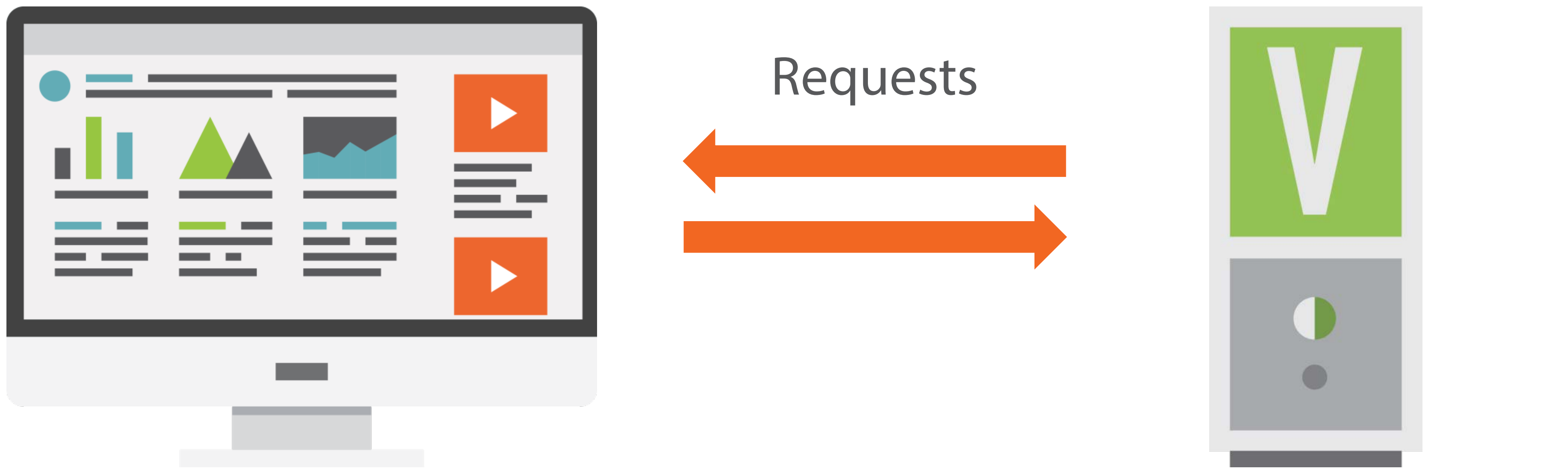
Extending Validation to Improve Data Integrity

Alex Wolf

www.alexwolfthoughts.com

Client and Server Validation



User Experience
(and Security)

Security
(and User Experience)

Default Validation Attributes

```
public class Address
{
      [Required]
      public string Street { get; set; }
}
```

- Required
- Range
- StringLength
- Regular Expression

Validating Models

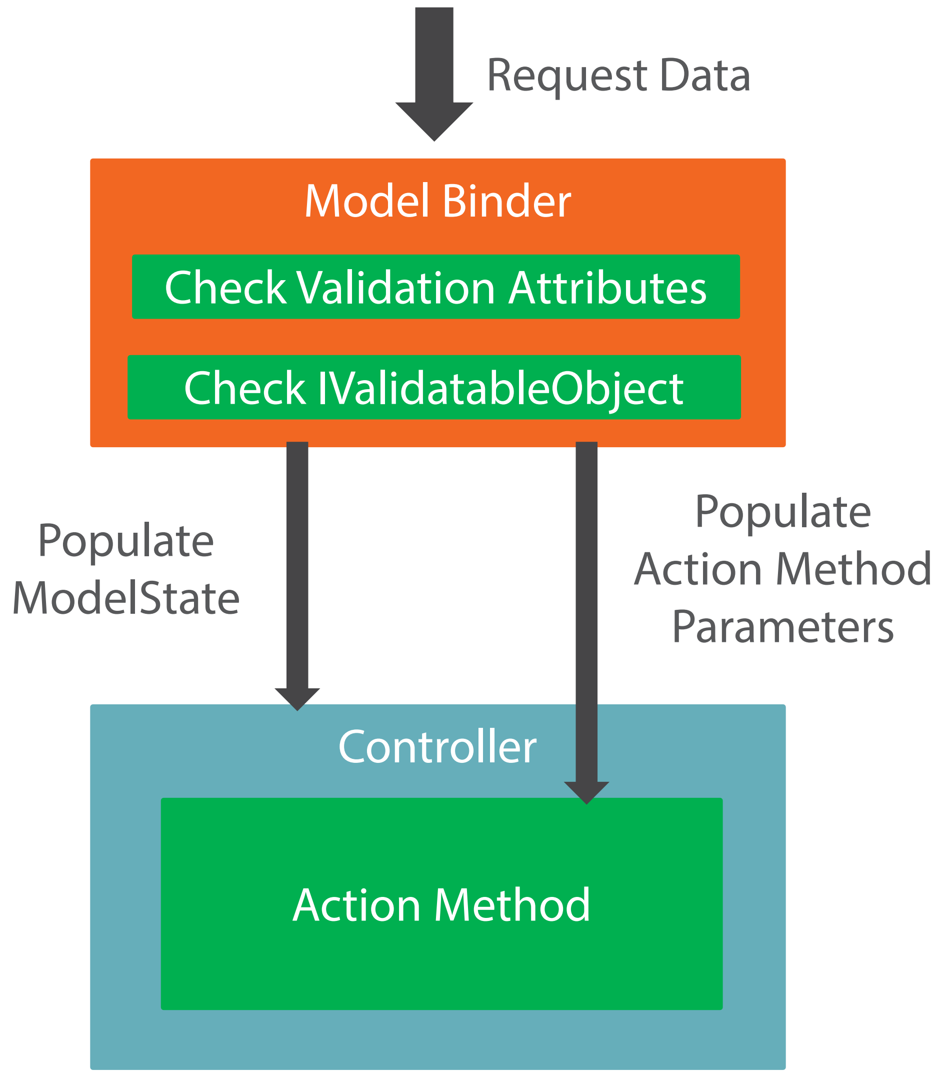
```
public  class Address : IValidatableObject
{
        public string Street { get; set; }

        public IEnumerable<ValidationResult> Validate(ValidationContext
        validationContext)
        {
                // Validation logic here
        }
}
```

Weighing the Pros and Cons

	Attributes	IValidatableObject
Advantages		
Disadvantages		

Model Binding and Validation



```
public ActionResult Index()
{
        if(ModelState.IsValid)
        {
                // Method Logic
        }
}
```

Summary

- MVC provides default validation attributes for common tasks
- Custom validation attributes are a reusable, isolated extension point
- The IValidatableObject interface can provide a convenient alternative for model level validation