

# 更好的使用 Angular



# About Me

ALi (啊梨)

JS, PHP

小小工程師

 esbb48



# Angular 源由

- 2009 年開始計畫
- Google 維護
- 目前分支兩個版本 1.4 與 2.0
- 使用MVC的結構，結合雙向綁定的，讓開發和測試變得更加容易。

# Why Angular

# 你會怎麼做？

好買 · 好逛 · 好好玩  
**fri**day 購物

我的購物車

1選擇付款方式

2填寫訂購資料

3訂購成功

宅配 1 件

超商取貨 0 件



【狀元油飯】櫻花蝦粽 6粒 (180g/粒)

商品名稱	規格	售價	數量	優惠折扣	小計	編輯
		399	- 1 +		399	<div>♥ 加入我的收藏</div> <div>🗑 刪除</div>

商品售價：

399

優惠折扣：

- 0

實付金額總計：

399

繼續購物

下一步 開始結帳

# 用 Angular 怎麼做

Demo

眾人眼中的 Angular

# Two Way Binding

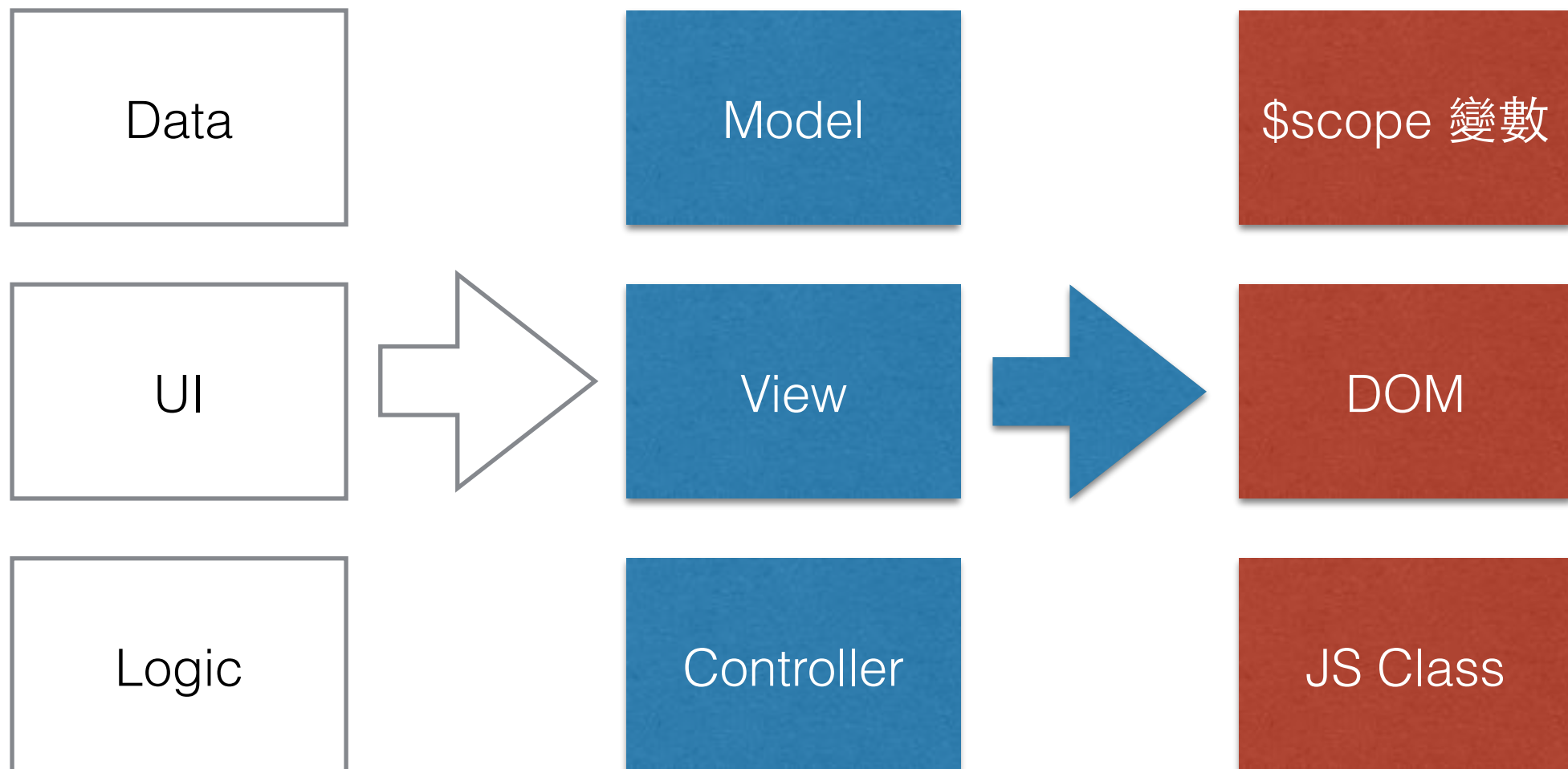


ng-model  
ng-controller  
ng-repeat...

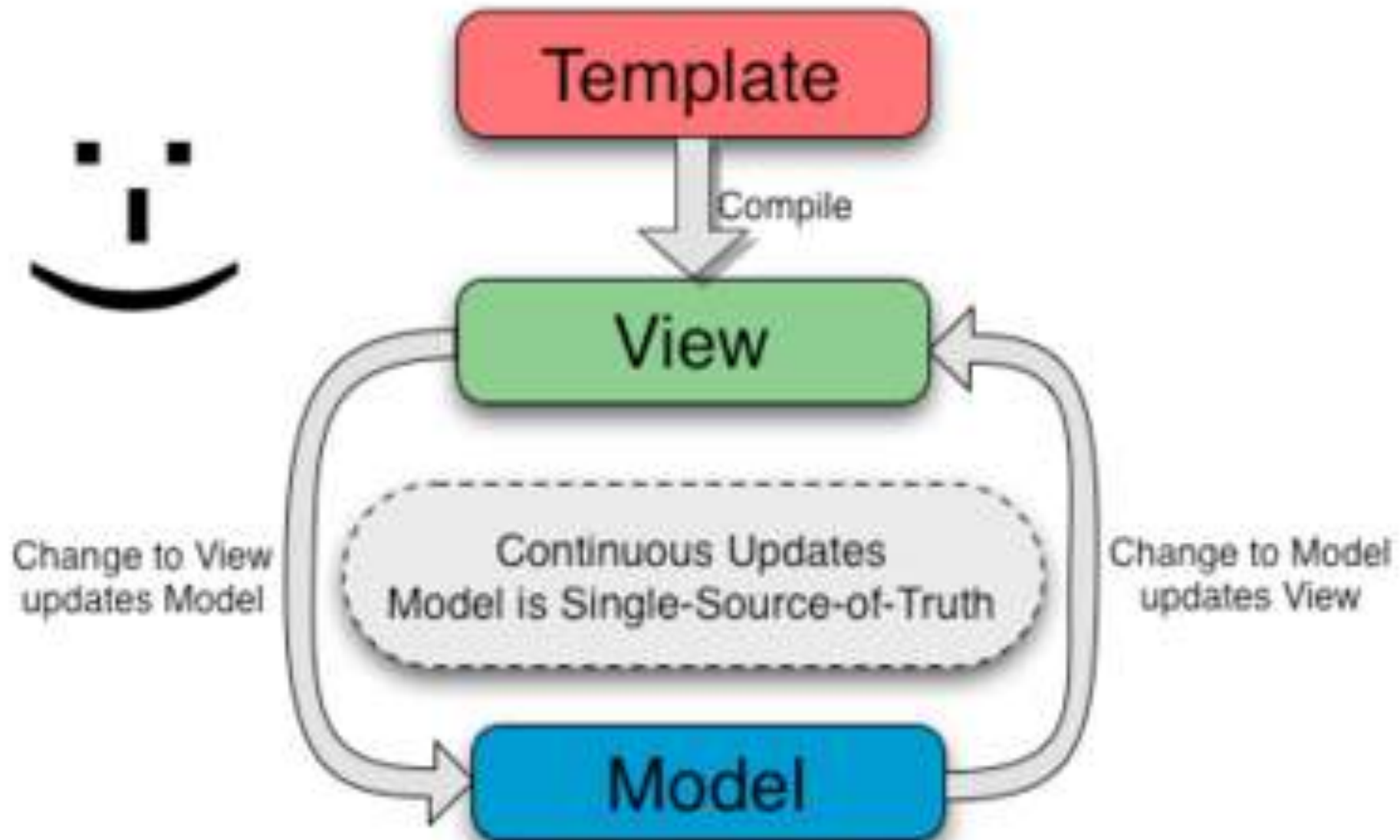
else?

Angular 特性

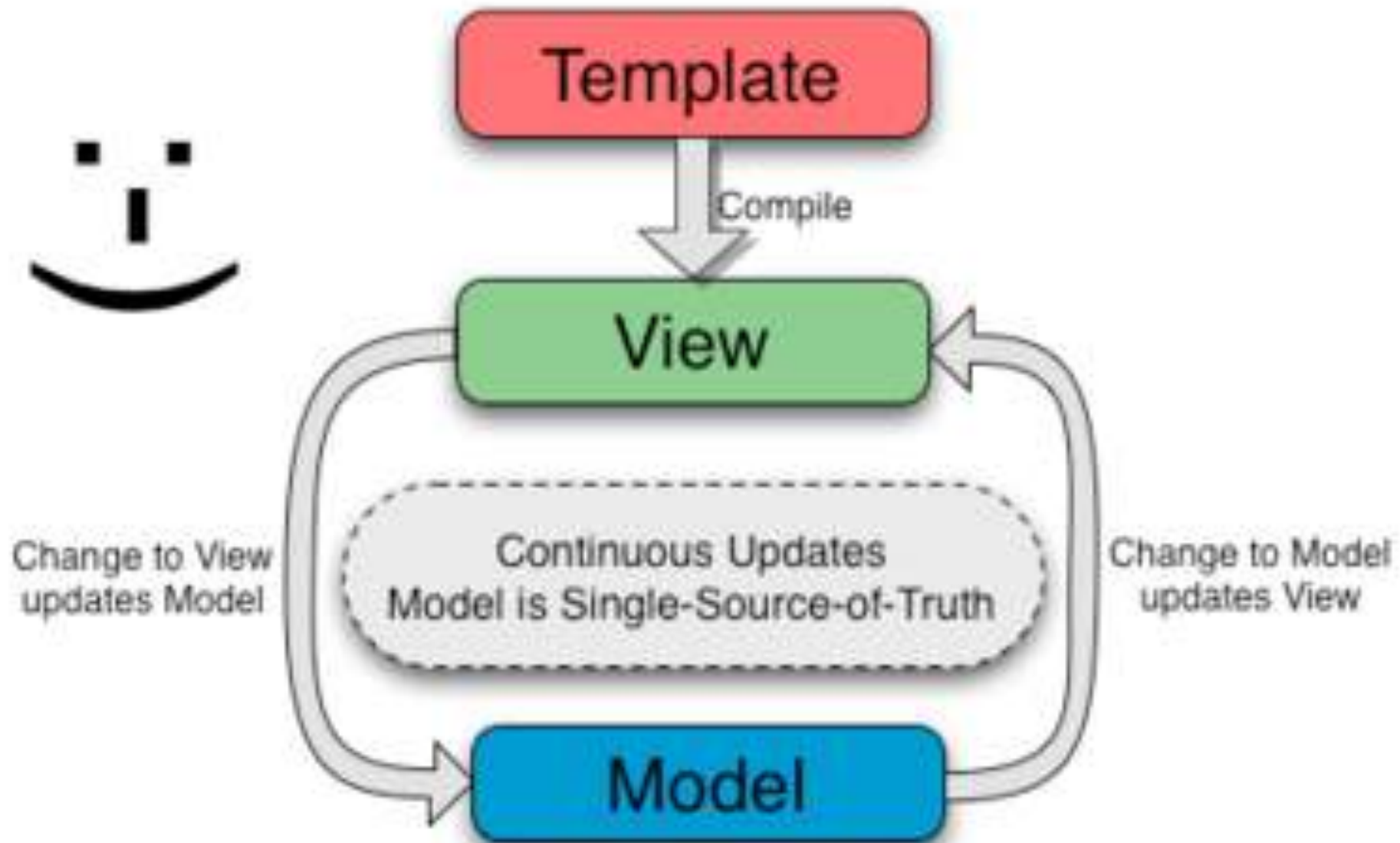
# MVC



# Two-Way Data Binding



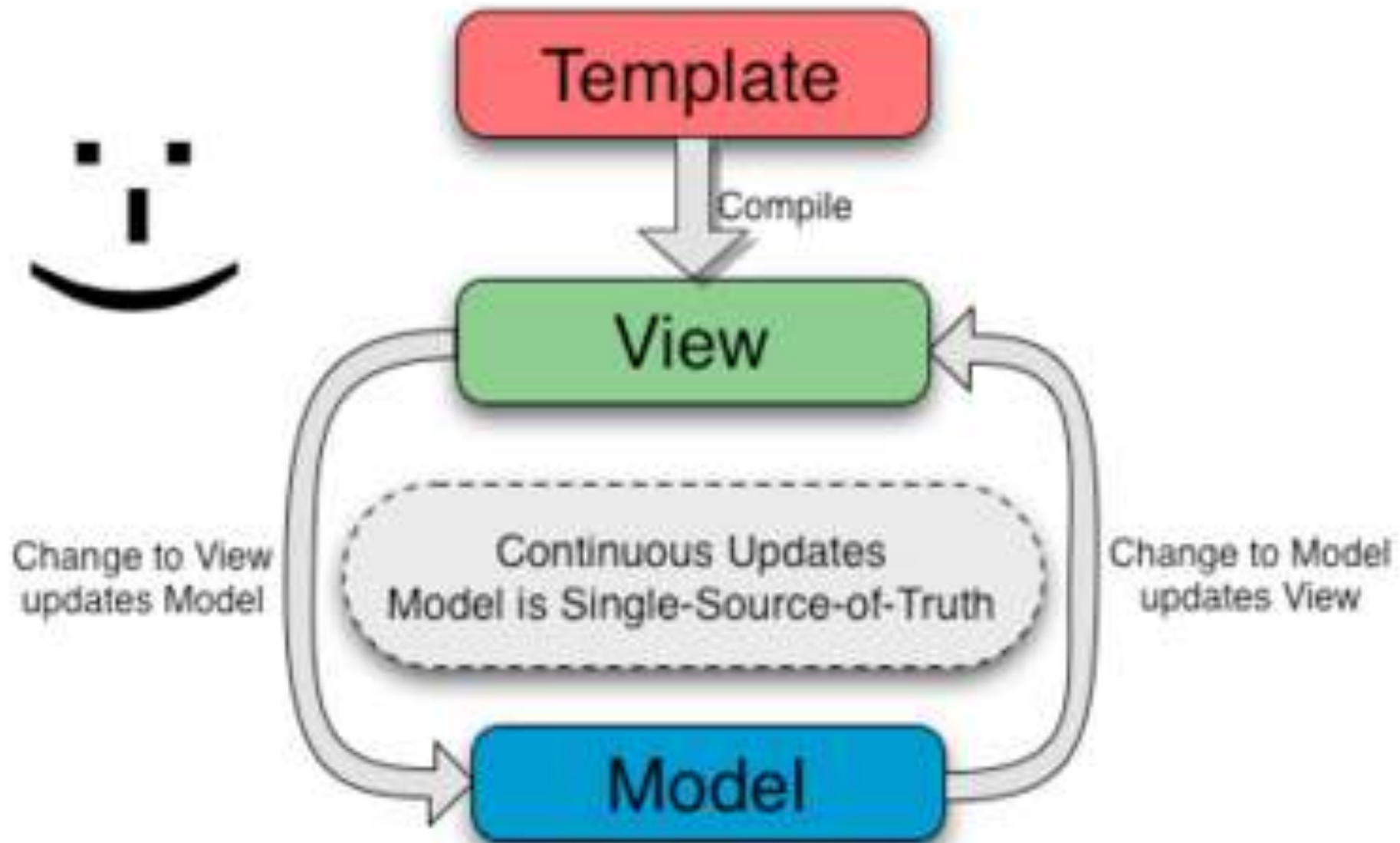
# Two-Way Data Binding



Dirty Check

<https://docs.angularjs.org/guide/databinding>

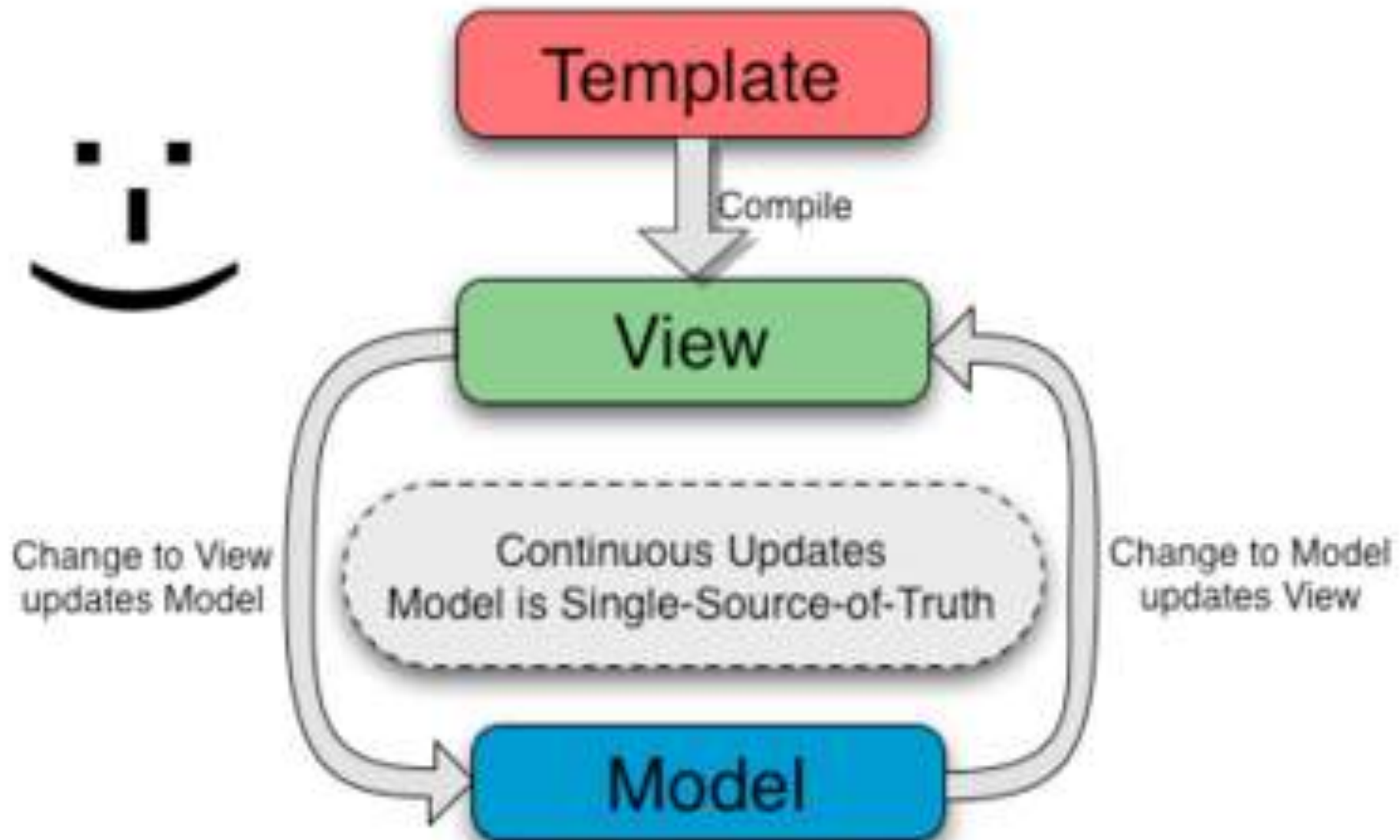
# Two-Way Data Binding



Dirty Check  
Object.observe()

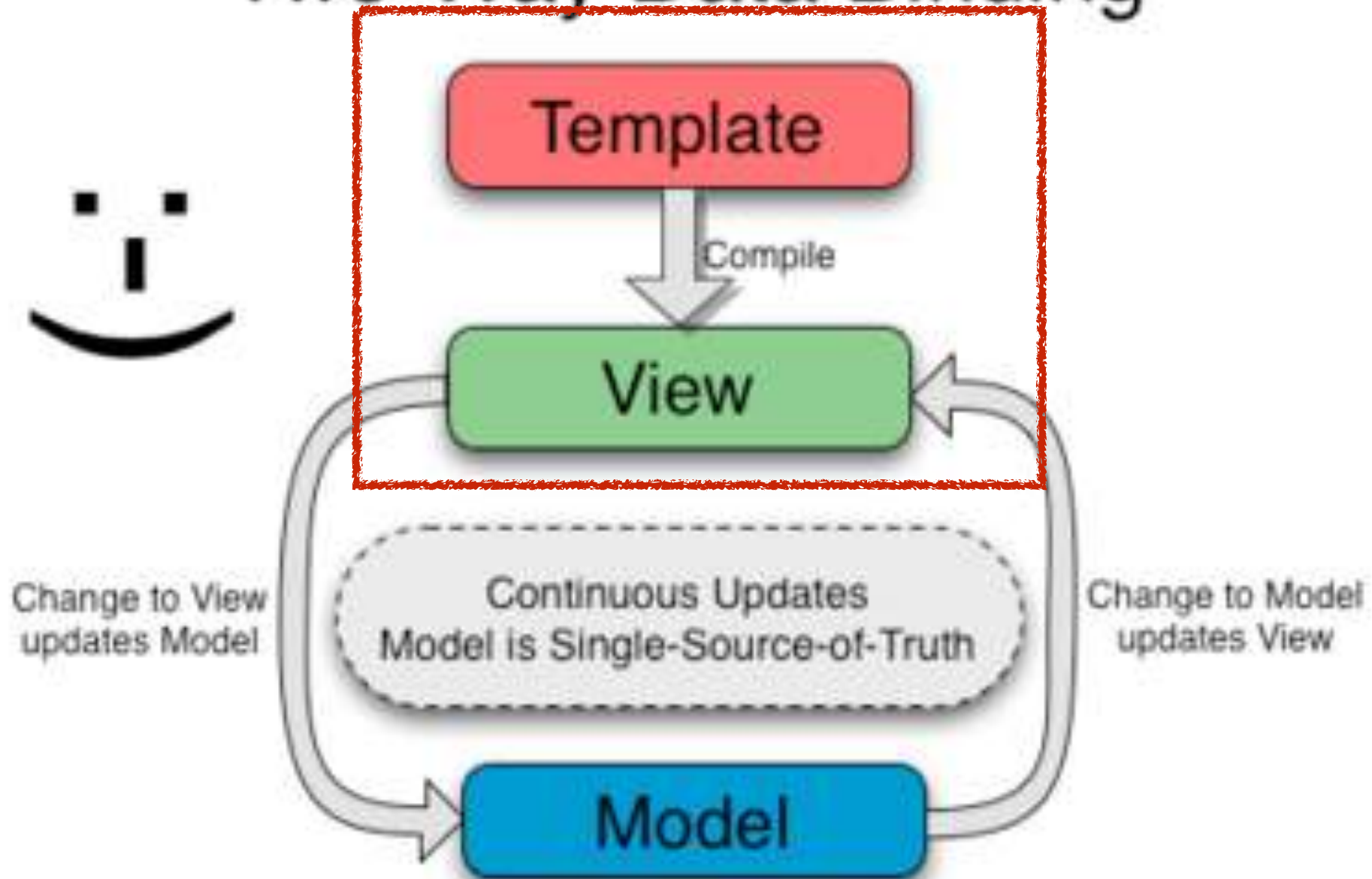
<https://docs.angularjs.org/guide/databinding>

# Two-Way Data Binding





# Two-Way Data Binding





Templates =  
HTML + **Directives**

Angular已提供

{{ xx }}  
ng-model  
ng-show  
ng-class  
ng-xx...

自定義

Custom Directives

# Custom Directives

```
angular.module('components', []).directive('demo', function() {  
  return {  
    restrict: 'A' / 'E' / 'C' / 'AEC'  
    scope: false / true / {},  
    link: function($scope, $element) {  
    },  
    template / templateUrl:  
  
  };  
})
```

# restrict

- 限制 directive 的適用範圍

Demo

# 基礎 scope

- 定義與外層 Scope的關係

Demo

# 進階 scope

- scope 為 {} 時，  
三種取得父 scope 的方式

Demo

# Dependency Injection

# 註冊 Service 的方式

```
module.service( 'serviceName', function );  
module.factory( 'factoryName', function );  
module.provider( 'providerName', function );
```



# Service

- Singleton（單例）
- 可以在跨 controller 不會被重置
- 可被調用的範圍不僅限於 controller
- Service 與 Factory 都是基於 Provider 的方式

# Service

```
angular.module('isService', function() {  
  this.returnThisFunction = true  
});
```

# Factory

```
angular.module('isFactory', function() {  
  this.hereIsPrivate = true  
  this.canProcessSthWhenRegisterService = true  
  
  return {  
    whenRegisterJustReturnHere: true  
  }  
});
```

# Provider

```
angular.module('isProvider', function() {  
  
    // 提供函式在 module.config 的時候被調用  
    self = this  
    self.messageString = '';  
    self.setMessageString = function (newString) {  
        self.messageString = newString;  
    }  
  
    // 這和 Factory 一樣  
    $get: function(){  
  
        self.hereIsPrivate = true  
        self.canProcessSthWhenRegisterService = true  
        return {  
            whenInjectionJustReturnHere: true  
            message: self.messageString  
        };  
    };  
});
```

Service 與 Factory  
是基於 Provider 的方式

可以被測試  
(Protractor)

# 常見的Angular範例

好用的ng-model



```
<label>Leave a Message</label>  
<textarea row="3" ng-model="message"> </textarea>  
</hr>  
<h1>You Say : {{message}}</h1>
```

好用的ng-bind

```
<label>Type Something ...</label>
<textarea row="3" ng-model="message"> </textarea>
</hr>
<h1>You Say : <span ng-bind="message"></span></h1>
```

Demo

好用的ng-controller

```
<div ng-controller="baseController">
  <label>Type Something ...</label>
  <textarea row="3" ng-model="message"> </textarea>
  </hr>
  <h1>You Say : {{message}}</h1>
</div>
```

```
angular.module('baseApp', [])
.controller('baseController', function($scope) {
  $scope.message = 'Default Message ...'
});
```

```
<div ng-controller="baseController">  
  <label>Type Something ...</label>  
  <textarea row="3" ng-model="message"> </textarea>  
</div>
```

← → ↻ 🏠 output.jsbin.com/xadihabowi/6

Type Something ...

You Say : Default Message ...

好用的ng-repeat



```
angular.module('baseApp', [])  
.controller('baseController', function($scope) {  
  $scope.messages = [  
    {content:"Hello!", from: "ALi"},  
    {content:"哈囉 !", from: "Ann" },  
    {content:"這是測試", from: "Bee"}  
  ];  
});
```

```
<div ng-controller="baseController">  
  <h1 ng-repeat="m in messages">  
    You Say : {{m.content}}  
  </h1>  
</div>
```



```
angular.module('baseApp', [])  
.controller('baseController', function($scope) {  
  $scope.messages = [  
    'You Say : Hello!',  
    'You Say : 哈囉!',  
    'You Say : 這是測試'  ];  
});
```

← → ↻ 🏠 output.jsbin.com/xadihabowi/8

You Say : Hello!

You Say : 哈囉!

You Say : 這是測試

好好用，一直用

好好用，一直用

伴隨著功能  
一直長一直長 ...





對不起  
這個才對



```

<!doctype html>
<html ng-app="baseApp">
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.15/angular.min.js"></script>
  </head>
  <body ng-controller="baseController">

    <H1> {{ article.title}} </H1>
    <div class="article-content"> {{ article.content }} </div>
    <hr />

    <H3> 留言 </H3>
    <div id="msg-form" >
      <div class="msg-box">
        <div class="msg-content">{{formData.content}}</div>
        <span>From : {{formData.from}}</span>
      </div>
      <div>
        <label>Name</label>
        <input type="text" ng-model="formData.from"/>
      </div>
      <textarea ng-model="formData.content" row="4"> </textarea>
      <button ng-click="sendMsg()"> Send Msg</button>
    </div>
    <hr/>

    <H3> 回應 </H3>
    <div id="msg-list" >
      <div class="msg-box" ng-repeat="m in messages">
        <div class="msg-content">{{m.content}}</div>
        <span>From : {{m.from}}</span>
      </div>
    </div>

  </body>
</html>

```



```

angular.module('baseApp', [])
.controller('baseController', function($scope) {

    $scope.messages = [
        {content:"Hello!", from: "ALi"},
        {content:"哈囉 !!!!!!!!!!!!!", from: "Ann" },
        {content:"這是測試", from: "Bee"}
    ];

    $scope.formData = {
        content: "Karma !!!!!",
        from: "MM ^__^"
    };

    $scope.sendMsg = function(){
        $scope.messages.push($scope.formData);
        $scope.formData = {content: "", form: ""};
    };

    $scope.article = {
        title: "Karma - Spectacular Test Runner for Javascript",
        content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh
    };
});

```

維護性低

分析

```

<!doctype html>
<html ng-app="baseApp">
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.15/angular.min.js"></script>
  </head>
  <body ng-controller="baseController">

    <H1> {{ article.title }} </H1>
    <div class="article-content"> {{ article.content }} </div>
    <hr />

    <H3> 留言 </H3>
    <div id="msg-form" >
      <div class="msg-box">
        <div class="msg-content">{{formData.content}}</div>
        <span>From : {{formData.from}}</span>
      </div>
      <div>
        <label>Name</label>
        <input type="text" ng-model="formData.from"/>
      </div>
      <textarea ng-model="formData.content" row="4"> </textarea>
      <button ng-click="sendMsg()"> Send Msg</button>
    </div>
    <hr/>

    <H3> 回應 </H3>
    <div id="msg-list" >
      <div class="msg-box" ng-repeat="m in messages">
        <div class="msg-content">{{m.content}}</div>
        <span>From : {{m.from}}</span>
      </div>
    </div>

  </body>
</html>

```

```

<!doctype html>
<html ng-app="baseApp">
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.15/angular.min.js"></script>
  </head>
  <body ng-controller="baseController">

    <H1> {{ article.title }} </H1>
    <div class="article-content"> {{ article.content }} </div>
    <hr />

    <H3> 留言 </H3>
    <div id="msg-form" >
      <div class="msg-box">
        <div class="msg-content">{{formData.content}}</div>
        <span>From : {{formData.from}}</span>
      </div>
      <div>
        <label>Name</label>
        <input type="text" ng-model="formData.from"/>
      </div>
      <textarea ng-model="formData.content" row="4"> </textarea>
      <button ng-click="sendMsg()"> Send Msg</button>
    </div>
    <hr/>

    <H3> 回應 </H3>
    <div id="msg-list" >
      <div class="msg-box" ng-repeat="m in messages">
        <div class="msg-content">{{m.content}}</div>
        <span>From : {{m.from}}</span>
      </div>
    </div>

  </body>
</html>

```



```

<!doctype html>
<html ng-app="baseApp">
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.15/angular.min.js"></script>
  </head>
  <body ng-controller="baseController">

    <H1> {{ article.title }} </H1>
    <div class="article-content"> {{ article.content }} </div>
    <hr />

    <H3> 留言 </H3>
    <div id="msg-form" >
      <div class="msg-box">
        <div class="msg-content">{{formData.content}}</div>
        <span>From : {{formData.from}}</span>
      </div>
      <div>
        <label>Name</label>
        <input type="text" ng-model="formData.from"/>
      </div>
      <textarea ng-model="formData.content" row="4"> </textarea>
      <button ng-click="sendMsg()"> Send Msg</button>
    </div>
    <hr/>

    <H3> 回應 </H3>
    <div id="msg-list" >
      <div class="msg-box" ng-repeat="m in messages">
        <div class="msg-content">{{m.content}}</div>
        <span>From : {{m.from}}</span>
      </div>
    </div>

  </body>
</html>

```

拉出重複的程式碼

```

angular.module('baseApp', [])
.controller('baseController', function($scope) {

    $scope.messages = [
        {content:"Hello!", from: "ALi"},
        {content:"哈囉 !!!!!!!!!!!!!", from: "Ann" },
        {content:"這是測試", from: "Bee"}
    ];

    $scope.formData = {
        content: "Karma !!!!!",
        from: "MM ^__^"
    };

    $scope.sendMsg = function(){
        $scope.messages.push($scope.formData);
        $scope.formData = {content: "", form: ""};
    };

    $scope.article = {
        title: "Karma - Spectacular Test Runner for Javascript",
        content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh
    };
});

```



```

angular.module('baseApp', [])
.controller('baseController', function($scope) {

    $scope.messages = [
        {content:"Hello!", from: "ALi"},
        {content:"哈囉 !!!!!!!!!!!!!", from: "Ann" },
        {content:"這是測試", from: "Bee"}
    ];

    $scope.formData = {
        content: "Karma !!!!!",
        from: "MM ^__^"
    };

    $scope.sendMsg = function(){
        $scope.messages.push($scope.formData);
        $scope.formData = {content: "", form: ""};
    };

    $scope.article = {
        title: "Karma - Spectacular Test Runner for Javascript",
        content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh
    };
});

```

```
angular.module('baseApp', [])  
.controller('baseController', function($scope) {
```

訊息

```
$scope.messages = [  
  {content:"Hello!", from: "ALi"},  
  {content:"哈囉 !!!!!!!!!!!!!", from: "Ann" },  
  {content:"這是測試", from: "Bee"}  
];
```

```
$scope.formData = {  
  content: "Karma !!!!!",  
  from: "MM ^__^"  
};
```

```
$scope.sendMsg = function(){  
  $scope.messages.push($scope.formData);  
  $scope.formData = {content: "", form: ""};  
};
```

```
$scope.article = {  
  title: "Karma - Spectacular Test Runner for Javascript",  
  content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh  
};  
});
```

```
angular.module('baseApp', [])  
.controller('baseController', function($scope) {
```

訊息

```
$scope.messages = [  
  {content:"Hello!", from: "ALi"},  
  {content:"哈囉 !!!!!!!!!!!!!", from: "Ann" },  
  {content:"這是測試", from: "Bee"}  
];  
  
$scope.formData = {  
  content: "Karma !!!!!",  
  from: "MM ^__^"  
};  
  
$scope.sendMsg = function(){  
  $scope.messages.push($scope.formData);  
  $scope.formData = {content: "", form: ""};  
};
```

文章

```
$scope.article = {  
  title: "Karma - Spectacular Test Runner for Javascript",  
  content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh  
};  
});
```

```
angular.module('baseApp', [])  
.controller('baseController', function($scope) {
```

訊息

```
$scope.messages = [  
  {content:"Hello!", from: "ALi"},  
  {content:"哈囉 !!!!!!!!!!!!!", from: "Ann" },  
  {content:"這是測試", from: "Bee"}  
];  
  
$scope.formData = {  
  content: "Karma !!!!!",  
  from: "MM ^__^"  
};  
  
$scope.sendMsg = function(){  
  $scope.messages.push($scope.formData);  
  $scope.formData = {content: "", form: ""};  
};
```

文章

```
$scope.article = {  
  title: "Karma - Spectacular Test Runner for Javascript",  
  content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh  
};  
});
```

依關聯性將程式碼分類

模組化

# 理想上的模組化

- 好用
- 方便維護
- 單一職責原則
- 保持程式碼簡潔

先從 View 開始



# 歸納出相同程式碼

預覽部分

```
<div class="msg-box">  
  <div class="msg-content">{{formData.content}}</div>  
  <span>From : {{formData.from}}</span>  
</div>
```

留言列

```
<div class="msg-box" ng-repeat="m in messages">  
  <div class="msg-content">{{m.content}}</div>  
  <span>From : {{m.from}}</span>  
</div>
```

# 包成 component

```
app = angular.module('baseApp', [])
app.directive('msgBox', function() {
  return {
    restrict: 'E',
    templateUrl: 'msg-box.html',
    scope: {
      data: "="
    }
  }
});
```

app.js

```
<div class="msg-box">
  <div class="msg-content">{{data.content}}</div>
  <span>From : {{data.from}}</span>
</div>
```

msg-box.html



# 使用方式

```
<body ng-controller="baseController">

  <H1> {{ article.title}} </H1>
  <div class="article-content"> {{ article.content }} </div>
  <hr />

  <H3> 留言 </H3>
  <div id="msg-form" >
    <msg-box data="formData"></msg-box>
    <div>
      <label>Name</label>
      <input type="text" ng-model="formData.from"/>
    </div>
    <textarea ng-model="formData.content" row="4"> </textarea>
    <button ng-click="sendMsg()"> Send Msg</button>
  </div>
  <hr/>

  <H3> 回應 </H3>
  <div id="msg-list" >
    <msg-box data="m" ng-repeat="m in messages"></msg-box>
  </div>

</body>
```

# Before Vs After

Before

```
<div class="msg-box">  
  <div class="msg-content">{{formData.content}}</div>  
  <span>From : {{formData.from}}</span>  
</div>
```

After

```
<msg-box data="formData"></msg-box>
```

Controller

# 依據關連性分類

```
angular.module('baseApp', [])  
.controller('baseController', function($scope) {
```

```
  $scope.messages = [  
    {content: "Hello!", from: "ALi"},  
    {content: "哈囉 !!!!!!!", from: "Ann" },  
    {content: "這是測試", from: "Bee"}  
  ];
```

訊息

```
  $scope.formData = {  
    content: "Karma !!!!!",  
    from: "MM ^__^"  
  };
```

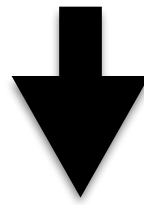
```
  $scope.sendMsg = function(){  
    $scope.messages.push($scope.formData);  
    $scope.formData = {content: "", form: ""};  
  };
```

```
  $scope.article = {  
    title: "Karma - Spectacular Test Runner for Javascript",  
    content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's w  
  };  
});
```

文章

# 文章

```
$scope.article = {  
  title: "Karma - Spectacular Test Runner for Javascript",  
  content: "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. That's wh  
};
```



```
app.service('article',function(){  
  this.title = "Karma - Spectacular Test Runner for Javascript";  
  this.content = "On the AngularJS team, we rely on testing and we always seek better tools to make our life easier. \  
  That's why we created Karma - a test runner that fits all our needs."  
});
```



# 調用

```
app = angular.module('baseApp', [])  
  
app.controller('baseController', function($scope, article) {  
    $scope.article = article  
})
```



# 訊息

```
$scope.messages = [
  {content:"Hello!", from: "ALi"},
  {content:"哈囉 !!!!!!!!!!!!!!!", from: "Ann" },
  {content:"這是測試", from: "Bee"}
];

$scope.formData = {
  content: "Karma !!!!!",
  from: "MM ^__^"
};

$scope.sendMsg = function(){
  $scope.messages.push($scope.formData);
  $scope.formData = {content: "", form: ""};
};
```

# 訊息

```
$scope.messages = [  
  {content:"Hello!", from: "ALi"},  
  {content:"哈囉 !!!!!!!!!!!!!!!", from: "Ann" },  
  {content:"這是測試", from: "Bee"}  
];
```

```
$scope.formData = {  
  content: "Karma !!!!!",  
  from: "MM ^__^"  
};
```

```
$scope.sendMsg = function(){  
  $scope.messages.push($scope.formData);  
  $scope.formData = {content: "", form: ""};  
  
};
```

# 訊息

```
$scope.messages = [  
  {content:"Hello!", from: "ALi"},  
  {content:"哈囉 !!!!!!!!!!!!!!!", from: "Ann" },  
  {content:"這是測試", from: "Bee"}  
];
```

messages

```
$scope.formData = {  
  content: "Karma !!!!!",  
  from: "MM ^__^"  
};
```

```
$scope.sendMsg = function(){  
  $scope.messages.push($scope.formData);  
  $scope.formData = {content: "", form: ""};  
};
```



# 訊息

```
$scope.messages = [  
  {content:"Hello!", from: "ALi"},  
  {content:"哈囉 !!!!!!!!!!!!!!!", from: "Ann" },  
  {content:"這是測試", from: "Bee"}  
];
```

messages

```
$scope.formData = {  
  content: "Karma !!!!!",  
  from: "MM ^__^"  
};
```

messageServ

```
$scope.sendMsg = function(){  
  $scope.messages.push($scope.formData);  
  $scope.formData = {content: "", form: ""};  
};
```

# Factory

```
app.factory('messageServ', function(messages){
  self = this
  self.defaultFormData = {
    content: 'Hello World !',
    from: 'Mm'
  };

  self.cleanFormData = function(formData) {
    formData.content = '';
    formData.from = '';
    return formData
  }

  self.clone = function(formData) {
    return JSON.parse(JSON.stringify(formData));
  }

  return {
    formData: self.defaultFormData,
    sendMsg: function(formData){
      cloneFormData = self.clone(formData);
      messages.push(cloneFormData);
      formData = self.cleanFormData(formData)
    }
  };
});
```

# 調用

```
app.controller('baseController', function($scope, article, messages, messageServ) {  
    $scope.messages = messages  
    $scope.sendMsg = messageServ.sendMsg  
    $scope.formData = messageServ.formData  
})
```

# Before Vs After

Before

```
angular.module('baseApp', [])
.controller('baseController', function($scope) {
  $scope.article = {
    title: "Karma - Spectacular Test Runner for Javascript",
    content: "On the AngularJS team, we rely on testing and we always \
seek better tools to make our life easier. That's why we created \
Karma - a test runner that fits all our needs."
  };

  $scope.formData = {
    content: "Karma !!!!!",
    from: "MM ^__^"
  };

  $scope.messages = [
    {content:"Hello!", from: "ALi"},
    {content:"哈囉 !!!!!!!!!!!!!!!", from: "Ann" },
    {content:"這是測試", from: "Bee"}
  ];

  $scope.sendMsg = function(){
    $scope.messages.push($scope.formData);
    $scope.formData = {content: "", form: ""};
  };
})
```

After

```
angular.module('baseApp', [])
.controller('baseController',
  function($scope, article, messages, messageServ) {
    $scope.article = article;
    $scope.formData = messageServ.formData;
    $scope.messages = messages;
    $scope.sendMsg = messageServ.sendMsg;
  }
);
```

# 回顧

- Why Angular
- Angular特性
- 常見範例
- 使用特性模組化



大家都說  
Angular不好學



My Feelings About AngularJS Over Time



善用框架概念  
而不被框架所局限

善用 Angular 特性  
寫出好維護的程式碼

謝謝大家 =]