

Building Apps with AngularJS and Breeze

Part 2

Client Validation

John Papa
<http://johnpapa.net>
Twitter: @john_papa



pluralsight 
hardcore developer training

What We'll Cover

Types of Client
Validation

Breeze
Validation
Directives

Free Validation
with Breeze

Creating
Custom
Validation Rules

Validation
Configuration
Options

Styling

Validation Mantra

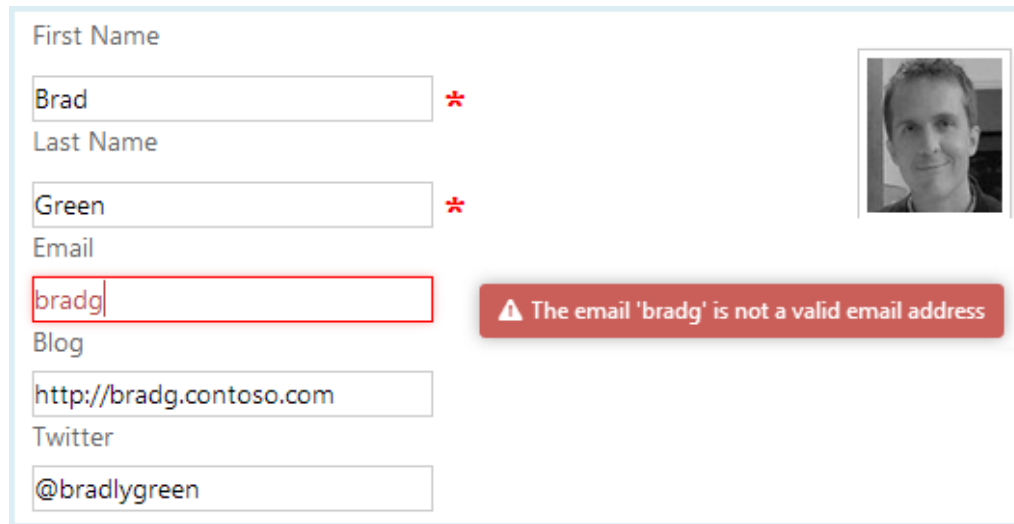
Validation on the Server for Data Integrity

Validation on the Client for User Experience

**ALWAYS
VALIDATE
ON THE SERVER**

Client Validation Experience

Do you validate changes on the client for a better user experience?



First Name

Brad *

Last Name

Green *

Email

bradg

Blog

http://bradg.contoso.com

Twitter

@bradlygreen

⚠ The email 'bradg' is not a valid email address

View Level vs Model Level Validation

View Level

Should you copy business rules to HTML?

Model Level

Shouldn't the model drive the UI?

HTML5 Validation

View Level Validation

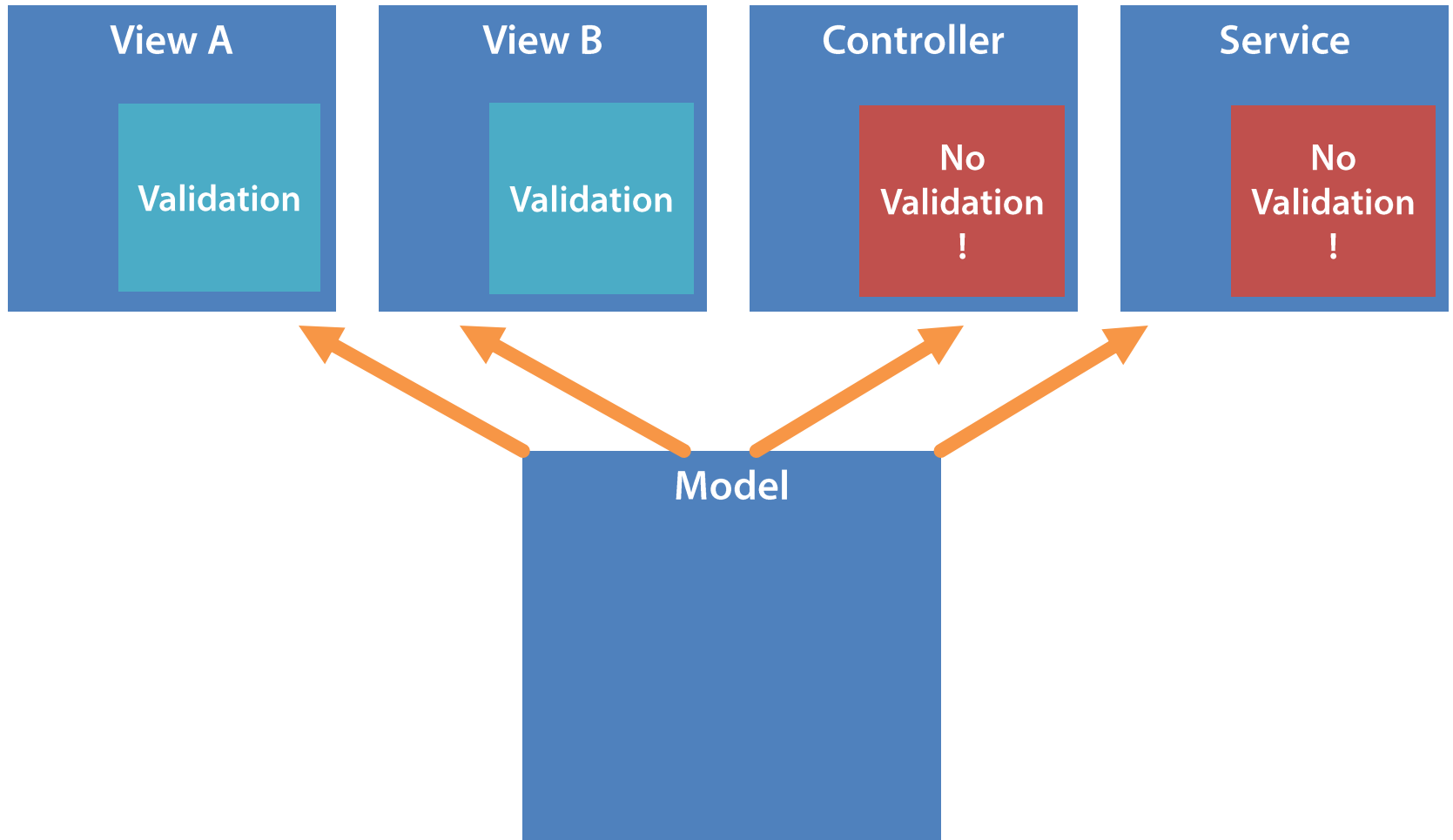
Lots of Attributes

Custom Rules Still Go to
JavaScript

View Has Awareness
of How to Validate

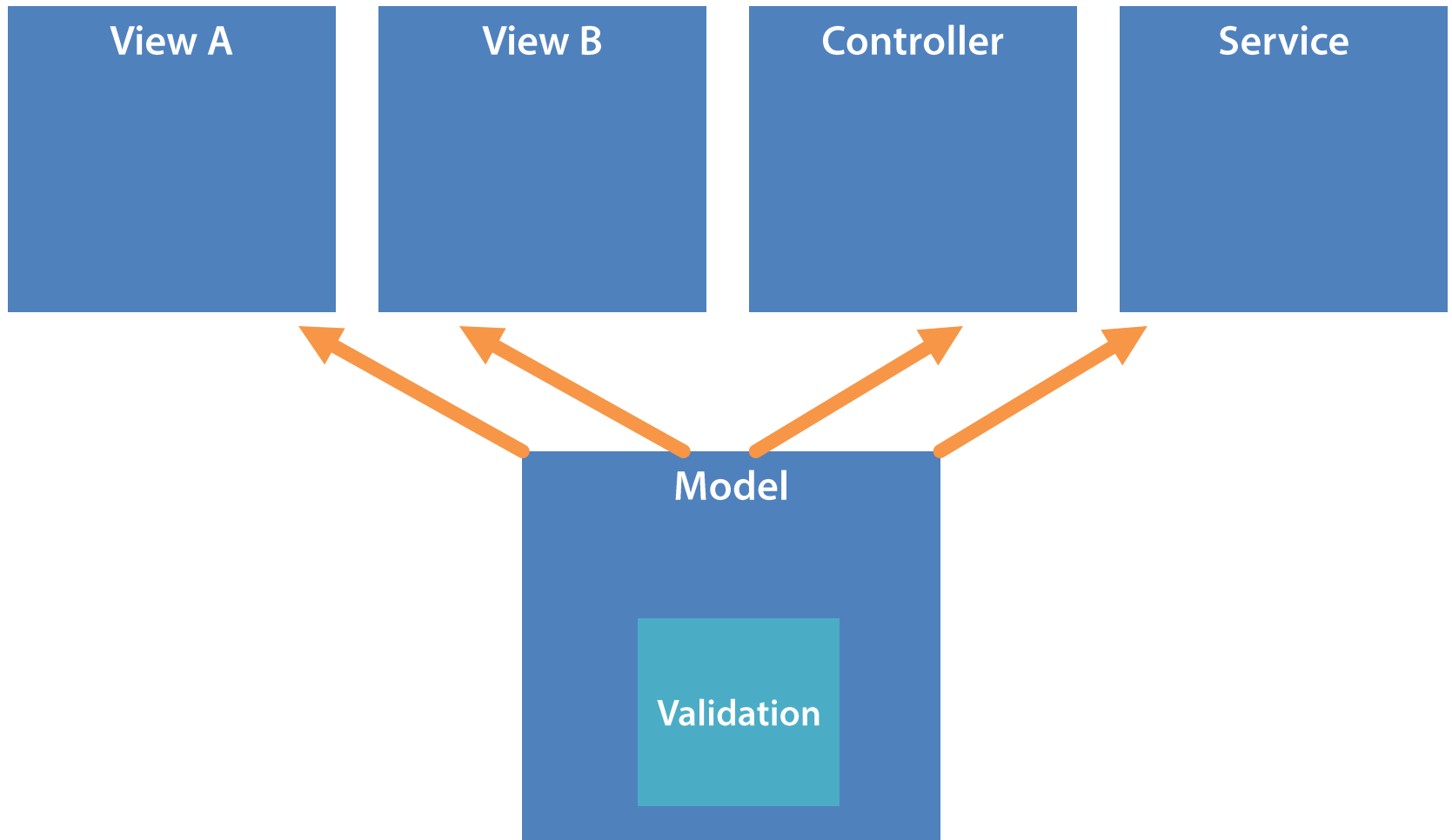
HTML5 Validation

We could miss a place
to validate



Model Validation

No matter where it
changes, we validate



zValidate Directive

Model Level Validation

Responds to Changes on
Properties

Separation of
How You
Calculate the
Validation Error

from How You
Display It

We Get Validation Whether the
Property is Displayed or Not

Validation for User Experience



Breeze Reads Metadata from the Server

Required

MaxLength

```
public class Session
{
    public int Id { get; set; }
    [Required, MaxLength(50)]
    public string Title { get; set; }
    public string Code { get; set; }
    public int SpeakerId { get; set; }
    public int TrackId { get; set; }
    public int TimeSlotId { get; set; }
    public int RoomId { get; set; }
    public string Level { get; set; }
    public string Tags { get; set; }
    public string Description { get; set; }

    public virtual Person Speaker { get; set; }
    public virtual Track Track { get; set; }
    public virtual TimeSlot TimeSlot { get; set; }
    public virtual Room Room { get; set; }

    public virtual ICollection<Attendance>
        AttendanceList { get; set; }
}
```

Model Level Validation

Back Cancel Save Delete

Edit New Session

Title
 *

Track
 *

Time slot
 *

Room
 *

Code

Level

Tags

[Select a person] *

Description

Inconceivable! 'title' is required

Inconceivable! Missing track

Inconceivable! Missing timeSlot

Inconceivable! Missing room

Inconceivable!

[CC Error] Save failed: Session (-1) -
"title" is required;
Session (-1) - 'Missing room';
Session (-1) - 'Missing speaker';
Session (-1) - 'Missing timeSlot';
Session (-1) - 'Missing track'

```
sessiondetail.html
<div class="widget-content user">
  <div>
    <label>Title</label>
    <input data-ng-model="vm.session.title" data-z-validate
  </div>
  <div>
    <div>
      <label>Track</label>
      <select data-ng-options="t.name for t in vm.tracks"
        data-ng-model="vm.session.track" data-z-validate>
    </select>
    </div>
    <div>
      <label>Time slot</label>
      <select data-ng-options="ts.name for ts in vm.timeslots"
        data-ng-model="vm.session.timeSlot" data-z-validate>
    </select>
    </div>
    <div>
      <label>Room</label>
      <select data-ng-options="r.name for r in vm.rooms"
        data-ng-model="vm.session.room" data-z-validate>
    </select>
    </div>
    <div>
      <label>Code</label>
      <input data-ng-model="vm.session.code" placeholder="Code" />
    </div>
    <div>
      <label>Level</label>
      <input data-ng-model="vm.session.level" placeholder="Level" />
    </div>
    <div>
      <label>Tags</label>
      <input data-ng-model="vm.session.tagsFormatted"
    </div>
  </div>
</div>
```

Breeze Validation Directive

```
<div>
  <label>Title</label>
  <input data-ng-model="vm.session.title"
        placeholder="Session title"
        data-z-validate />
</div>
```

Property to Validate



Tell Breeze to
Validate and
Display, if
Appropriate



Validation Examples

Required

Title

Session title



⚠ Inconceivable! 'title' is required

Max Length

Title

This is an extremely tremendously



⚠ Inconceivable! 'title' must be a string with 50 characters or less

Regular Expressions

Email

J

⚠ Inconceivable! The email 'J' is not a valid email address

Applying Custom Validation

Create

```
var name = 'requireReferenceEntity';  
var ctx = { messageTemplate: 'Missing %displayName%', isRequired: true };  
var requireReferenceValidator = new Validator(name, valFunction, ctx);  
function valFunction(value) { return value ? value.id !== 0 : false; }
```

Register

```
Validator.register(requireReferenceValidator);
```

Apply

```
var entityType = metadataStore.getEntityType('session');  
entityType.getProperty('room')  
    .validators.push(requireReferenceValidator);
```


Key Validation Learnings

Types of Client Validation

zValidate

Some Validation is Free

Custom Validation

Configure

Style