

# DATA BINDING



*Client-side View of Data*

*Client*

MY BLOG

This is my first post.

ADD POST

MY BLOG

02/23/15

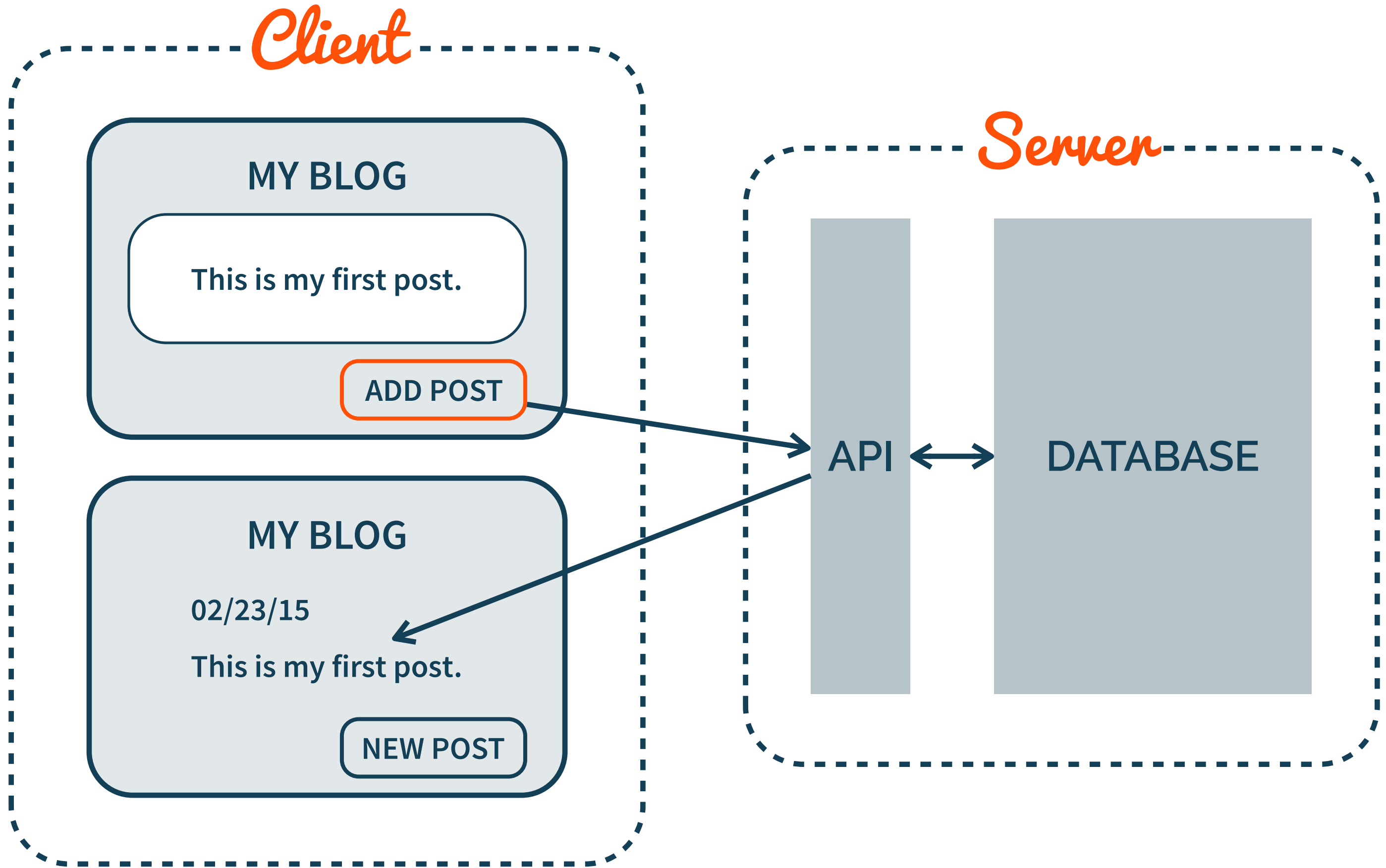
This is my first post.

NEW POST

*Server*

API

DATABASE



*Client*

MY BLOG

This is my first post.

ADD POST

MY BLOG

02/23/15

This is my first post.

NEW POST

*Server*

API



DATABASE

?

How is data being sent/received?

*Http*

# HTTP

## Hypertext Transfer Protocol

request-response protocol

sent using TCP/IP sockets

“all about applying verbs to nouns”

nouns: resources (*i.e.*, concepts)

verbs: GET, POST, PUT, DELETE



*More details in  
Socket lecture*

# URL

## Uniform Resource Locator

type of URI (Identifier)

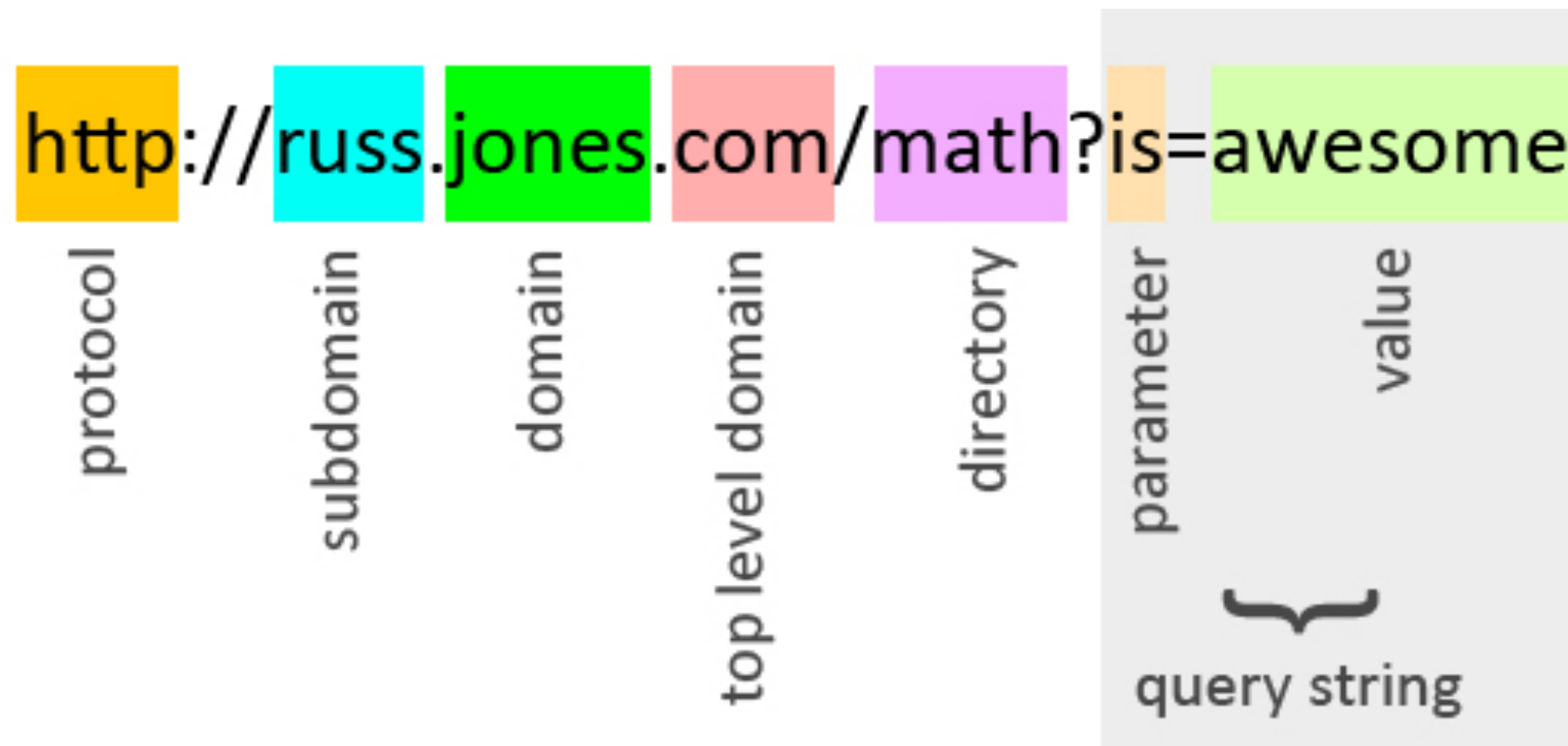
specifies the location of a resource on a network

server responds with **representations** of resources  
and not the resources themselves



*Rest lecture*

# URL ANATOMY



# LOADING A PAGE IN A BROWSER

*representations of resources*

Browser

HTML

Other Resources



→  
HTTP GET

```
http://creativecommons.org
<a><span id="home-button">
</span></a>
<div id="logo">
  <span>
    Creative Commons
  </span>
</div>
```

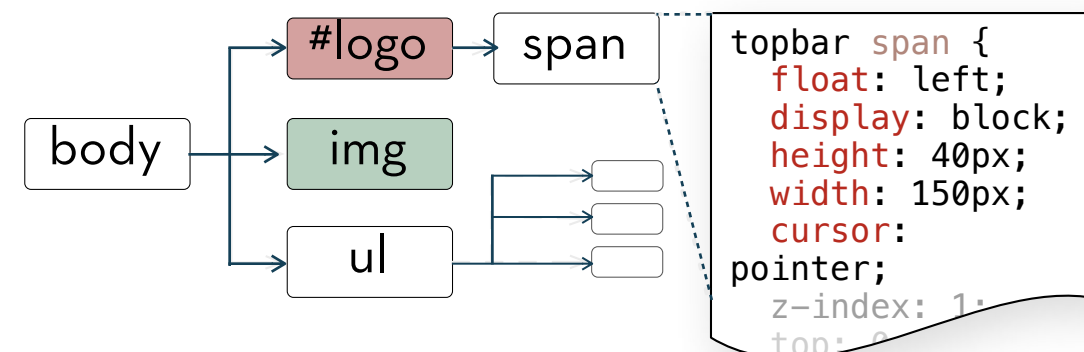
→  
HTTP GET

```
cforms.js
//Collap
String.p
function
return
this.rep

creativecommons.css
topbar #home-button{
  position: relative;
  float: left;
  display: block;
  height: 40px;
  width: 150px;

cc-logo.png
creative commons
```

Document Object Model (DOM)



Rendered Page





Get Creative Commons updates

Subscribe

https://donate.creativecommons.org/?utm\_campaign=2014fund&utm\_source=ccorg1&utm\_medium=site\_header&utm\_medium=site\_h

☐ Preserve log ☐ Disable cache

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline
creativecommons.org	GET	200 OK	text/html	Other	7.0 KB 25.5 KB	510 ms 505 ms	
facebook.png /wp-content/themes/creativecommons.org/img	GET	(failed) net::ERR_B...		creativecommons.o... Parser	0 B 0 B	675 ms -	
style.css /wp-content/themes/creativecommons.org/css	GET	200 OK	text/css	creativecommons.o... Parser	15.7 KB 80.9 KB	268 ms 187 ms	
twitter.png /wp-content/themes/creativecommons.org/img	GET	(failed) net::ERR_B...		creativecommons.o... Parser	0 B 0 B	676 ms -	
modernizr-2.0.6.min.js /wp-content/themes/creativecommons.org/js/libs	GET	200 OK	applicatio...	creativecommons.o... Parser	6.9 KB 15.8 KB	277 ms 265 ms	
widget.css?ver=4.1 /wp-content/plugins/yet-another-related-posts-plugin/style	GET	200 OK	text/css	creativecommons.o... Parser	766 B 771 B	260 ms 248 ms	
pagenavi-css.css?ver=2.70 /wp-content/plugins/wp-pagenavi	GET	200 OK	text/css	creativecommons.o... Parser	621 B 374 B	260 ms 245 ms	
jquery.js?ver=1.11.1 /wordpress/wp-includes/js/jquery	GET	200 OK	applicatio...	creativecommons.o... Parser	32.8 KB 93.6 KB	352 ms 259 ms	
jquery-migrate.js?ver=1.2.1 /wordpress/wp-includes/js/jquery	GET	200 OK	applicatio...	creativecommons.o... Parser	6.1 KB 16.7 KB	373 ms 347 ms	
creativecommons.css	GET	200	text/css	creativecommons.o...	2.3 KB	267 ms	

# HTTP Request

method

url

version

GET /index.html HTTP/1.1

Host: www.example.com

User-Agent: Mozilla/5.0

Accept: text/xml,application/  
xml,application/xhtml+xml,text/html\*/\*

Accept-Language: en-us

Accept-Charset: ISO-8859-1,utf-8

Connection: keep-alive

<blank line>

request  
headers

# GET

*vs*

# POST

retrieve representations of  
resources

no side effects

no data in request body

upload data from the browser  
to server

returns information from the  
server

side effects are likely

data contained in request body

# HTTP Response

version    status code    text explanation

HTTP/1.1 200 OK

Date: Mon, 23 May 2005 22:38:34 GMT

Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)

Content-Type: text/html; charset=UTF-8

Content-Length: 131

response headers

<!DOCTYPE html>

<html>

...

</html>

content

# HTTP Response

HTTP/1.1 200 OK

Date: Mon, 23 May 2005 22:38:34 GMT

Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)

Content-Type: text/html; charset=UTF-8

Content-Length: 131



MIME Type

<!DOCTYPE html>

<html>

...

</html>

# AN SEO'S GUIDE TO HTTP STATUS CODES

Every web page you visit returns a status code, to give the browser additional information and instructions. Search bots see these codes, and some of them can impact SEO. Here are a few of the big ones:

## CAST OF CHARACTERS



## HTTP STATUS CODES

**200**  **OK/Success** Everyone arrives at Page A. There is much rejoicing!

**301**  **Permanent\*** Everyone is redirected to the new location, Page B.

**302**  **Temporary\*** Visitors and bots are redirected. Juice is left behind.

**404**  **Not Found** Original page is gone. Visitors may see a 404 page.

**500**  **Server Error** No page is returned. Everyone is lost and confused :|

**503**  **Unavailable** Asks everyone to come back later. A 404 alternative.

\* Technically, code 301 is "Moved Permanently" and 302 is "Found", but SEOs refer to them as "Permanent Redirect" and "Temporary Redirect".

## THE CANONICAL TAG

**REL**  **Canonical** Alternative to 301-redirects. Visitors still see Page A.

# HTTP STATUS CODES

[moz.com/learn/seo/http-status-codes](https://moz.com/learn/seo/http-status-codes)

# HTTPS

*More details in  
Security lecture*

request and response messages are  
transmitted securely using encryption

Ajax



# AJAX

## **Asynchronous JavaScript and XML**

send and receive data without reloading page

Before, every user interaction required the complete page to be reloaded

# AJAX

Issue HTTP request to the server from Javascript

Process response with Javascript in the browser

# AJAX TECHNOLOGIES

HTML and CSS

DOM

XML

**XMLHttpRequest** object

JavaScript

# JSON

AJAX doesn't require XML

JSON has become de facto standard data interchange format

lightweight and simple format

types: Number, String, Boolean, Array, Object, **null**

objects are key/value pairs

# JSON CODE EXAMPLE

```
{  
  "camelids": [  
    {  
      "name": "llama",  
      "height": 1.8  
    },  
    {  
      "name": "alpaca",  
      "height": 0.9  
    }  
  ]  
}
```

*Look familiar?*

# XMLHttpRequest

```
var xhr = new XMLHttpRequest();  
xhr.onreadystatechange = xhrHandler;  
xhr.open('get', 'llama.json');  
xhr.send(null);
```

# XMLHttpRequest

```
function xhrHandler() {  
    if (xhr.readyState == 4  
        && xhr.status == 200) {  
        var data = JSON.parse(xhr.responseText);  
        myFunction(data);  
    }  
};
```

CODEPEN

# AJAX CHALLENGES

hard to go back to a particular state

URL fragment  
identifier



content retrieved by AJAX not easily indexable

The same origin policy prevents some Ajax techniques from being used across domains

JSONP



callback-style programming is hard to maintain/test



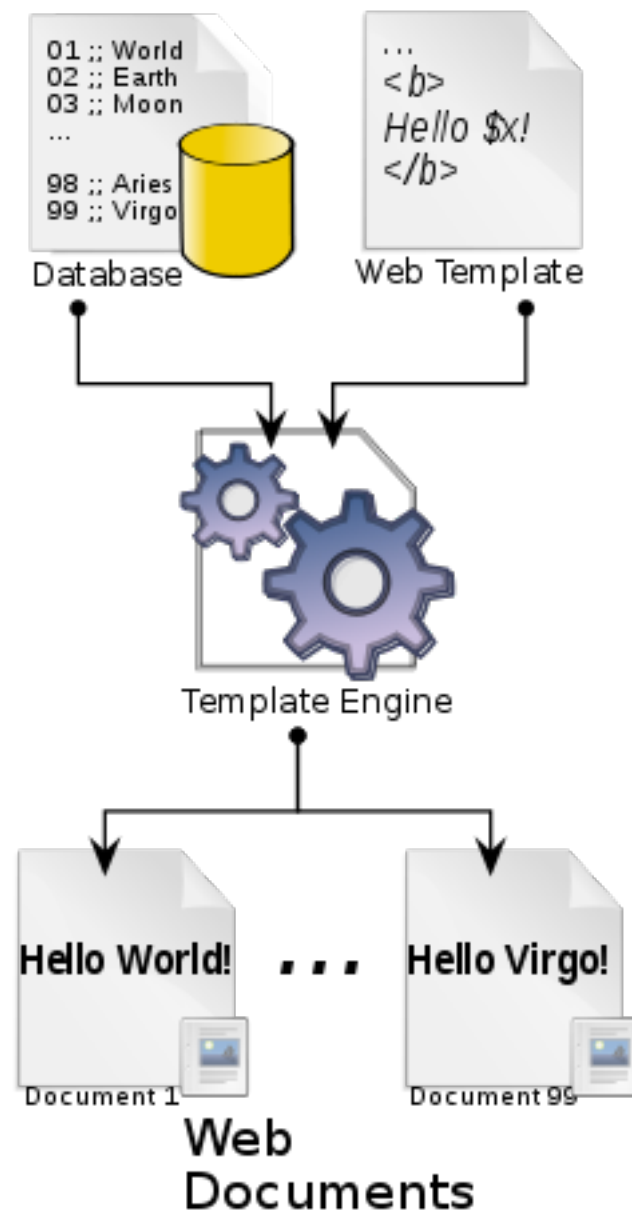
# *Client-side Templating*

# TEMPLATES

common way to generate dynamic HTML for multi-page web sites and apps

separation of markup and data (content)

# SERVER-SIDE TEMPLATES



server puts HTML and data together  
and sends it to the browser

platforms like Rails, PHP, JSP

<http://www.w3.org/TR/XMLHttpRequest/>

# CLIENT-SIDE TEMPLATES

AngularJS 

browser receives HTML and data and puts it together

server serves templates and data required by the templates

made popular by AJAX

*Model View Controller*

# MODEL VIEW CONTROLLER (MVC)

introduced in 1970s as part of SmallTalk

popular in desktop UI development (C++, Java)

more recently introduced to the Web

mental model makes it easier to extend, maintain,  
and test apps

# MODEL VIEW CONTROLLER (MVC)

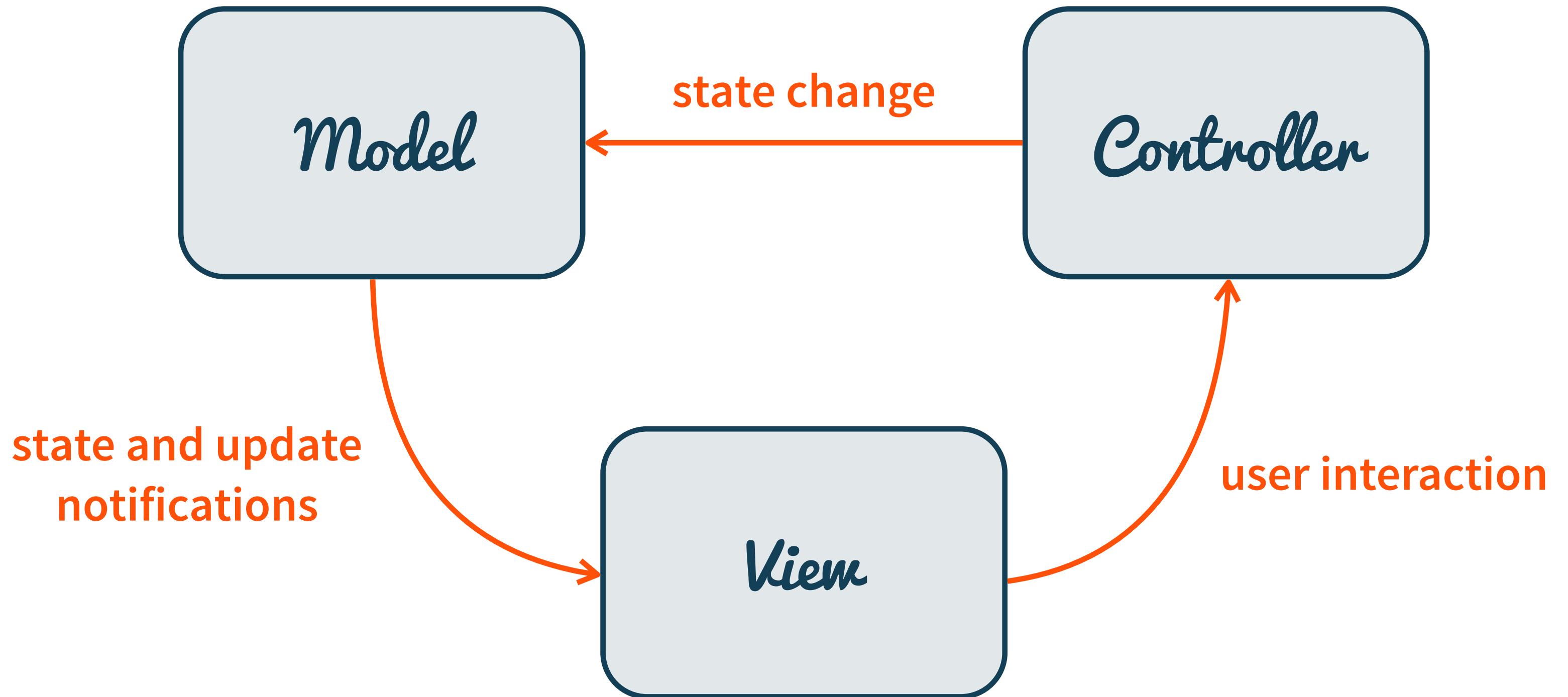
*Separation between*

*Model* managing data

*Controller* application logic

*View* presenting the data

# MODEL VIEW CONTROLLER (MVC)





# MVC CHALLENGE

non-trivial to get the data into the  
correct state, both in the *View* and in  
the *Model*

# *Data Binding*

Just declare mapping between *View*  
and *Model* and have them sync  
automatically?

# DATA BINDING

automatically keep state in View and Model in sync

frameworks provide scaffolding to eliminate a lot of code

# EXAMPLE

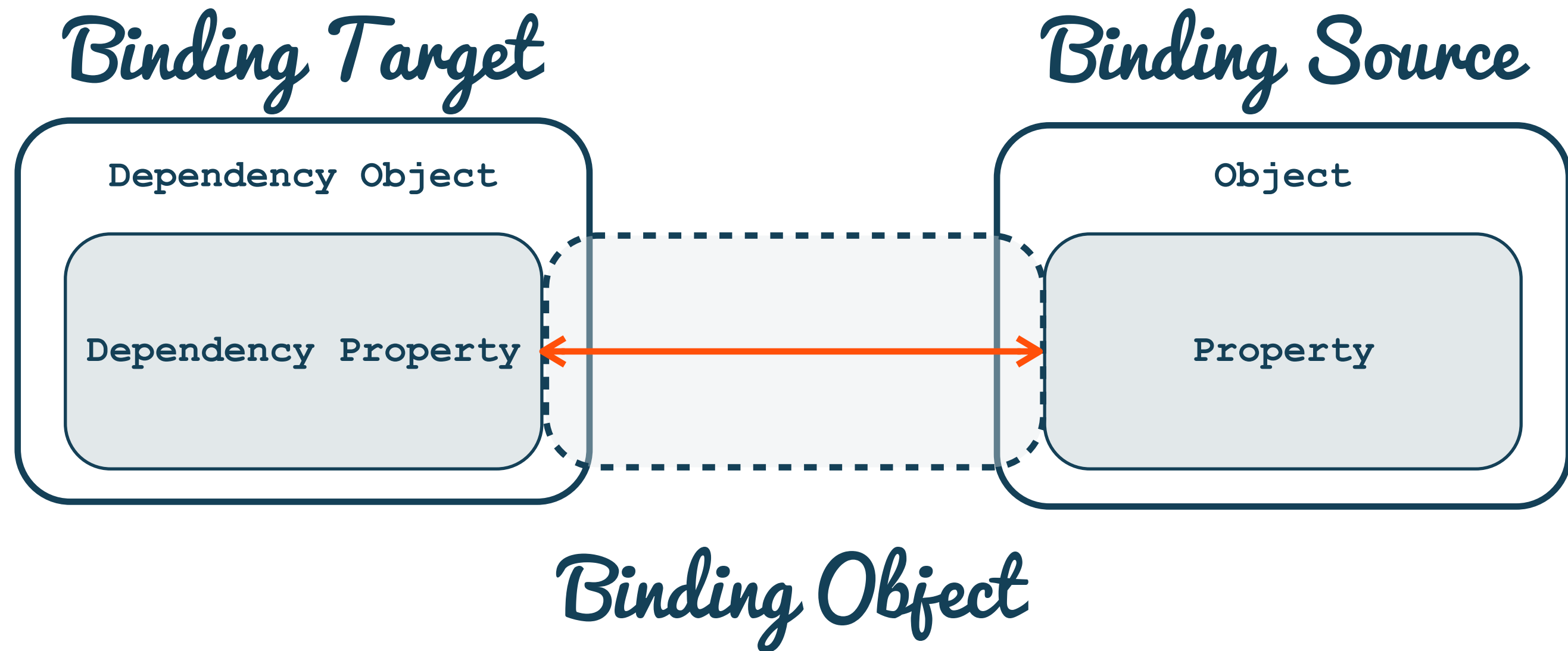
*View*



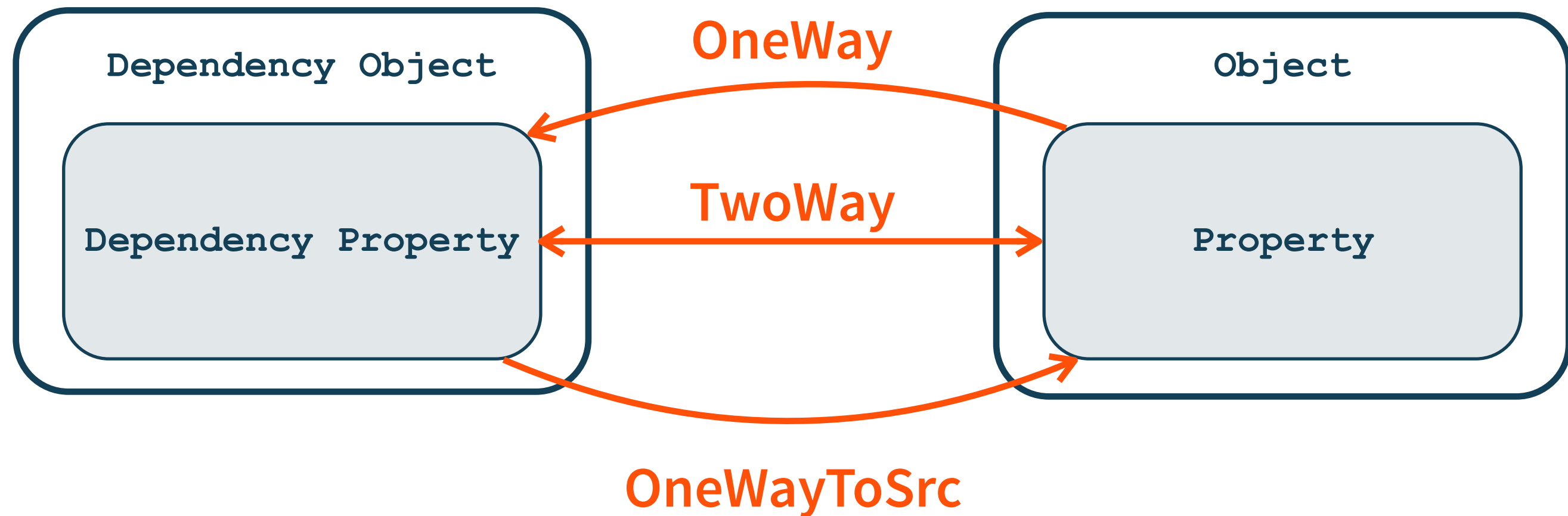
*Model*



# MENTAL MODEL



# MENTAL MODEL



NEXT CLASS: ANGULARJS

[courses.engr.illinois.edu/cs498rk1/](https://courses.engr.illinois.edu/cs498rk1/)