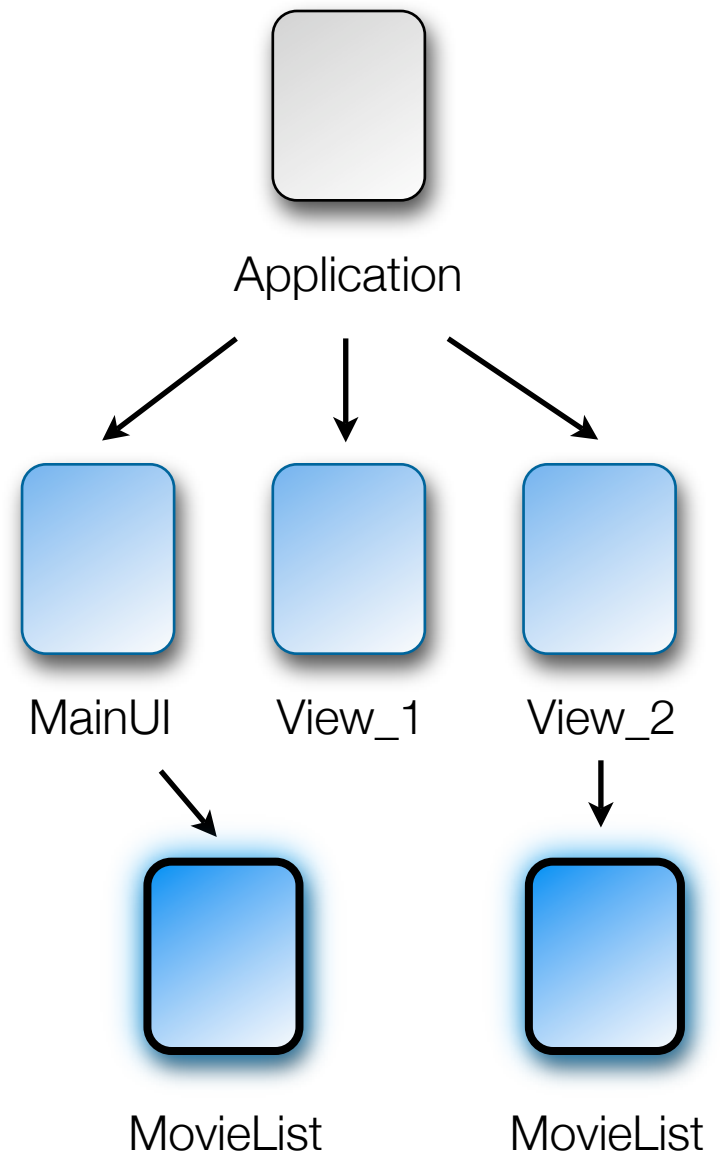
```
<Product
```

parents do not
need to know what
children need



```
</Injectors>
```

```
</EventManager>
```



Using Injection

```
<EventManager ...>
```

```
<Injectors target="{MovieList}">
```

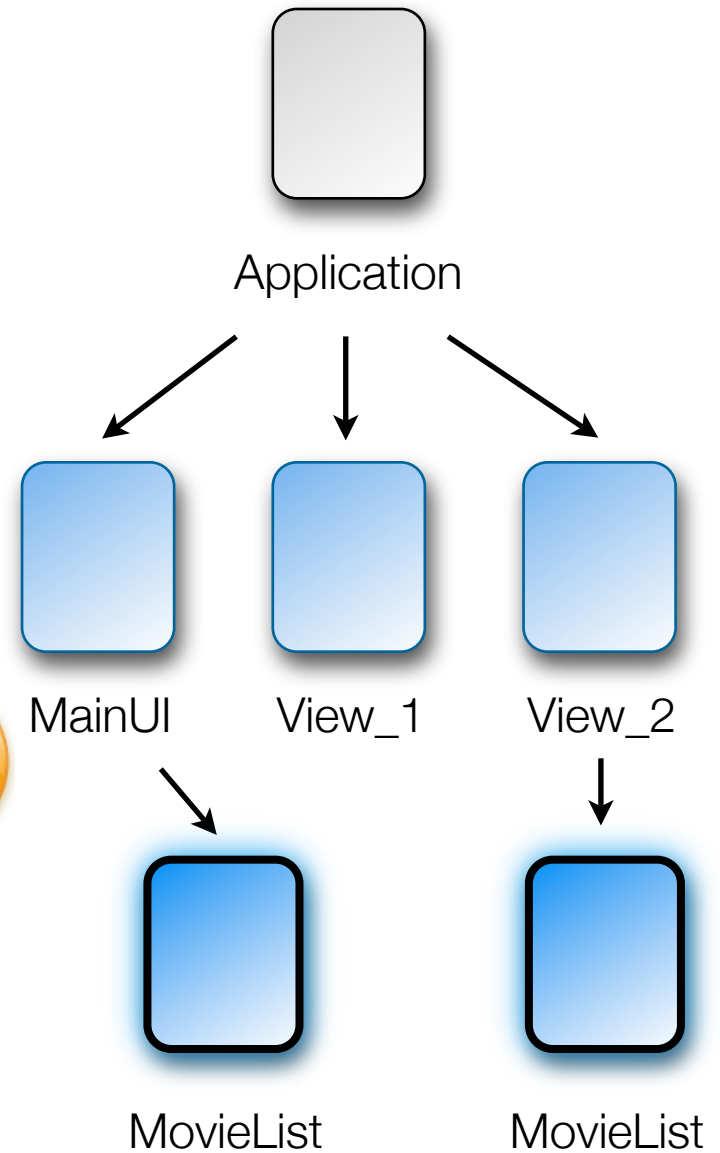
```
<Property
```

```
source  
source  
target
```

no unnecessary
properties in
parents

```
</Injectors>
```

```
</EventManager>
```



Using a singleton

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
<mx:Script>
```

```
<![CDATA[
```

```
import com.movix.model.MyModelLocator;
```

```
[Bindable]
```

```
private var model:MyModelLocator = MyModelLocator.getInstance();
```

```
]]>
```

```
</mx:Script>
```

```
<mx>List id="list" dataProvider="{model.movies}" />
```

```
</mx:Panel>
```

Using a singleton

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">

  <mx:Script>
    <![CDATA[
      import com.movix.model.MyModelLocator;

      [Bindable]
      private var model:MyModelLocator = MyModelLocator.getInstance();
    ]]>
  </mx:Script>

  <mx:List id="list" dataProvider="{model.movies}" />

</mx:Panel>
```

Using a singleton

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
<mx:Script>
```

```
<![CDATA[
```

```
import com.movix.model.MyModelLocator;
```

```
[Bindable]
```

```
private var model:MyModelLocator =
```

```
]]>
```

```
</mx:Script>
```

```
<mx>List id="list" dataProvider="{model.movies}" />
```

```
</mx:Panel>
```

view is dependent
on singleton for data ;
= hard to test



Using a singleton

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
<mx:Script>
```

```
<![CDATA[
```

```
import com.movix.model.MyModelLocator;
```

```
[Bindable]
```

```
private var model:MyModelLocator = MyModelLocator.getInstance();
```

```
]]>
```

```
</mx:Script>
```

```
<mx>List id="list" dataProvider="{model.movies}" />
```

```
</mx:Panel>
```

tied to particular
model class



Using a singleton

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
  <mx:Script>
```

```
    <![CDATA[
```

```
      import com.movix.model.MyModelLocator;
```

```
      [Bindable]
```

```
      private var model:MyModelLocator = MyModelLocator.getInstance();
```

```
    ]]>
```

```
  </mx:Script>
```

```
  <mx>List id="list" dataProvider="{model.movies}"
```

```
</mx:Panel>
```

dependent on
particular model
properties and
implementation



Using Injection

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">  
  
  <mx:Script>  
    <![CDATA[  
      import mx.collections.ArrayCollection;  
  
      [Bindable]  
      public var movies:ArrayCollection;  
  
    ]]>  
  </mx:Script>  
  
  <mx>List id="list" dataProvider="{movies}" />  
  
</mx:Panel>
```

view doesn't
know where data
comes from



Using Injection

```
<mx:Panel xmlns:mx="http://www.adobe.com/2006/..">
```

```
  <mx:Script>
```

```
    <![CDATA[
```

```
      import mx.collections.ArrayCollection;
```

```
      [Bindable]
```

```
      public var movies:ArrayCollection;
```

```
    ]]>
```

```
  </mx:Script>
```

```
  <mx>List id="list" dataProvider="{movies}" />
```

```
</mx:Panel>
```

independent from
framework



Using Injection

```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
  </Injectors>
```

```
</EventManager>
```

view is not
dependent on model
class



Using Injection

```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector  
      source="{MoviesManager}"  
      sourceKey="movies"  
      targetKey="movies" />
```

```
  </Injectors>
```

```
</EventManager>
```

view does not know
about specific model
properties



Using Injection

```
<EventManager ...>
```

```
  <Injectors target="{MovieList}">
```

```
    <PropertyInjector
```

```
      source="{MoviesManager}"
```

```
      sourceKey="movies"  
      targetKey="movies" />
```

```
  </Injectors>
```

```
</EventManager>
```

takes advantage of
bindings



Injecting responses

(Callbacks)



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

Your rating:



Sending review...



Nice movie!

Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was ve
well done. The special effects and the acting is very good. I recommend this movie to everyon
not just comic book fans.

send
notification

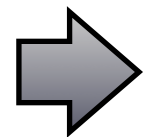
Your rating:



Sending review...



Nice movie!



Cancel

Submit



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

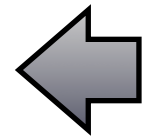
Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

Your rating:



Nice movie!

Review added



receive
notification

Cancel

Submit



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
        event.review = new Review(movieID,  
            rating.value,  
            textArea.text);  
        dispatchEvent(event);  
        showLoadingBar();  
    }  
</mx:VBox>
```



ReviewForm

<mx

```
public class ReviewEvent
    extends flash.events.Event {

    public static const ADD:String = "addReviewEvent";

    public var review:Review;

    public function ReviewEvent(
        type:String,
        bubbles:Boolean=true,
        cancelable:Boolean=false) {

        super(type, bubbles, cancelable);

    }
}
```

dispatchEvent(event);

showLoadingBar();

}

</mx:VBox>



ReviewForm

<mx

```
public class ReviewEvent
    extends flash.events.Event {
    public static const ADD:String = "addReviewEvent";
    public var review:Review;
    public function ReviewEvent(
        type:String,
        bubbles:Boolean=true,
        cancelable:Boolean=false) {
        super(type, bubbles, cancelable);
    }
}
```

dispatchEvent(event);

showLoadingBar();

}

</mx:VBox>



ReviewForm

<mx

```
public class ReviewEvent
    extends flash.events.Event {
    public static const ADD:String = "addReviewEvent";
    public var review:Review;

    public function ReviewEvent(
        type:String,
        bubbles:Boolean=true,
        cancelable:Boolean=false) {
        super(type, bubbles, cancelable);
    }
}
```

dispatchEvent(event);

showLoadingBar();

}

</mx:VBox>



ReviewForm

<mx

```
public class ReviewEvent
    extends flash.events.Event {

    public static const ADD:String = "addReviewEvent";

    public var review:Review;

    public function ReviewEvent(
        type:String,
        bubbles:Boolean=true,
        cancelable:Boolean=false) {

        super(type, bubbles, cancelable);

    }
}
```

dispatchEvent(event);

showLoadingBar();

}

</mx:VBox>



ReviewForm

<mx

```
public class ReviewEvent
    extends flash.events.Event {

    public static const ADD:String = "addReviewEvent";

    public var review:Review;

    public function ReviewEvent(
        type:String,
        bubbles:Boolean=true,
        cancelable:Boolean=false) {

        super(type, bubbles, cancelable);

    }
}
```

dispatchEvent(event);

showLoadingBar();

}

</mx:VBox>



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
        event.review = new Review(movieID,  
            rating.value,  
            textArea.text);  
        dispatchEvent(event);  
        showLoadingBar();  
    }  
</mx:VBox>
```



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
  
        event.review = new Review(movieID,  
                                rating.value,  
                                textArea.text);  
  
        dispatchEvent(event);  
  
        showLoadingBar();  
  
    }  
</mx:VBox>
```



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
  
        event.review = new Review(movieID,  
            rating.value,  
            textArea.text);  
  
        dispatchEvent(event);  
  
        showLoadingBar();  
    }  
</mx:VBox>
```



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
  
        event.review = new Review(movieID,  
            rating.value,  
            textArea.text);  
  
        dispatchEvent(event);  
        showLoadingBar();  
    }  
</mx:VBox>
```

Event Bus

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
  
        event.review = new Review(movieID,  
            rating.value,  
            textArea.text);  
  
        dispatchEvent(event);  
  
        showLoadingBar();  
  
    }  
</mx:VBox>
```

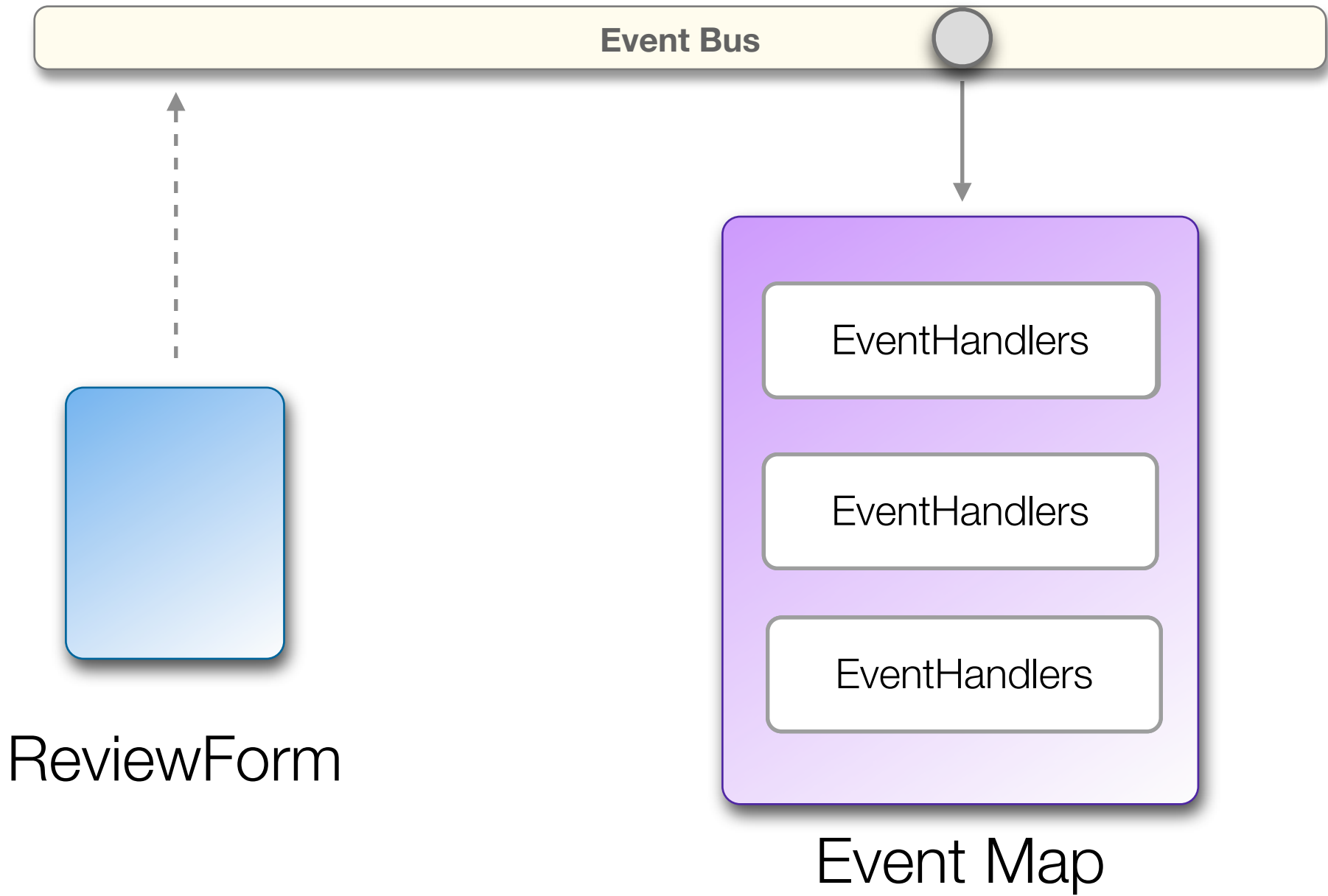
ReviewForm

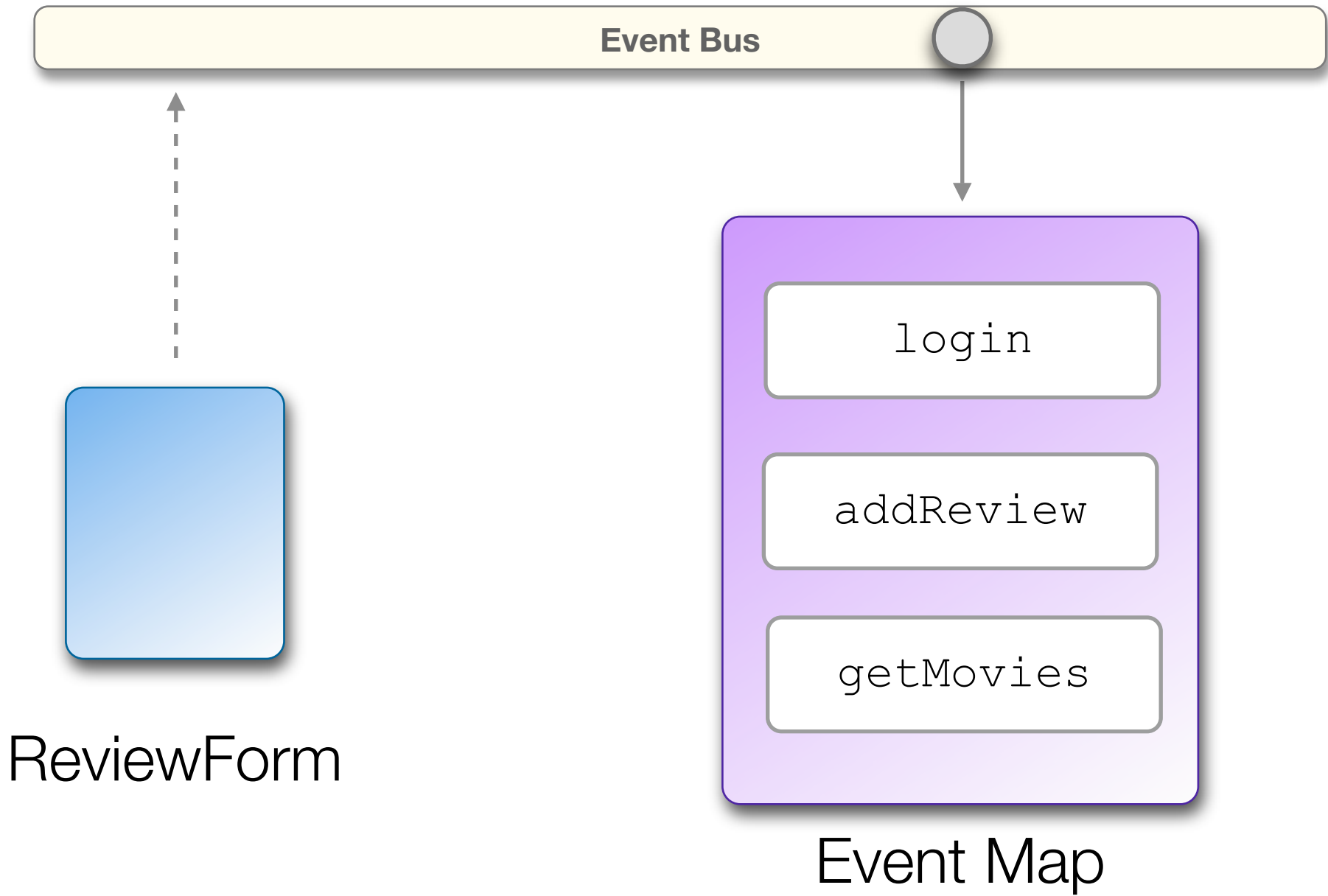




ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
        //validate data, if ok, then:  
  
        var event:ReviewEvent =  
            new ReviewEvent(ReviewEvent.ADD);  
  
        event.review = new Review(movieID,  
            rating.value,  
            textArea.text);  
  
        dispatchEvent(event);  
  
        showLoadingBar();  
  
    }  
</mx:VBox>
```



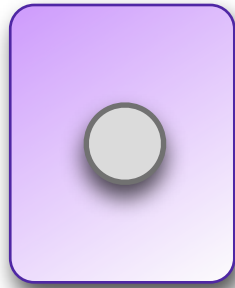




ReviewForm



event



EventMap

✓ Call the server

✓ Perform other actions

✓ Remove loading animation

Event Bus

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">
```

```
  private function submitReview():void {
```

```
    //validate data, if ok, create event,  
    // then dispatch it:
```

```
    dispatchEvent(event);
```

```
    showLoadingBar();
```

```
  }
```

```
  private function handleReviewResult  
  (result:Boolean):void {
```

```
    // remove the loading bar, change the label, etc
```

```
  }
```

```
</mx:VBox>
```

ReviewForm



Event Bus

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">
```

```
    private function submitReview():void {
```

```
        //validate data, if ok, create event,  
        // then dispatch it:
```

```
        dispatchEvent(event);
```

```
        showLoadingBar();
```

```
    }
```

```
    private function handleReviewResult  
    (result:Boolean):void {
```

```
        // remove the loading bar, change the label, etc
```

```
    }
```

```
</mx:VBox>
```

ReviewForm



Event Bus

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">
```

```
  private function submitReview():void {
```

```
    //validate data, if ok, create event,  
    // then dispatch it:
```

```
    dispatchEvent(event);
```

```
    showLoadingBar();
```

```
  }
```

```
  private function handleReviewResult  
  (result:Boolean):void {
```

```
    // remove the loading bar, change the label, etc
```

```
  }
```

```
</mx:VBox>
```

ReviewForm

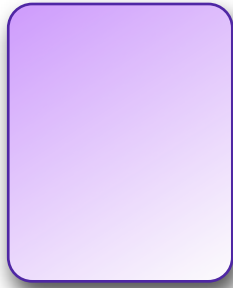




ReviewForm



event



EventMap



✓ Call the server

✓ Perform other actions

✓ Remove loading animation



MainEventMap.mxml



```
<?xml version="1.0" encoding="utf-8"?>  
<EventMap ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
  </EventHandlers>
```

```
</EventMap>
```

MainEventMap.mxml



```
<?xml version="1.0" encoding="utf-8"?>  
<EventMap ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
  </EventHandlers>
```

```
</EventMap>
```



```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

✓ Call the server

✓ Perform other actions

(after server responds)

```
</EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
<EventHandlers type="{ReviewEvent.ADD}">
```

```
<RemoteObjectInvoker >
```

```
</EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
    <RemoteObjectInvoker
```

```
      source="services.ReviewService" ...
```

```
      method="addReview" arguments="{event.review}" >
```

```
  </EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>  
<EventManager ...>
```

```
<EventHandlers type="{ReviewEvent.ADD}">
```

```
<RemoteObjectInvoker
```

```
source="{services.ReviewService" ...
```

```
method="{addReview}" arguments="{event.review}" >
```



```
</EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
    <RemoteObjectInvoker
```

```
      source="services.ReviewService" ...
```

```
      method="addReview" arguments="{event.review}" >
```

✓ Perform other actions

```
  </EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
    <RemoteObjectInvoker
```

```
      source="services.ReviewService" ...
```

```
      method="addReview" arguments="{event.review}" >
```

```
        <resultHandlers>
```

 Perform other actions

```
        </resultHandlers>
```

```
    </RemoteObjectInvoker>
```

```
  </EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
    <RemoteObjectInvoker
```

```
      source="services.ReviewService" ...
```

```
      method="addReview" arguments="{event.review}" >
```

```
    <resultHandlers>
```

```
      <Callback method="handleReviewResult"  
        arguments="{responseObject}" />
```

```
    </resultHandlers>
```

```
  </RemoteObjectInvoker>
```

```
</EventHandlers>
```

```
</EventManager>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type="{ReviewEvent.ADD}">
```

```
    <RemoteObjectInvoker
```

```
      source="services.ReviewService" ...
```

```
      method="addReview" arguments="{event.review}" >
```

```
    <resultHandlers>
```

```
      <Callback method="handleReviewResult"
        arguments="{responseObject}" />
```

```
    </resultHandlers>
```

```
  </EventHandlers>
```

```
</EventManager>
```

```
</EventManager>
```



```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type=
```

```
    <RemoteObjectIn
```

```
      source='
```

```
      method="addReview" arguments="{event.review}" >
```

```
  <resultHandlers>
```

```
    <Callback method="handleReviewResult"  
      arguments="{responseObject}" />
```

```
  </resultHandlers>
```

```
  </RemoteObjectInvoker>
```

```
</EventHandlers>
```

```
</EventManager>
```

```
private function handleReviewResult  
(result:Boolean):void {
```

```
    // remove the loading bar, change the label, etc
```

```
}
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EventManager ...>
```

```
  <EventHandlers type=
```

```
    <RemoteObjectIn
```

```
      source='
```

```
        method="addReview" arguments="{event.review}" >
```

```
  <resultHandlers>
```

```
    <Callback method="handleReviewResult"  
      arguments="{responseObject}" />
```

```
  </resultHandlers>
```

```
  </RemoteObjectInvoker>
```

```
</EventHandlers>
```

```
</EventManager>
```

```
private function handleReviewResult  
(result:Boolean):void {
```

```
    // remove the loading bar, change the label, etc
```

```
}
```



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">  
    private function submitReview():void {  
  
        //validate data, if ok, create event,  
        // then dispatch it:  
  
        dispatchEvent(event);  
  
        showLoadingBar();  
  
    }  
    private function handleReviewResult(result:Boolean)  
    :void {  
  
        // remove the loading bar, change the label, etc  
  
    }  
</mx:VBox>
```



ReviewForm

```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">
```

```
    private function submitReview():void {
```

```
        //validate data, if ok, create event,  
        // then dispatch it:
```

```
        dispatchEvent(event);
```

```
        showLoadingBar();
```

```
    }
```

```
    private function handleReviewResult(result:Boolean)  
    :void {
```

```
        // remove the loading bar, change the label, etc
```

```
    }
```

```
</mx:VBox>
```

independent from
framework



```
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">
```

```
    private function submitReview():void {
```

```
        //validate data, if ok, create event,  
        // then dispatch it:
```

```
        dispatchEvent(event);
```

```
        showLoadingBar();
```

```
    }
```

```
    private function handleReviewResult(result:Boolean)  
    :void {
```

```
        // remove the loading bar, change the label, etc
```

```
    }
```

```
</mx:VBox>
```

independent of
what happens
after event is
dispatched



Listeners



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz

Director Jean-Pierre Jeunet

Year 2001

Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal

Director Alejandro González Iñárritu

Year 2000

Category Drama



Add Movie

 Search



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz

Director Jean-Pierre Jeunet

Year 2001

Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal

Director Alejandro González Iñárritu

Year 2000

Category Drama



Add Movie

 Search



Spider Man II

Starring Tobey Maguire, Kirsten Dunst

Director Sam Raimi

Year 2004

Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

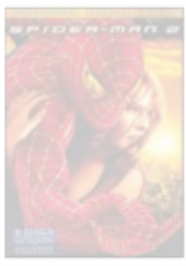
For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

Add Review

MovieList



Spider Man I

Starring Tobey Maguire, Kirsten Dunst
Director Sam Raimi
Year 2004
Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz
Director Jean-Pierre Jeunet
Year 2001
Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal
Director Alejandro González Iñárritu
Year 2000
Category Drama



Add Movie

 Search



Spider Man II

Starring Tobey Maguire, Kirsten Dunst
Director Sam Raimi
Year 2004
Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

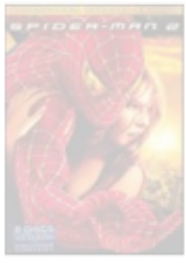
For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

Add Review

MovieList



Spider Man II

Starring Tobey Maguire, Kirsten Dunst
Director Sam Raimi
Year 2004
Category Action & Adventure, Sci-Fi & Fantasy



Amelie

Starring Audrey Tautou, Mathieu Kassovitz
Director Jean-Pierre Jeunet
Year 2001
Category Comedy, Foreign, Romance



Amores Perros

Starring Emilio Echeverria, Gael García Bernal
Director Alejandro González Iñárritu
Year 2000
Category Drama



Add Movie

 Search



MovieDetail

Director Sam Raimi
Year 2004
Category Action, Sci-Fi & Fantasy



★★★★☆ **Joe H.**

Another great spider-man movie. I have always been a spider man fan and this movie was very well done. The special effects and the acting is very good. I recommend this movie to everyone, not just comic book fans.

★★★★☆ **Christine M.**

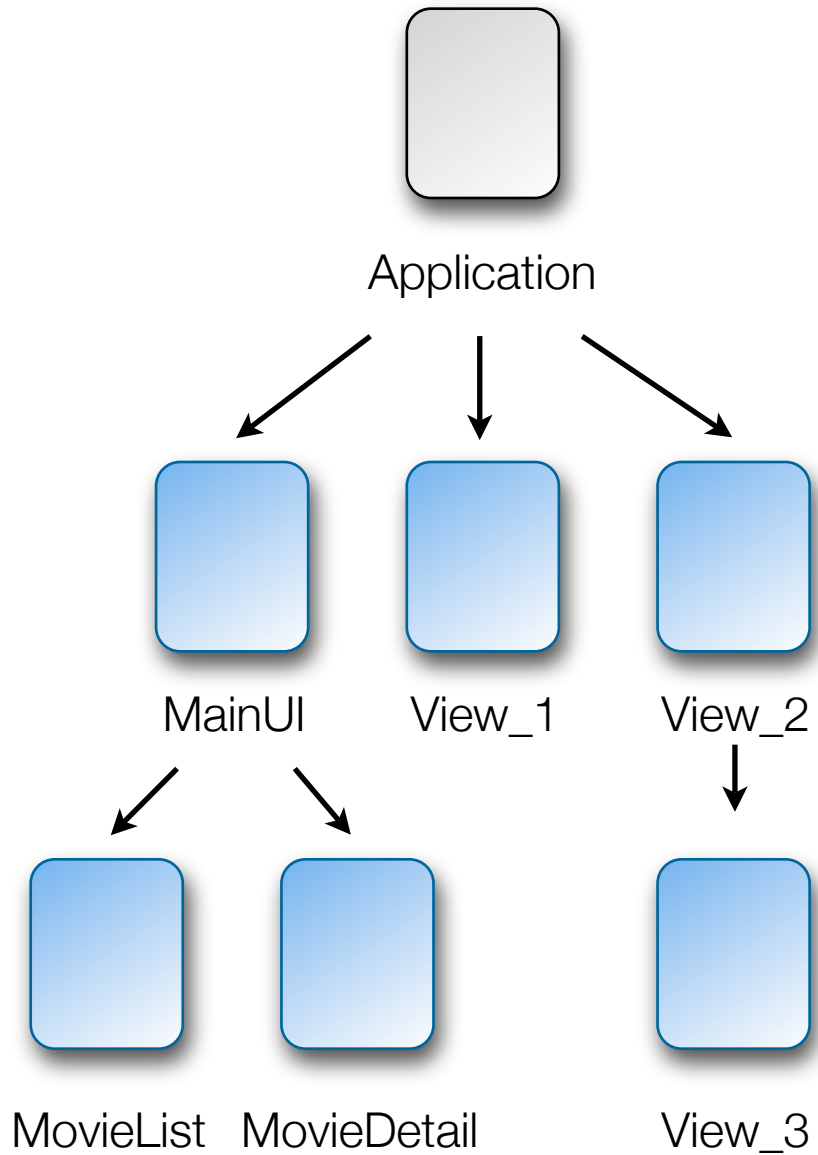
For a non-Spiderman fan, this was a very enjoyable movie that kept my attention. I am not much of a comic book movie fan and certainly not a cartoon fan, but the story and acting was good.

★★★★☆ **Matthew N.**

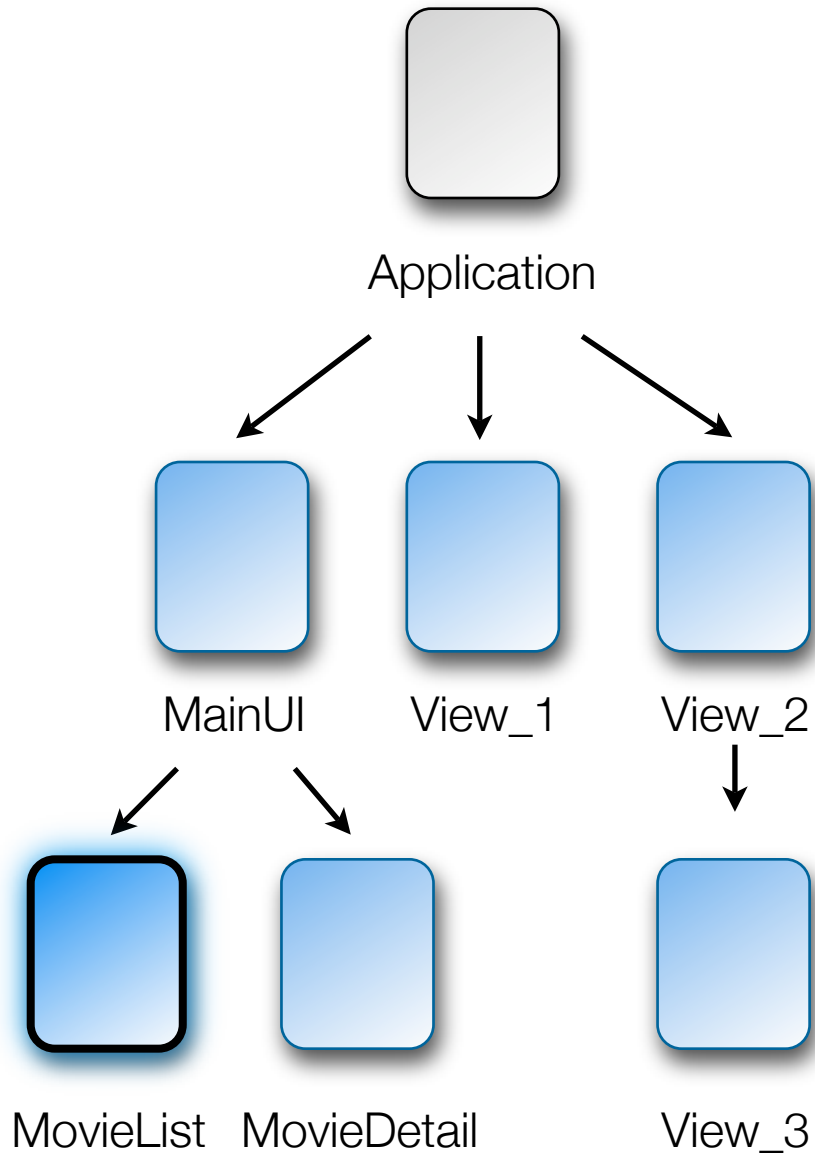
Spider-man once again proves to be the best superhero! Alfred Molina also portrayed the menacing Doc Ock very well!

Add Review

Bubbling up event

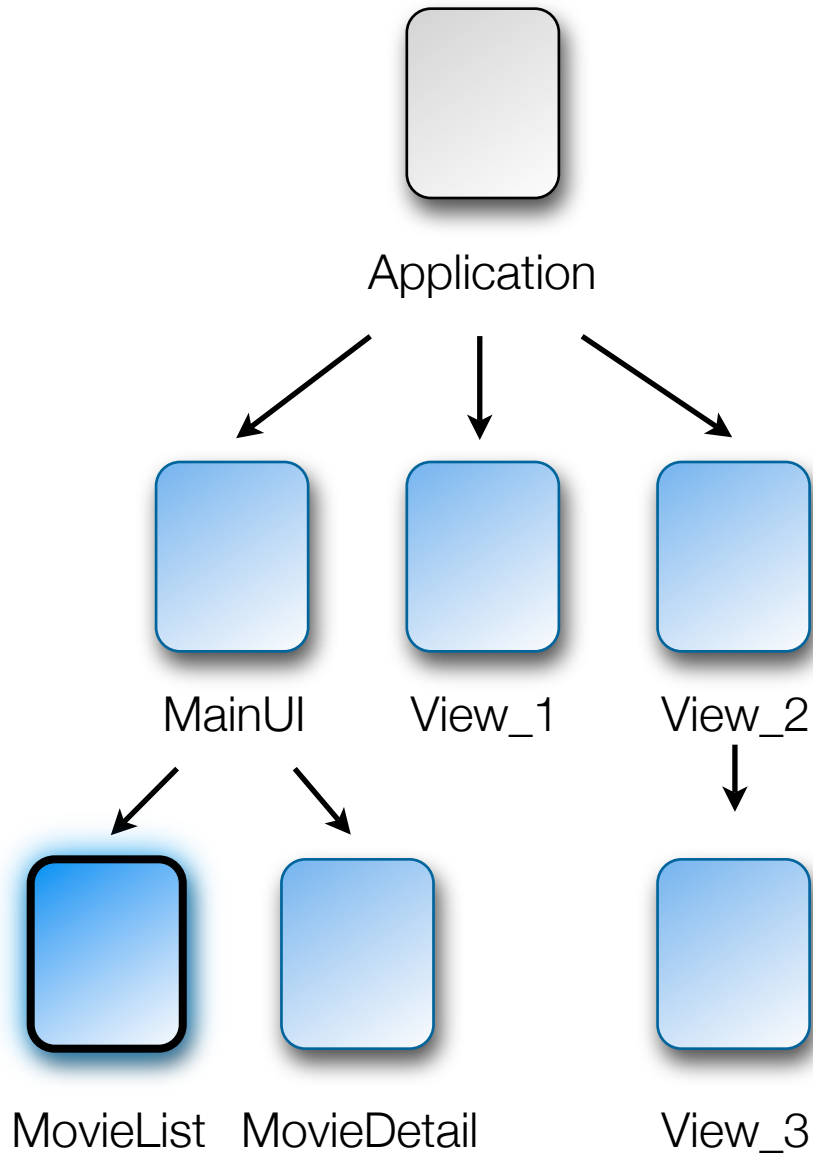


Bubbling up event



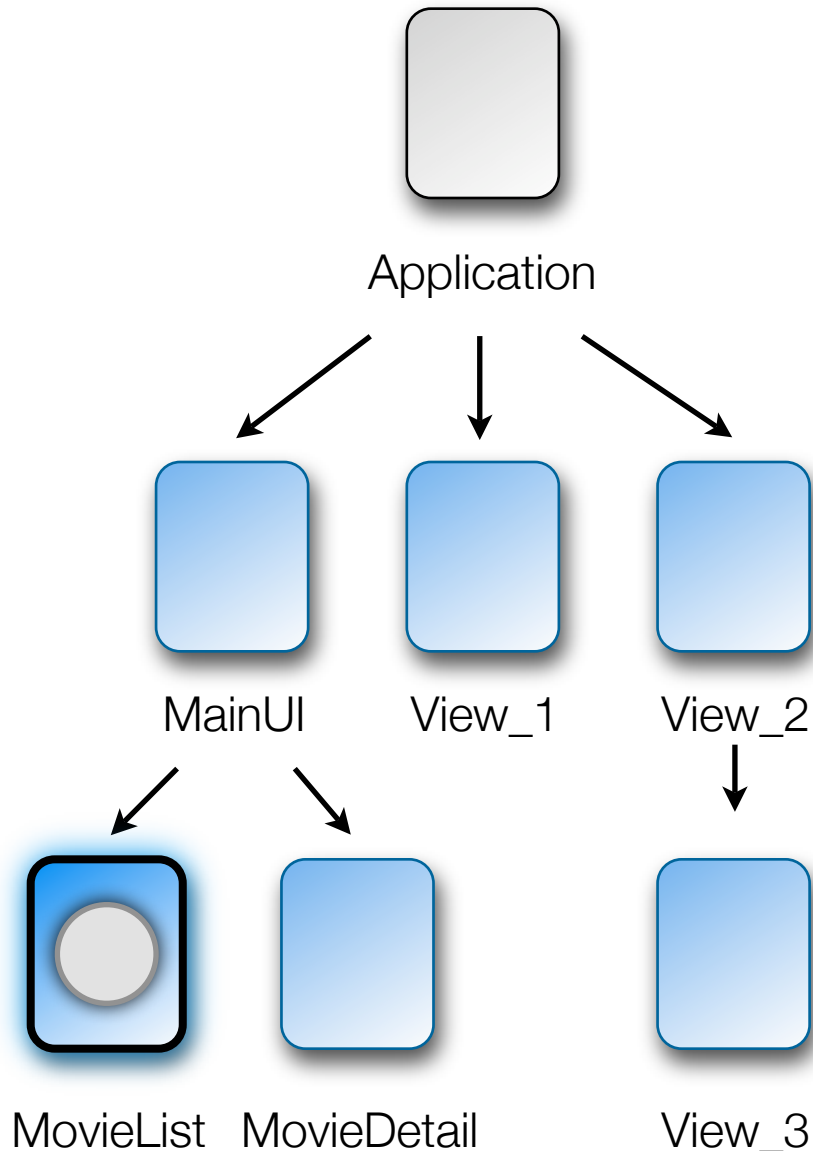
Bubbling up event

```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);  
  
event.movie = list.selectedItem as Movie;  
  
dispatchEvent(event);
```



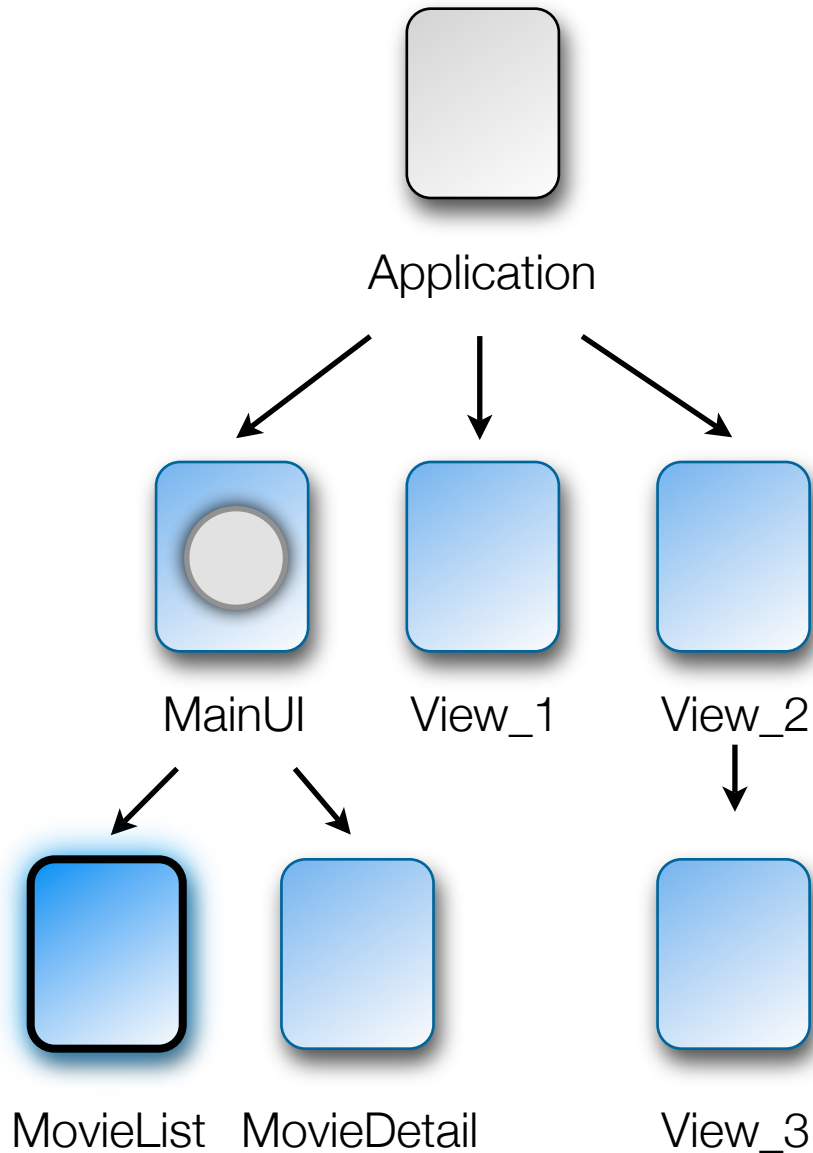
Bubbling up event

```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);  
  
event.movie = list.selectedItem as Movie;  
  
dispatchEvent(event);
```

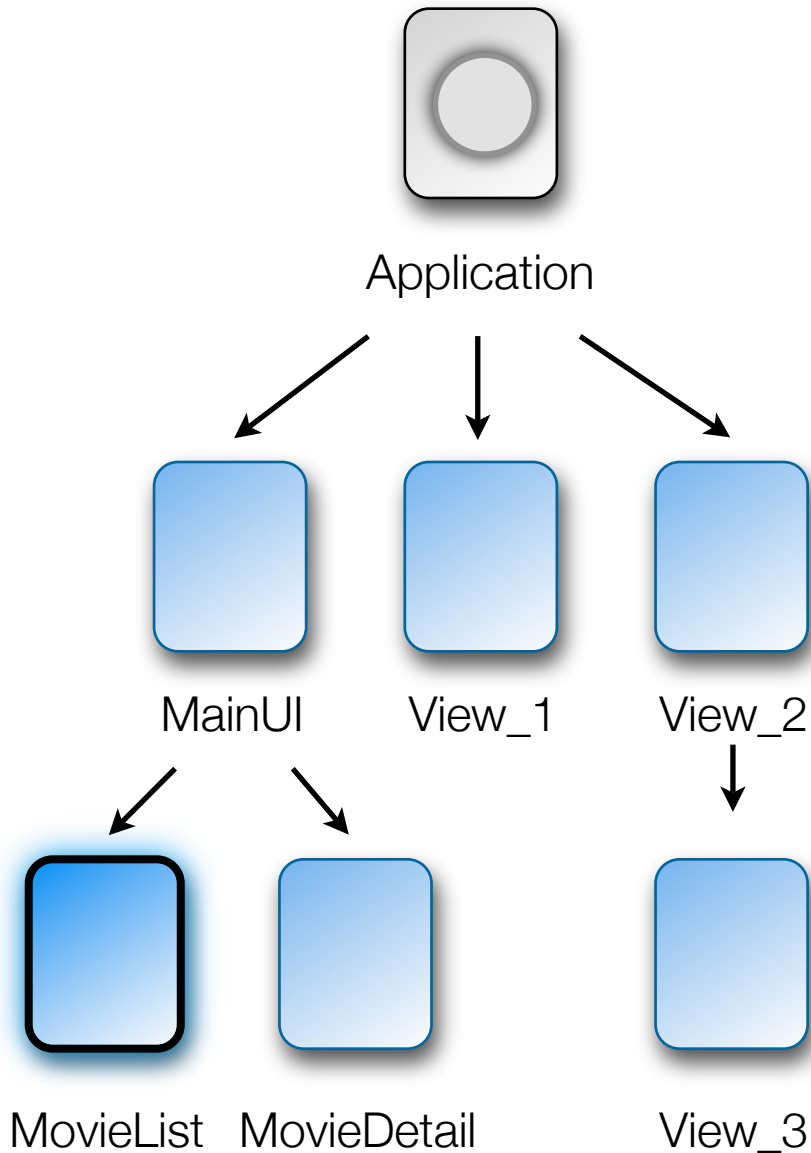


Bubbling up event

```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);  
  
event.movie = list.selectedItem as Movie;  
  
dispatchEvent(event);
```

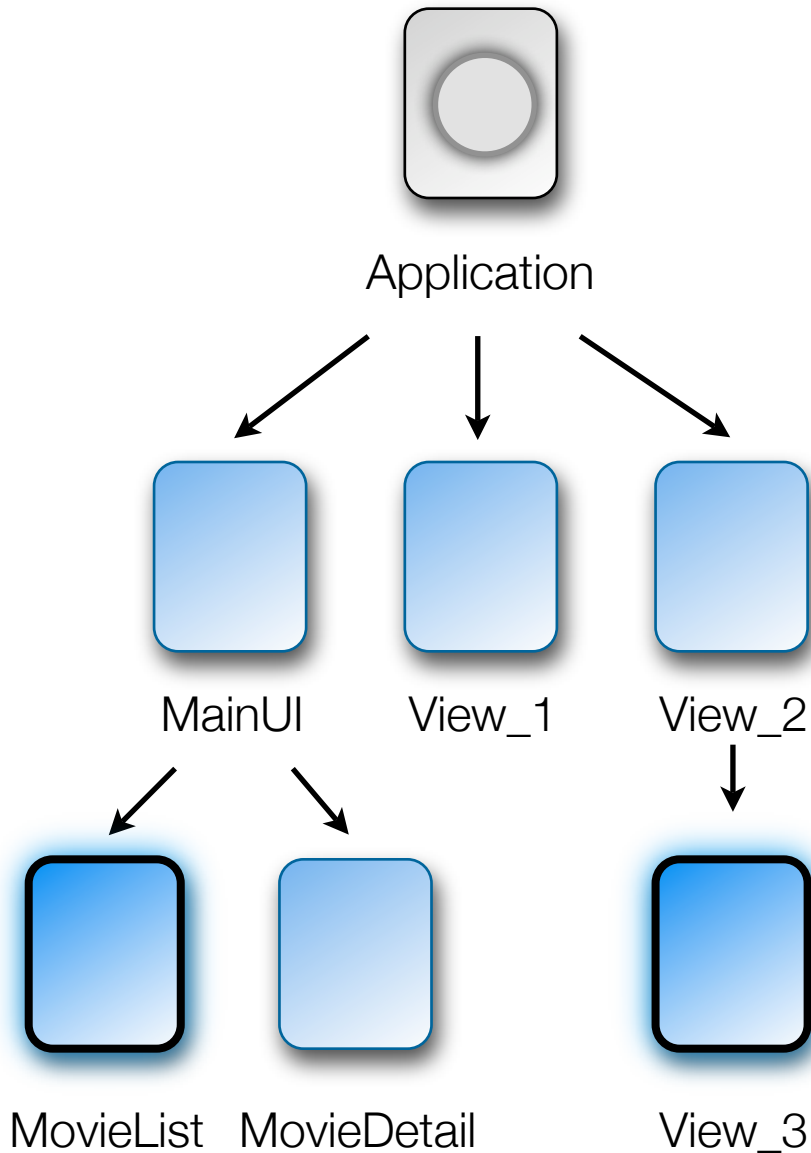


Bubbling up event



```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);  
  
event.movie = list.selectedItem as Movie;  
  
dispatchEvent(event);
```


Bubbling up event



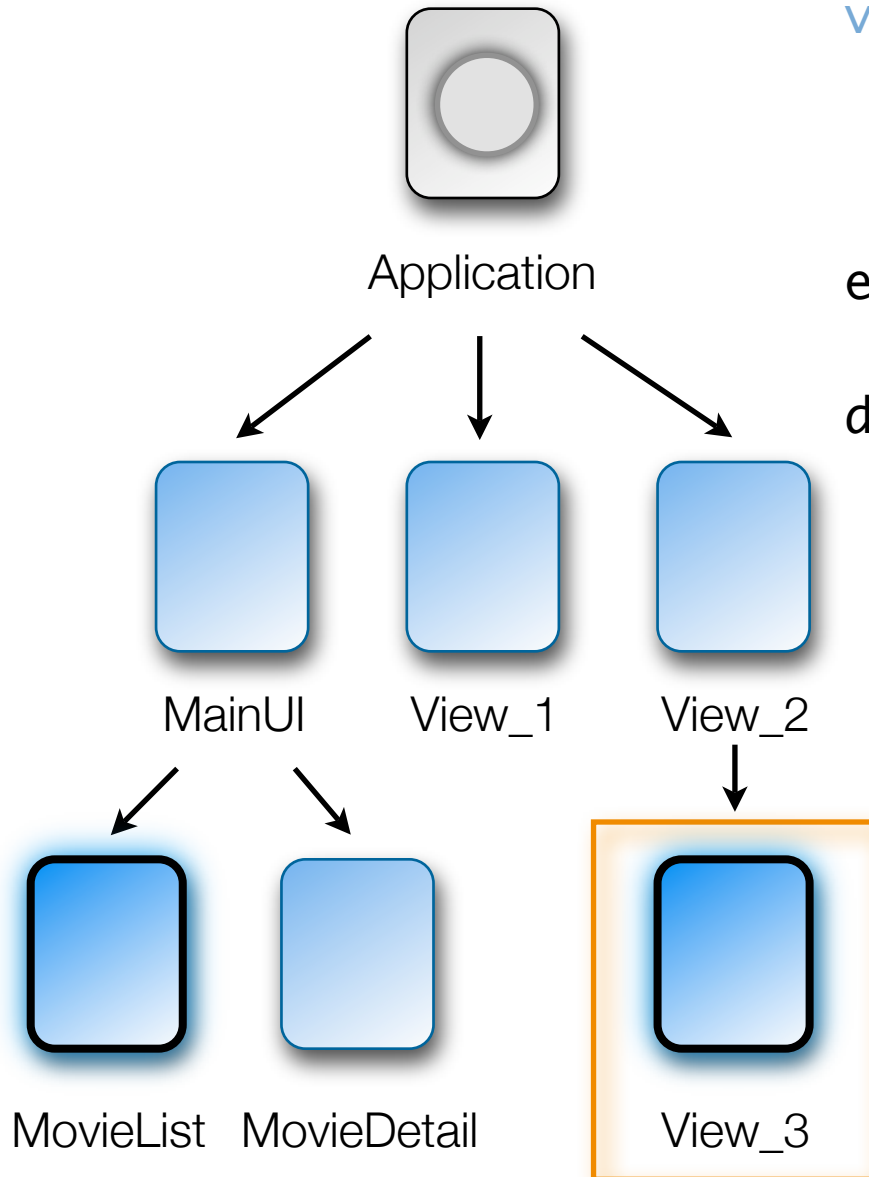
```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);
```

```
event.movie = list.selectedItem as Movie;
```

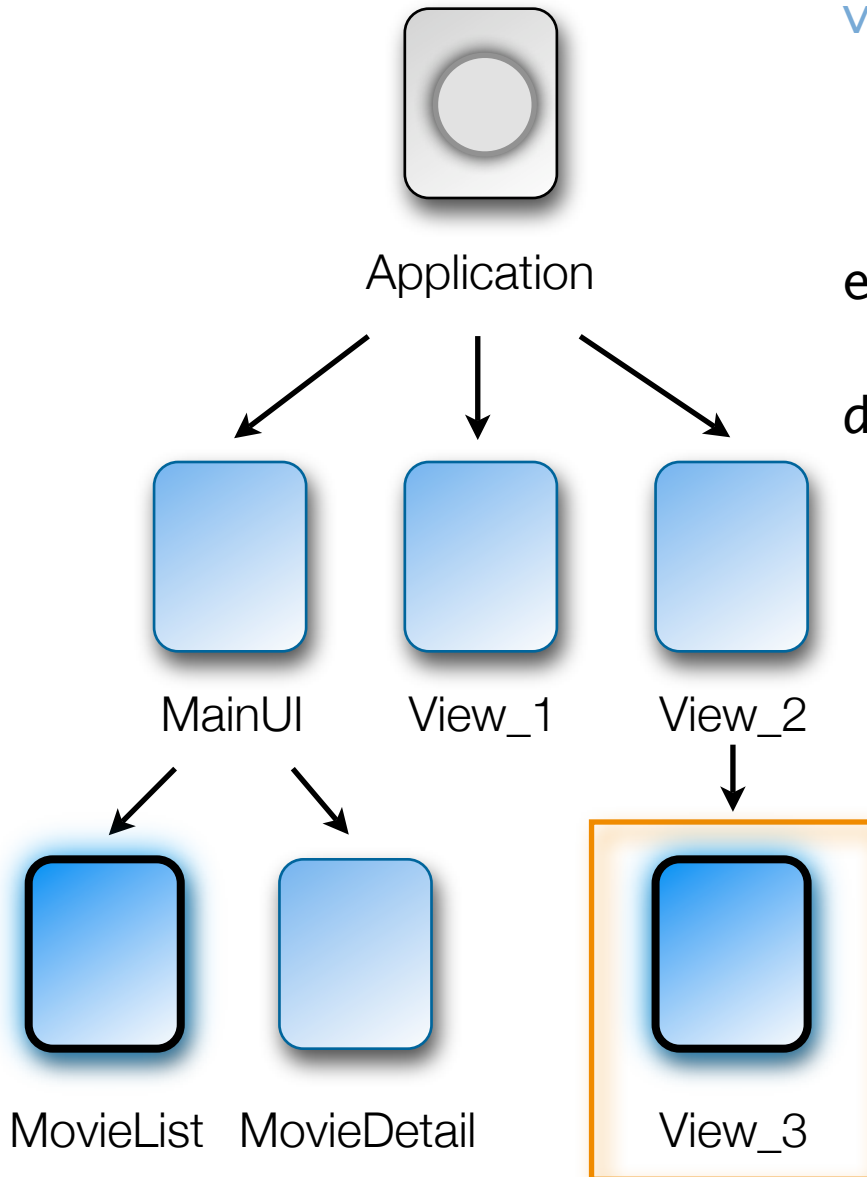
```
dispatchEvent(event);
```

Bubbling up event

```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);  
  
event.movie = list.selectedItem as Movie;  
  
dispatchEvent(event);
```



Bubbling up event



```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);
```

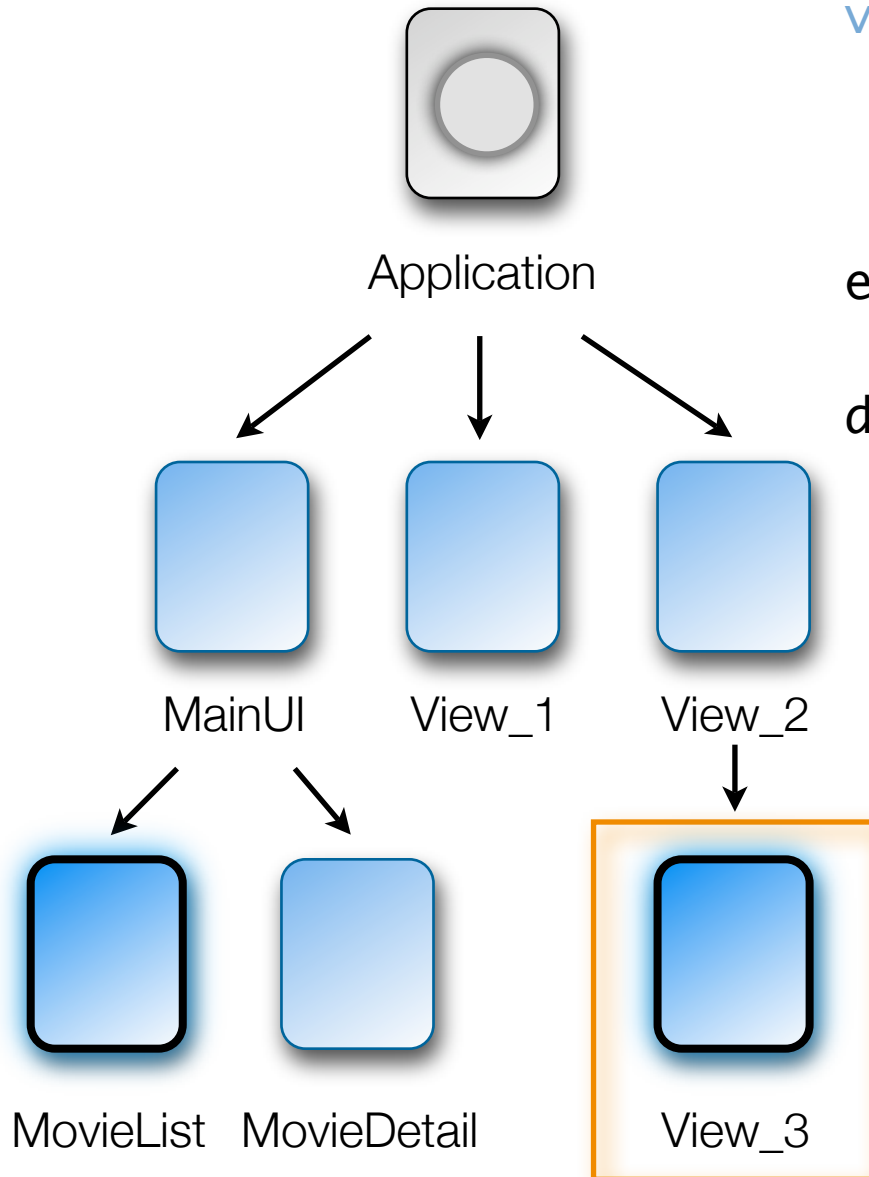
```
event.movie = list.selectedItem as Movie;
```

```
dispatchEvent(event);
```

```
public function movieSelected  
(event:MovieEvent):void  
{  
    //do something  
}
```

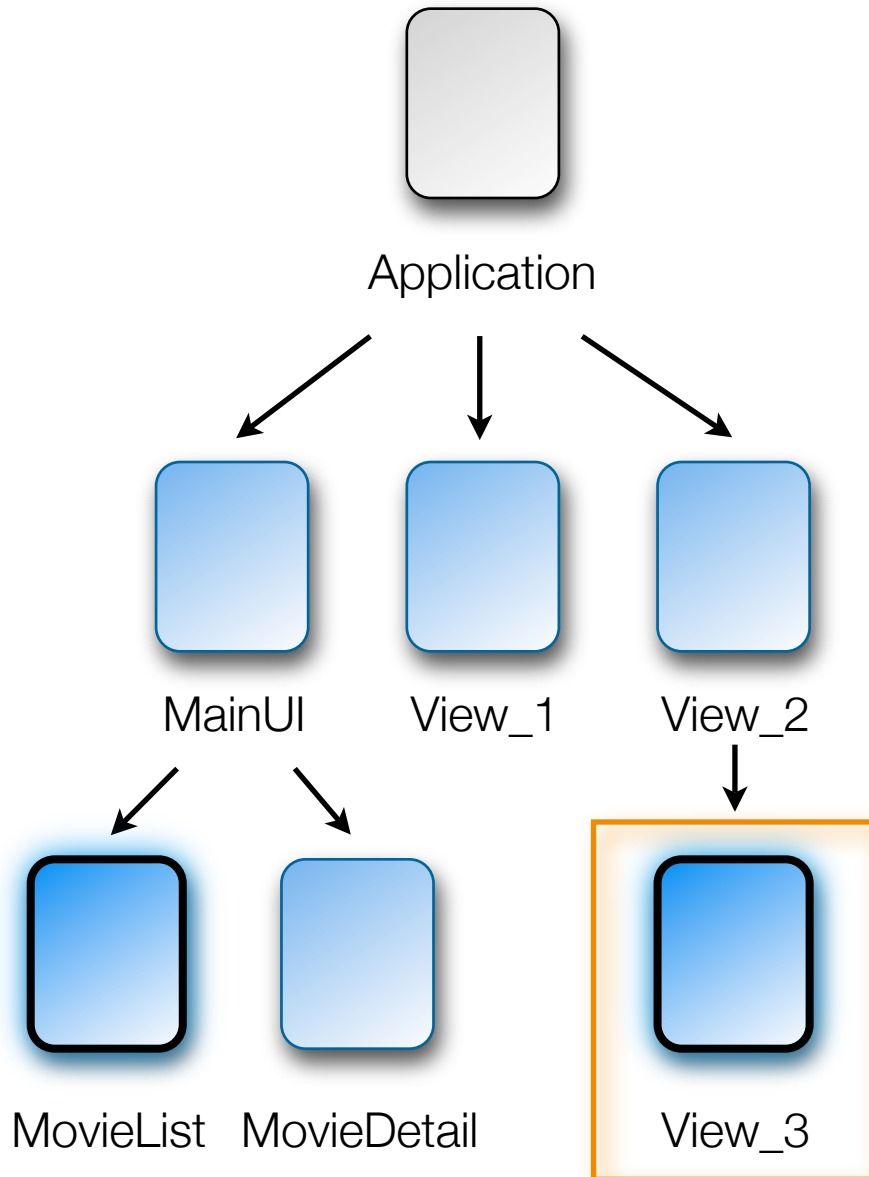
Bubbling up event

```
var event:MovieEvent =  
    new MovieEvent(MovieEvent.SELECTED,  
        true);  
  
event.movie = list.selectedItem as Movie;  
  
dispatchEvent(event);
```



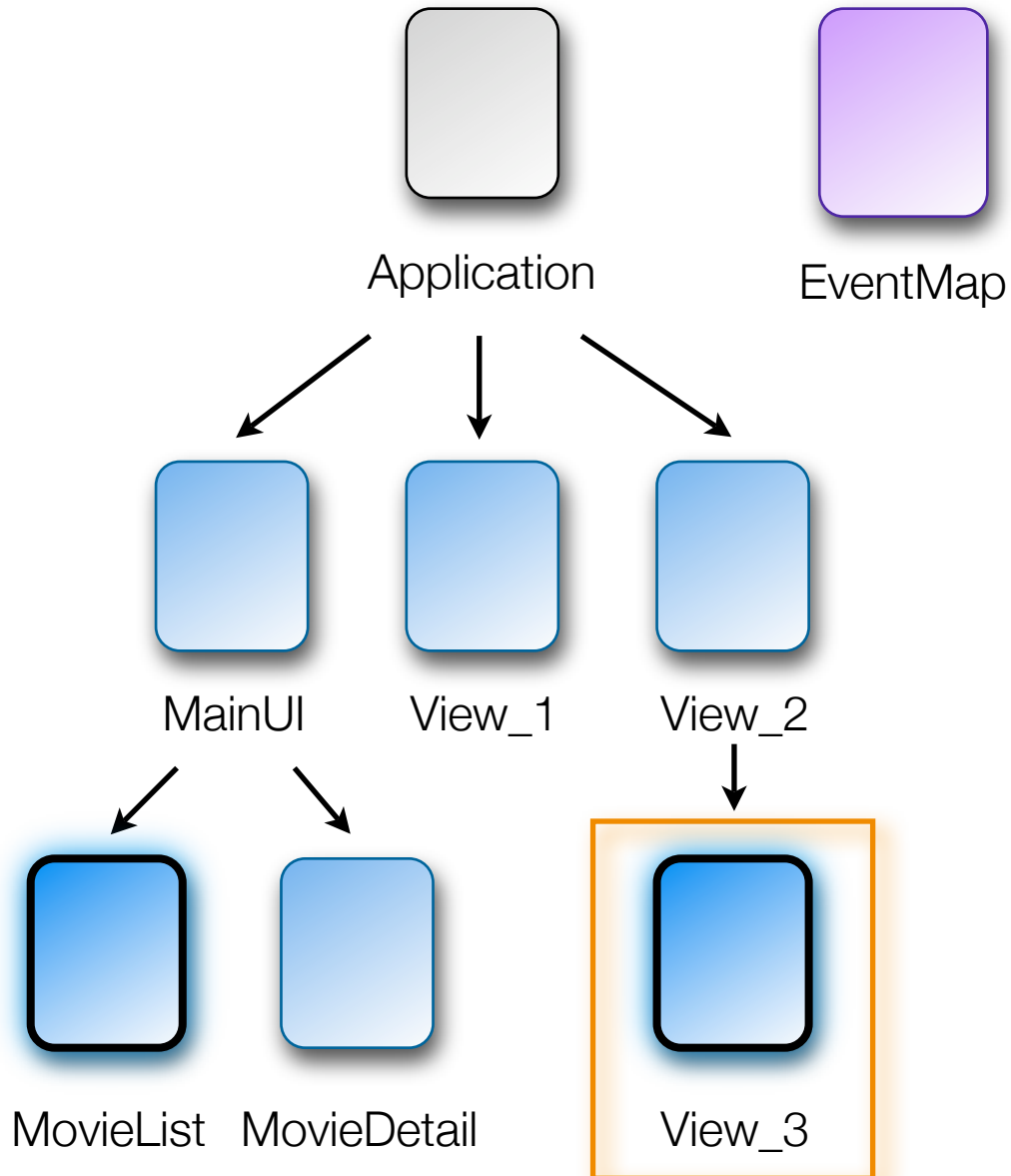
```
public function movieSelected  
(event:MovieEvent):void  
{  
    //do something  
}
```

Bubbling up event



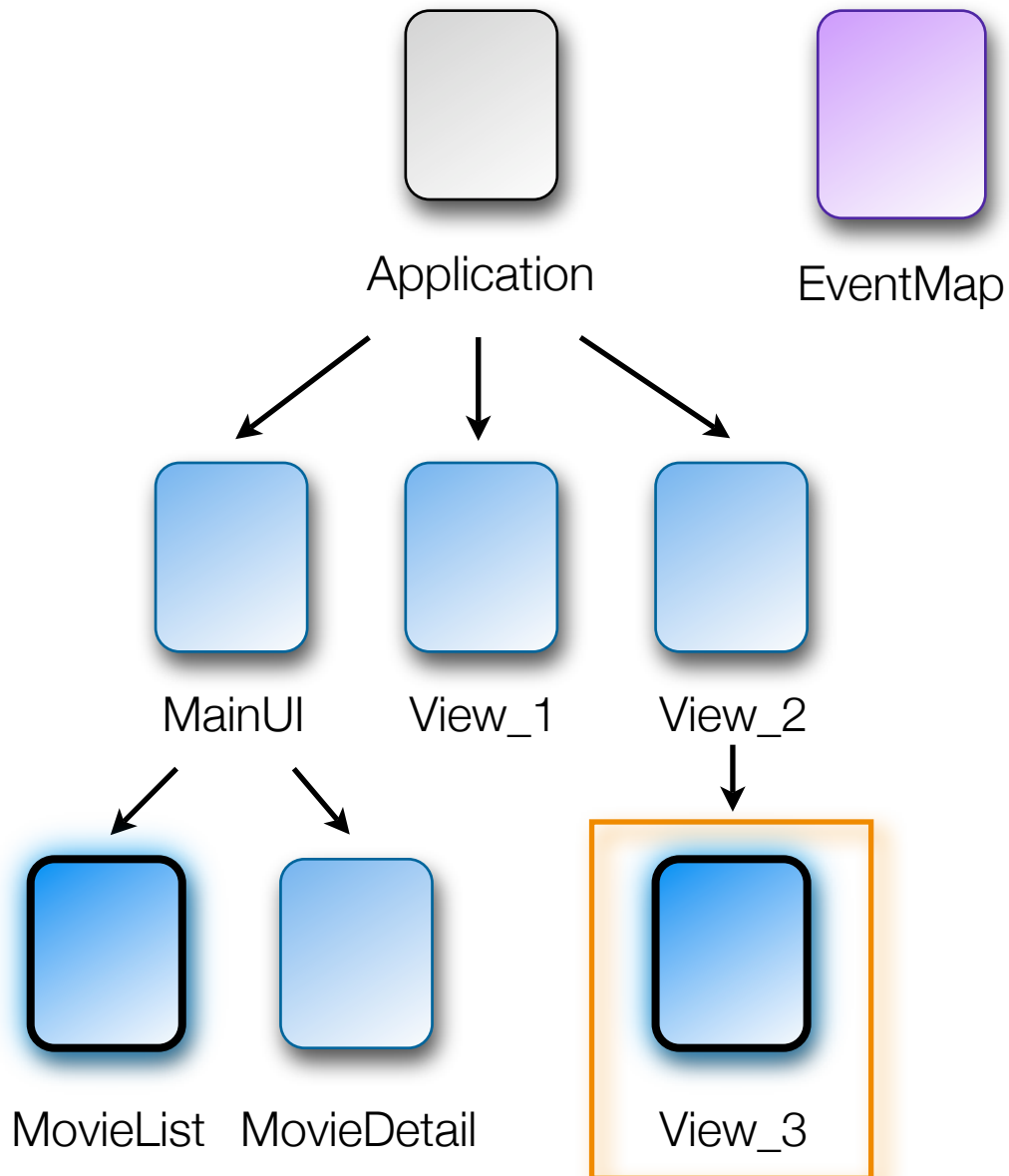
```
public function movieSelected  
(event:MovieEvent):void  
{  
    //do something  
}
```

Setting up the listener



```
public function movieSelected  
(event:MovieEvent):void  
{  
    //do something  
}
```

Setting up the listener



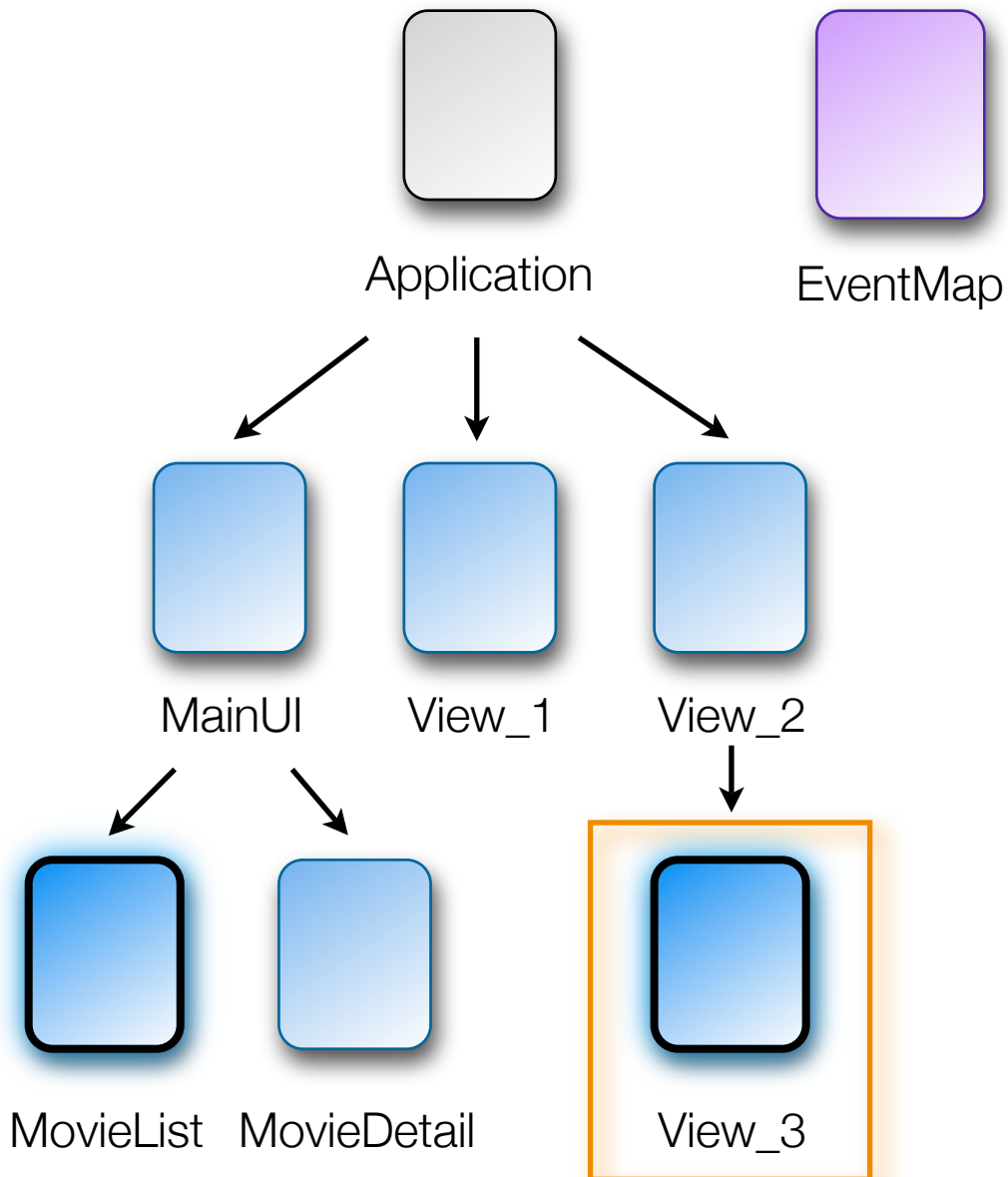
```
<Injectors target="{View_3}">
```

```
<ListenerInjector  
  eventType="{MovieEvent.SELECT}"  
  method="movieSelected" />
```

```
</Injectors>
```

```
public function movieSelected  
(event:MovieEvent):void  
{  
  //do something  
}
```

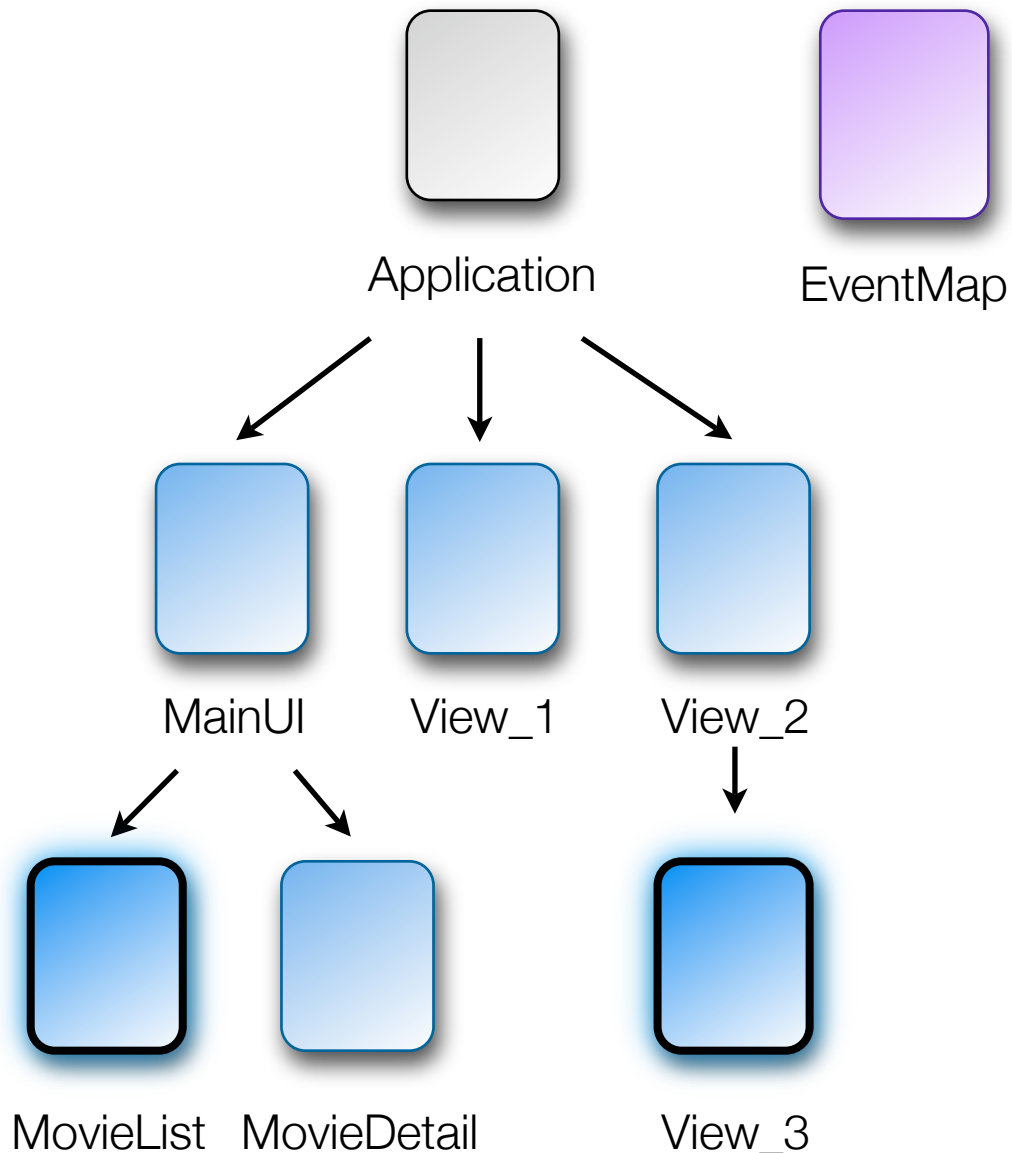
Setting up the listener



```
<Injectors target="{View_3}">  
<ListenerInjector  
  eventType="{MovieEvent.SELECT}"  
  method="movieSelected" />  
</Injectors>
```

```
public function movieSelected  
(event:MovieEvent):void  
{  
  //do something  
}
```


Setting up the listener



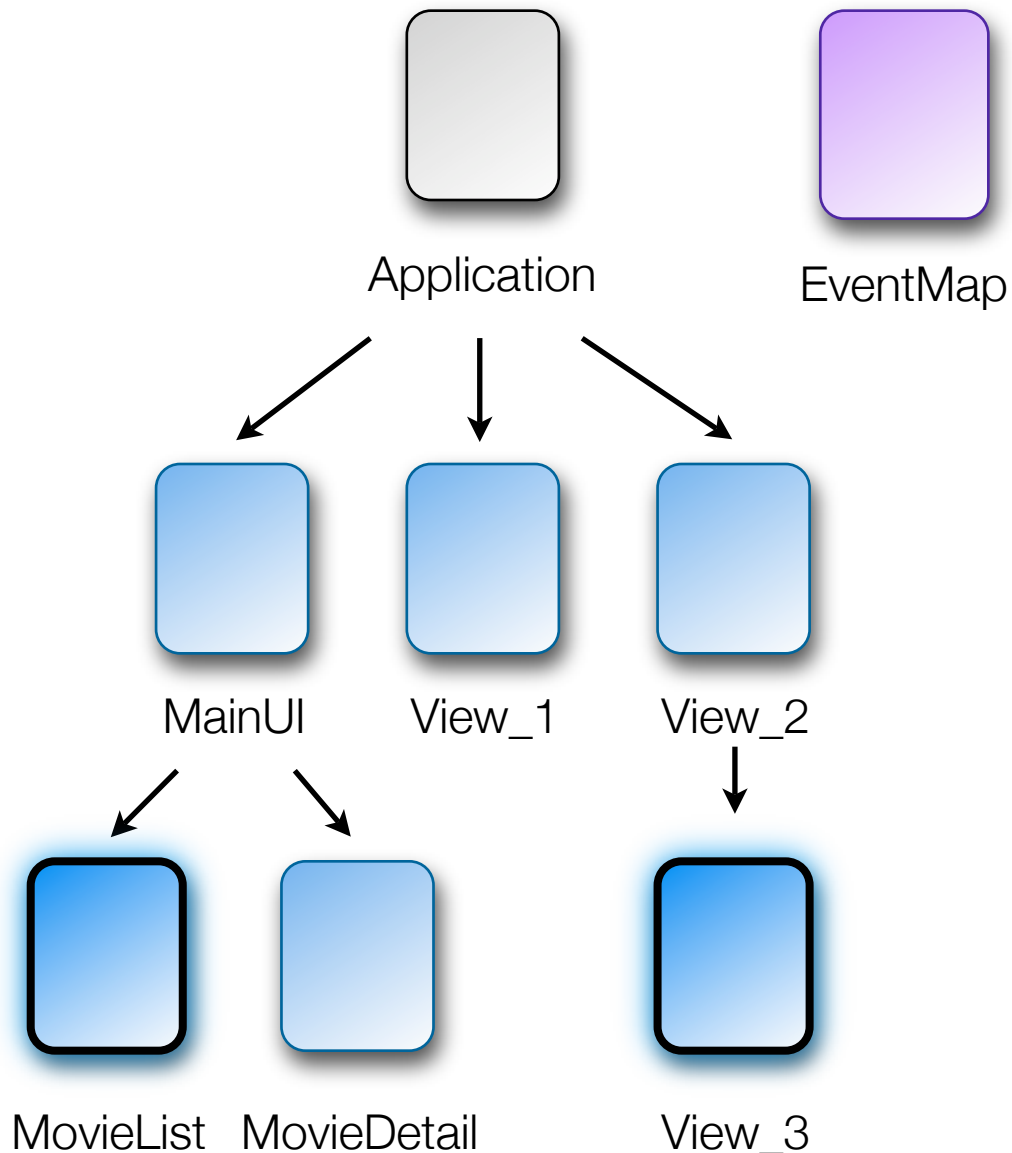
```
<Injectors target="{View_3}">
```

```
<ListenerInjector  
  eventType="{MovieEvent.SELECT}"  
  method="movieSelected" />
```

```
</Injectors>
```

```
public function movieSelected  
(event:MovieEvent):void  
{  
    //do something  
}
```

Setting up the listener



```
<Injectors target="{View_3}">  
  <ListenerInjector  
    eventType="{MovieEvent.SELECT}"  
    method="movieSelected" />  
</Injectors>
```

```
public function movieSelected  
(event:MovieEvent):void  
{  
  //do something  
}
```

Learn more at

<http://mate.asfusion.com>

