

Caching and Sharing Data



Brice Wilson

@brice_wilson | www.BriceWilson.net

Overview

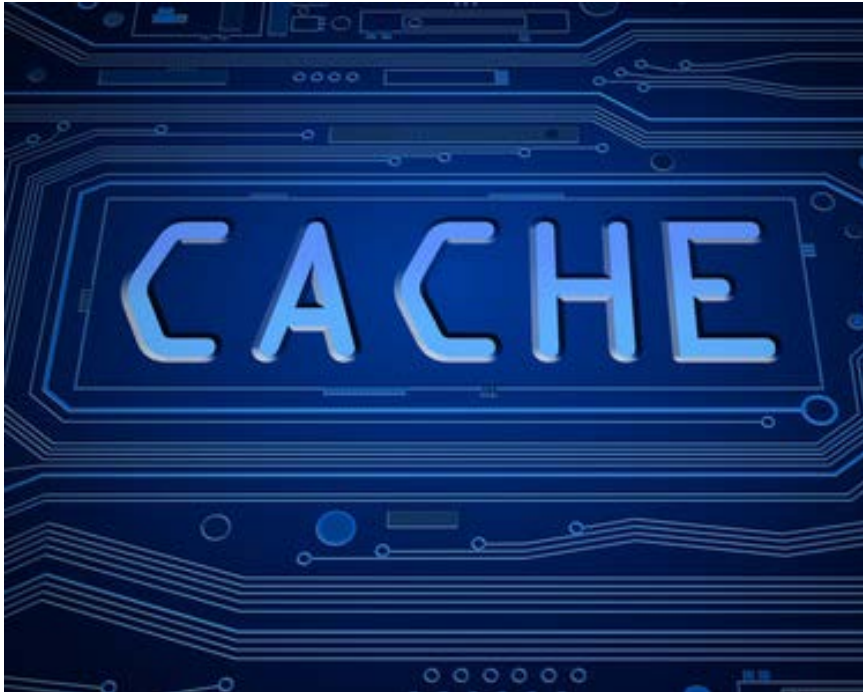


Understanding `$cacheFactory` and cache objects

Caching `$http` requests

Using custom services to share data between controllers

Caching



`$cacheFactory` creates cache objects

Add and remove items to and from cache objects

Cache objects requested at any time from `$cacheFactory`

Primarily used with `$http`

```
var dataCache = $cacheFactory( 'bookLoggerCache' );
```

```
var dataCache = $cacheFactory.get( 'bookLoggerCache' );
```



Using \$cacheFactory

Call as a function to create new cache objects

Call get() method to retrieve existing cache objects

Using Cache Objects

- `put (key, value)`
- `get (key)`
- `remove (key)`
- `removeAll ()`
- `destroy ()`
- `info ()`




\$http Caching

- Set with cache property
- Added to \$http cache object by default
- URL is the cache key
- Response object is the cache value
- Promises resolved with cached response objects




\$http Cache Objects

// use the default \$http cache object

```
return $http({  
  method: 'GET',  
  url: 'api/books',  
  cache: true   
});
```

// use a custom cache object

```
var dataCache = $cacheFactory.get('bookLoggerCache');  
return $http({  
  method: 'GET',  
  url: 'api/books',  
  cache: dataCache   
});
```

Sharing Data

```
// Service (bookService)  
var getBooks = function() { ... };  
var favoriteBook = 'Anna Karenina';
```

```
graph TD; A["// Service (bookService)  
var getBooks = function() { ... };  
var favoriteBook = 'Anna Karenina';"] --> B["// Controller 1  
vm.bookSvc = bookService;"]; A --> C["// Controller 2  
vm.bookSvc = bookService;"]; B --> D["// View 1  
<div>  
  {{ vm.bookSvc.favoriteBook }}  
</div>"]; C --> E["// View 2  
<div>  
  {{ vm.bookSvc.favoriteBook }}  
</div>"];
```

```
// Controller 1  
vm.bookSvc = bookService;
```

```
// View 1  
<div>  
  {{ vm.bookSvc.favoriteBook }}  
</div>
```

```
// Controller 2  
vm.bookSvc = bookService;
```

```
// View 2  
<div>  
  {{ vm.bookSvc.favoriteBook }}  
</div>
```


Summary



Caching to prevent unnecessary processing

Reduces network traffic and database load

Cache objects and custom services allow easy data sharing