

Angular ui-router and Resolve

Jogesh K. Muppala



THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系



香港科技大學
THE HONG KONG UNIVERSITY OF
SCIENCE AND TECHNOLOGY

Angular Controller and Asynchronous Calls

- The Angular controllers that we have implemented deal with the asynchronous calls within the controller using promises
- The controller should be mainly concerned with bridging between the services and the view
- Can we shift the asynchrony out of the controller?

Angular ui-router and Resolve

- Each state in the ui-router can include a resolve object
 - map of dependencies that should be injected into the controller
 - key-value pairs
 - key: name of the resolved dependency that you can inject into the controller
 - value: function that returns the value of the dependency
 - Resolution is done before the controller is instantiated

Example: Resolve

```
.state('app.favorites', {  
  url: '/favorites',  
  views: {  
    'mainContent': {  
      templateUrl: 'templates/favorites.html',  
      controller: 'FavoritesController',  
      resolve: {  
        dishes: ['menuFactory', function(menuFactory){  
          return menuFactory.query();  
        }],  
        favorites: ['favoriteFactory', function(favoriteFactory) {  
          return favoriteFactory.getFavorites();  
        }]  
      }  
    }  
  }  
})
```

Example: Resolve

```
.controller('FavoritesController', ['$scope', 'dishes', 'favorites', 'favoriteFactory', ...  
    function ($scope, dishes, favorites, favoriteFactory, ... ) {
```

```
    $scope.baseURL = baseURL;
```

```
    $scope.shouldShowDelete = false;
```

```
    $scope.favorites = favorites;
```

```
    $scope.dishes = dishes;
```

Exercise: Angular ui-router and Resolve

- Use the resolve object to resolve the dependencies in the state
- Inject the data after resolution into the controller