specialist ICT learning

GURUTEAM

H&R BLOCK

AngularJS: Databinding

GuruTeam Instructor: Sang Shin

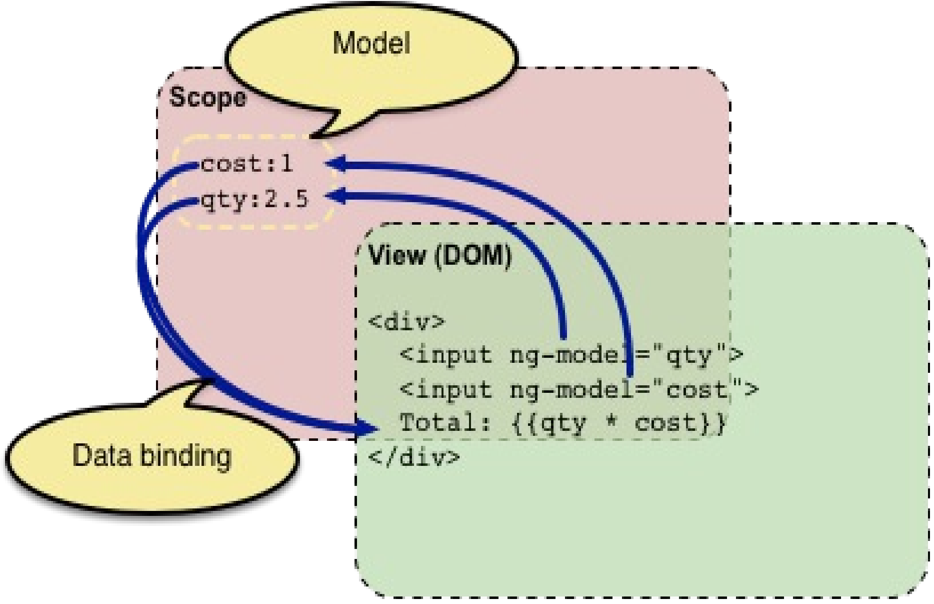
Topics

- Two-way data-binding
- Data-binding internals
- ngModelOptions

Two-way Data-binding

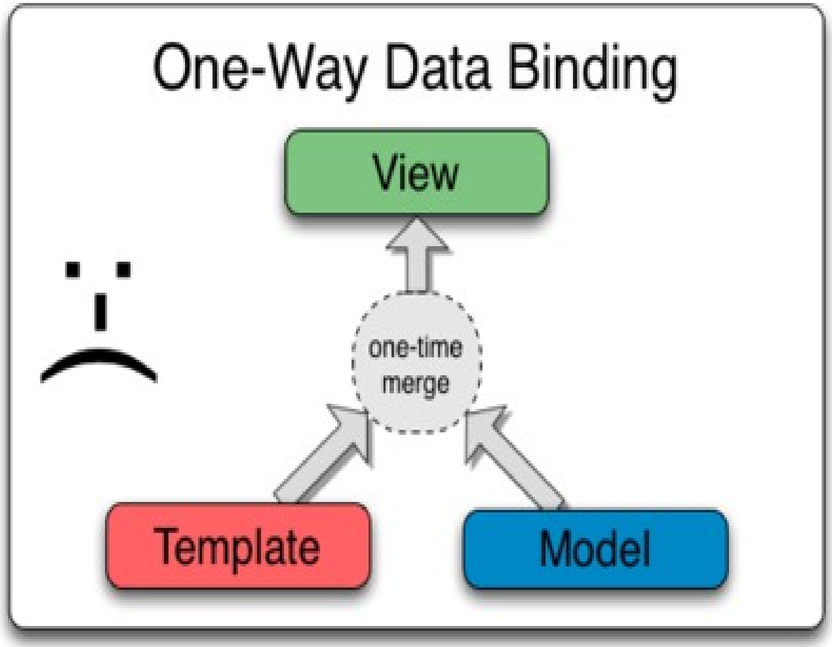
What is (Two-way) Data-binding?

- Data-binding in Angular apps is the automatic synchronization of data between the model and view components
  - Lets you treat the model as the single-source-of-truth in your application
  - The view is a projection of the model at all times
  - When the model changes, the view reflects the change



```
<div>
  <input ng-model="qty">
  <input ng-model="cost">
  Total: {{qty * cost}}
</div>
```

Data-binding in Classical Template System

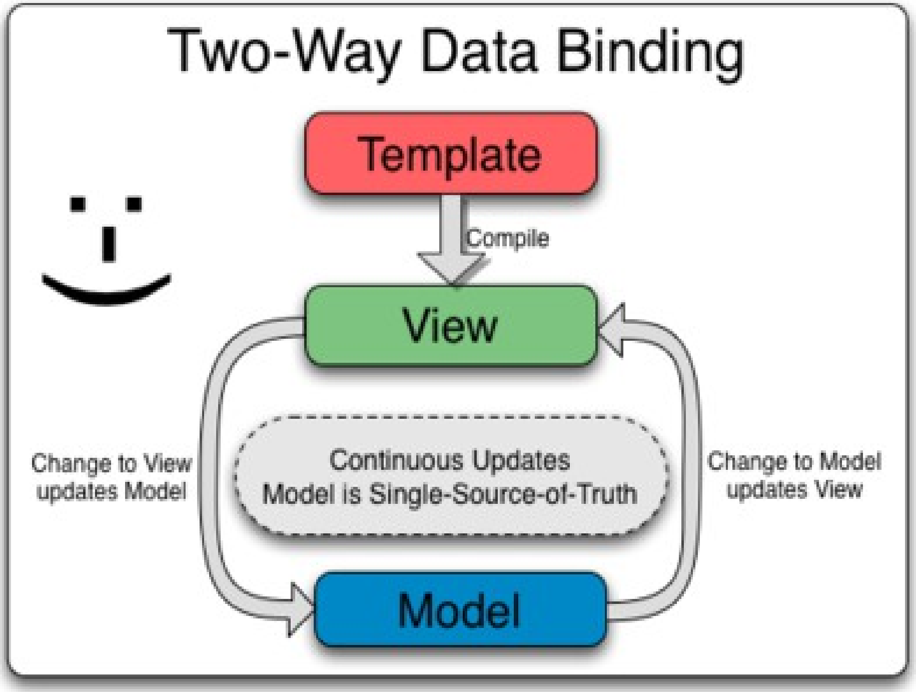


Most templating systems bind data in only one direction: they merge template and model components together into a view.

After the merge occurs, changes to the model or related sections of the view are not automatically reflected in the view.

Worse, any changes that the user makes to the view are not reflected in the model. This means that the developer has to write code that constantly syncs the view with the model and the model with the view.

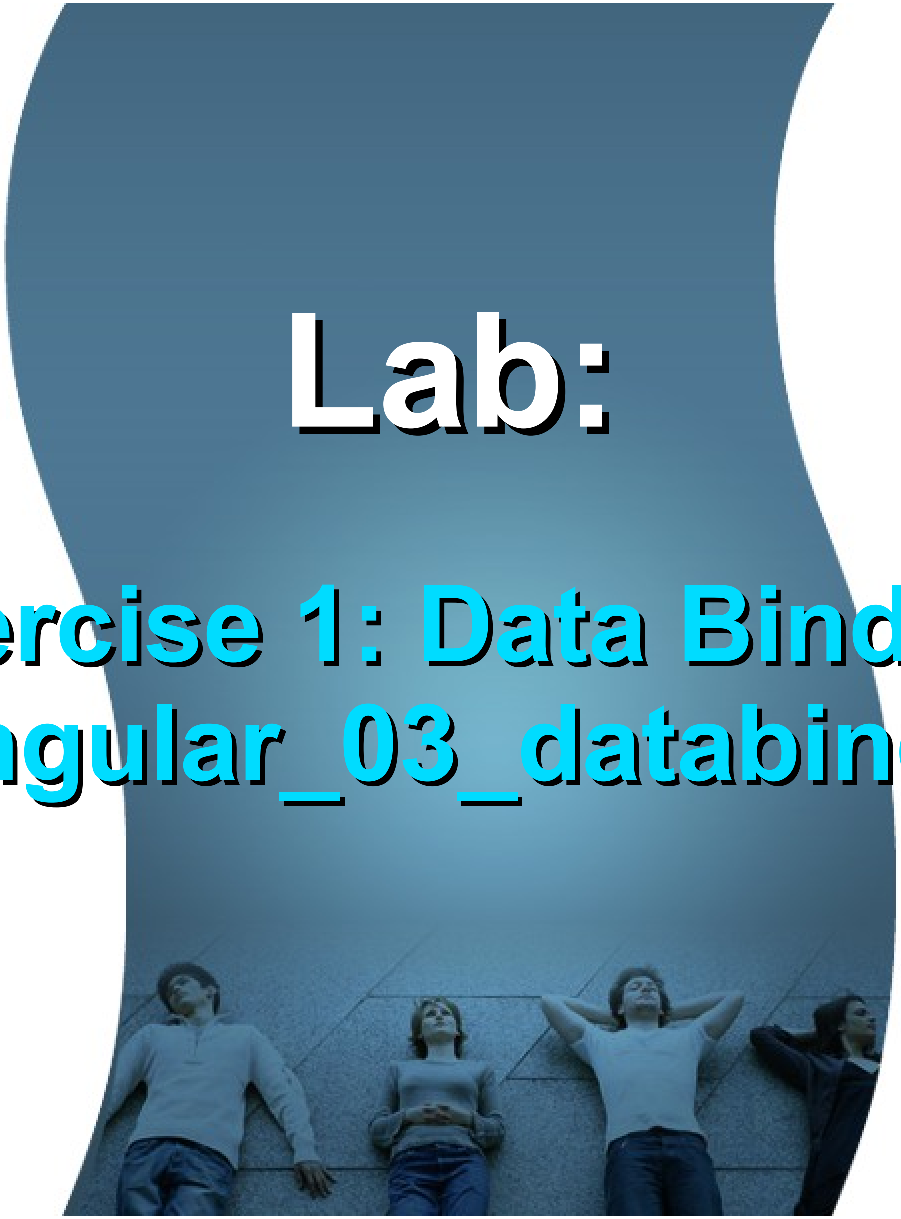
Data-binding in AngularJS Template System


Two-Way Data Binding
Template
Compile
View
Continuous Updates
Model is Single-Source-of-Truth
Change to View updates Model
Change to Model updates View
Model

First the template (which is the uncompiled HTML along with any additional markup or directives) is compiled on the browser.

The compilation step produces a live view. Any changes to the view are immediately reflected in the model, and any changes in the model are propagated to the view.

The model is the single-source-of-truth for the application state, greatly simplifying the programming model for the developer. You can think of the view as simply an instant projection of your model.



Lab:

Exercise 1: Data Binding
3303_angular_03_databinding.zip

What is Data-binding again?

- The core feature of data-binding
  - When you change something in the view, the scope model automatically gets updated
  - Similarly, whenever the scope model changes, the view updates itself with the new value
- How does AngularJS do this?
  - When you write an expression *{{aModel}},* behind the scenes, Angular sets up a watcher on *aModel*, which in turn updates the view whenever the model changes
  - It works like following:

```
// When 'aModel” changes, the listener function is called
$scope.$watch('aModel', function(newValue, oldValue) {
  //update the view with newValue
});
```

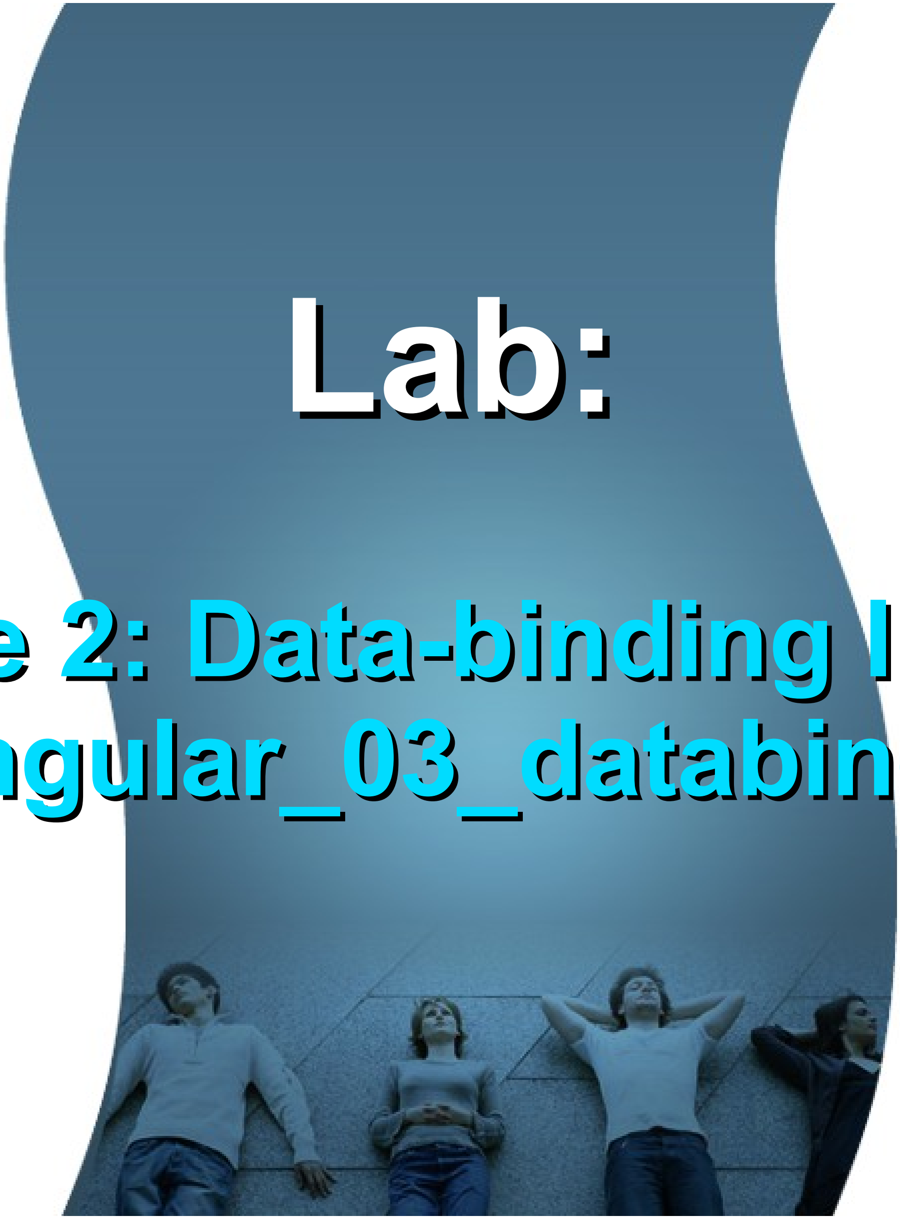
Data-binding Internals

How Data-binding triggered?

- How does Angular figure out when to call this listener function?
  - In other words, how does AngularJS know when aModel changes so it can call the corresponding listener
- It’s the $digest cycle where the watchers are fired and it is triggered
  - When a model is changed by a handler invoked via *ng-click*
  - When a model annotated with *ng-model* is changed

When do you call $apply manually?

- If you change any model outside of the Angular context, then you need to inform Angular of the changes by calling *$apply()* manually
  - It’s like telling Angular that you are changing some models and it should fire the watchers so that your changes propagate properly
- For example, if you use JavaScript’s *setTimeout()* function to update a scope model, Angular has no way of knowing what you might change. In this case it’s your responsibility to call *$apply()* manually



Lab:

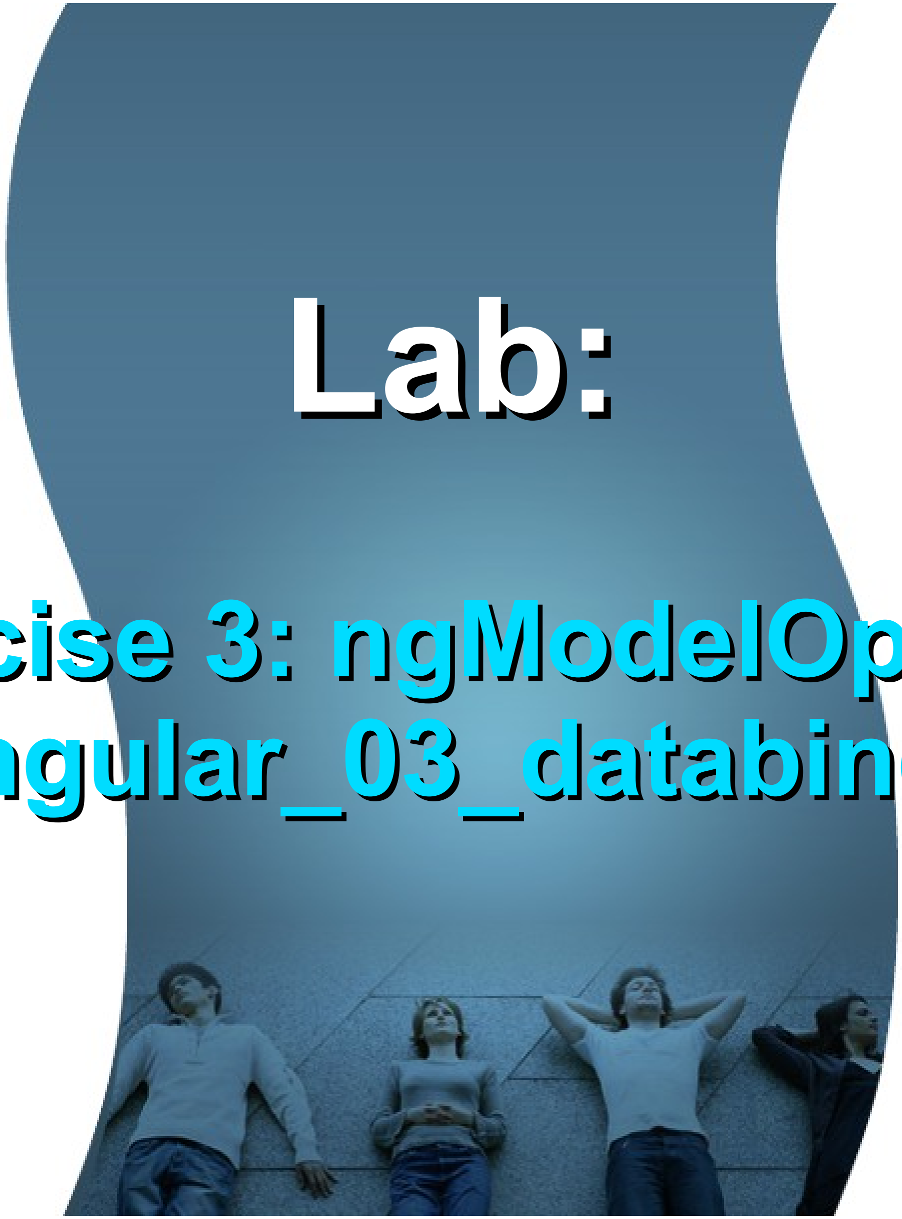
Exercise 2: Data-binding Internals 3303_angular_03_databinding.zip

ng-model options

ng-model-options (ngModelOptions)

- Used to tune how model updates are done
  - By using *ngModelOptions* you can specify a custom list of events that will trigger a model update and/or a debouncing delay so that the actual update only takes place when a timer expires.
- Options
  - *updateOn* - string specifying which event should the input be bound to
  - *debounce* - integer value which contains the debounce model update value in milliseconds. A value of 0 triggers an immediate update.

```
ng-model-options="{ updateOn: 'default blur', debounce: {'default': 500, 'blur': 0} }
```



Lab:

Exercise 3: ngModelOptions
3303_angular_03_databinding.zip

Who are GuruTeam?

- Specialist onsite training in Linux, Cloud, Database, Architecture, Software and Web Development Technologies
- Accredited by the LPI, CompTIA, Hortonworks and the Cloud Credential Council to deliver training, examinations and certifications.
- Over 230 courses available
- All GuruTeam instructors have extensive real-world experience in their technologies
- Clients are indigenous Irish Companies and Multinationals
- We can bring high spec preconfigured equipment for deliveries in Ireland, the UK and Europe.

Linux Professional Institute Approved Training Partner

CompTIA AUTHORIZED PARTNER

Hortonworks Partner TRAINING DELIVERY

CLOUD CREDENTIAL COUNCIL ACCREDITED PARTNER

Exceed your expectations...
GURUTEAM