Communicating Through a State Management Service

Deborah Kurata
CONSULTANT | SPEAKER | AUTHOR | MVP | GDE

@deborahkurata | blogs.msmvps.com/deborahk/

Managing State

Property Bag

Basic State Management

State Management with Notifications

ngrx/Redux

```
@Injectable()
export class ParamService {
    showImage: boolean;
    filterBy: string;
}
```

Module
Overview

State Management Service
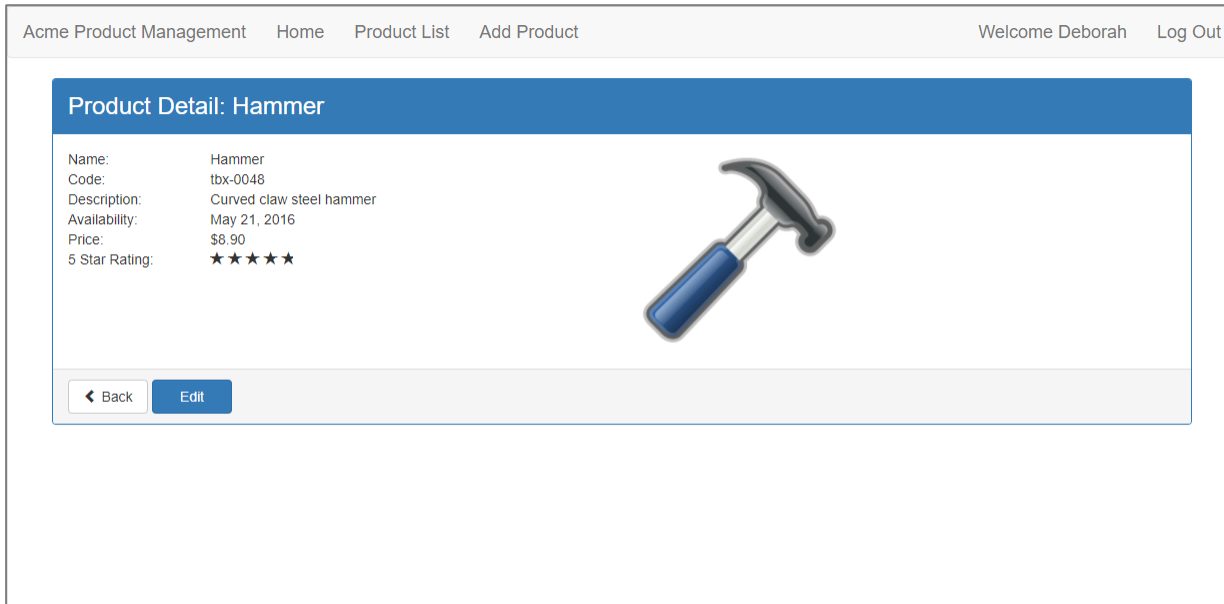- Sharing Entity State

Displaying Concurrent Components

Keeping State in Sync

Change Detection
- Communicating State Changes

Retrieving Entity Data

Acme Product Management   Home   Product List   Add Product   Welcome Deborah   Log Out

Product Detail: Hammer

Name: Hammer
Code: tbx-0048
Description: Curved claw steel hammer
Availability: May 21, 2016
Price: $8.90
5 Star Rating: ★ ★ ★ ★ ★

‹ Back   Edit

Data Access Service

```
@Injectable()
export class ProductService {
    getProducts() {}
    getProduct(id: number) {}

    createProduct(product: IProduct) {}
    deleteProduct(id: number} {}
    updateProduct(product: IProduct) {}
}
```

Backend Server

Sharing Entity State

Acme Product Management   Home   Product List   Add Product   Welcome Deborah   Log Out

Product Detail: Hammer

Name: Hammer
Code: tbx-0048
Description: Curved claw steel hammer
Availability: May 21, 2016
Price: $8.90
5 Star Rating: ★ ★ ★ ★ ★

◄ Back   Edit
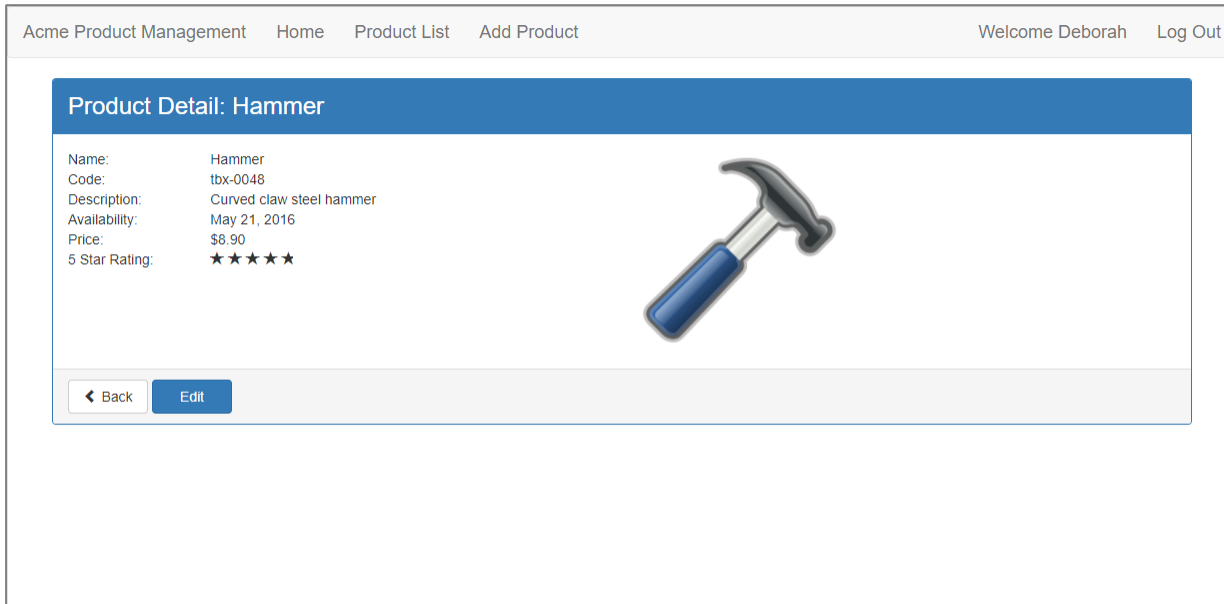
State Management Service

```
@Injectable()
export class ProductService {
    private products: IProduct[];

    getProducts() {}
    getProduct(id: number) {}

    createProduct(product: IProduct) {}
    deleteProduct(id: number) {}
    updateProduct(product: IProduct) {}
}
```

Backend Server

Demo

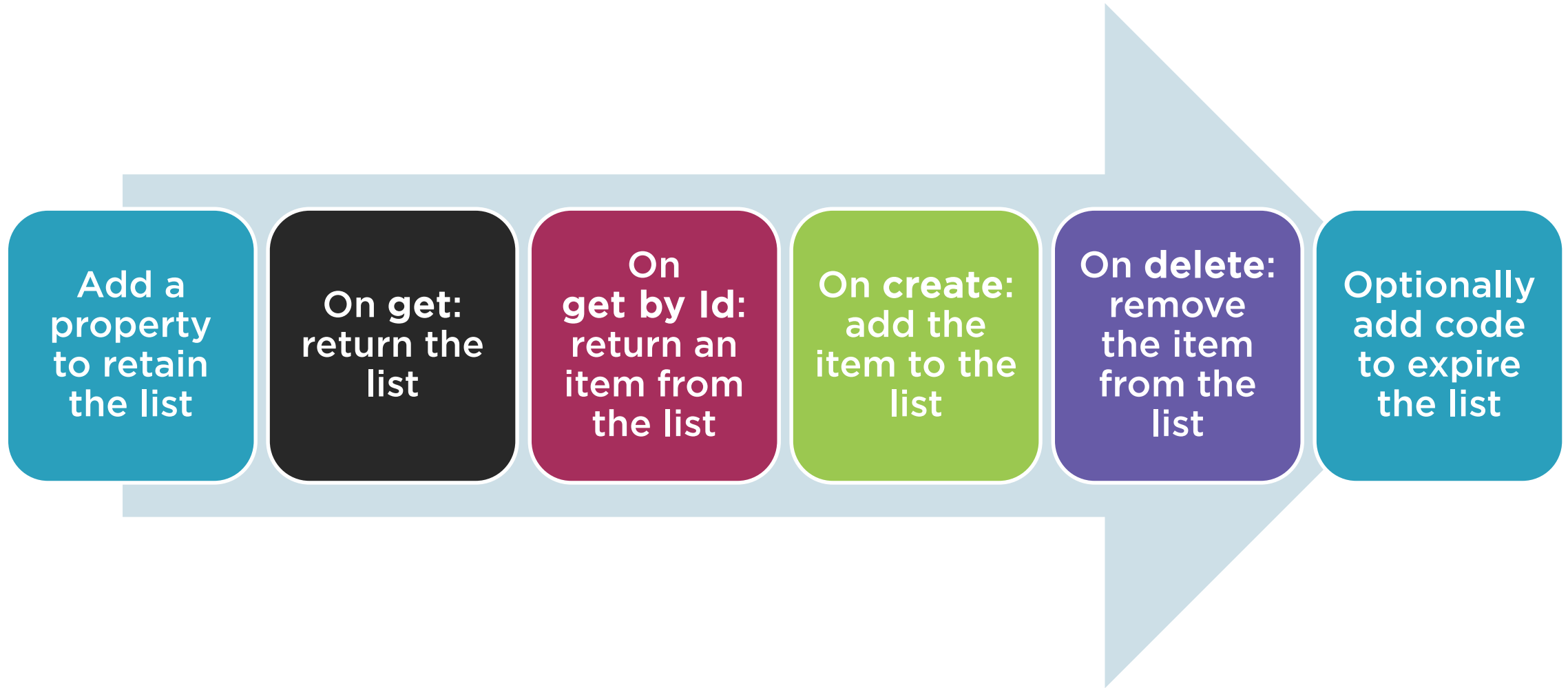**Current Data Access Service**

State Management Service

- **Provide state values**
- **Maintain and update state**
- **Observe state changes**

State Management Service

Demo

State Management Service:

Retrieving State

Demo

State Management Service:

Maintaining State

Product List

Filter by: [   ]

Filtered by: none Hits:5

Show Image	Product	Code	Available	Price	5 Star Rating	
	Leaf Rake	gdn-0011	March 19, 2016	$19.95	★★★	Edit
	Garden Cart	gdn-0023	March 18, 2016	$32.99	★★★★	Edit
	Hammer	tbx-0048	May 21, 2016	$8.90	★★★★★	Edit
	Saw	tbx-0022	May 15, 2016	$11.55	★★★★	Edit
	Video Game Controller	gmg-0042	October 15, 2015	$35.95	★★★★★	Edit

Displaying Concurrent Components

Demo

Keeping State in Sync

Binding

Product Detail Template

{{pageTitle}}

Product Detail Component

```
export class ProductDetailComponent {
    pageTitle: string = 'Product Detail';
}
```

Product Detail

Name:	Hammer
Code:	tbx-0048
Description:	Curved claw steel hammer
Availability:	May 21, 2017
Price:	$8.90
5 Star Rating:	★ ★ ★ ★ ★

Edit

Binding

Product Detail Template

```
Name: {{product.productName}}
Code: {{product.productCode}}
Description: {{product.description}}
```

Product Detail Component

```
export class ProductDetailComponent {
    product: IProduct;
}
```

Product Detail

Name:	Hammer
Code:	tbx-0048
Description:	Curved claw steel hammer
Availability:	May 21, 2017
Price:	$8.90
5 Star Rating:	★★★★★

Edit

Change Detection

Product Detail Template

```
Name: {{product.productName}}
Code: {{product.productCode}}
Description: {{product.description}}
```

Product Detail Component

```
export class ProductDetailComponent {
    product: IProduct;
}
```

```
this.product.productName = 'Steel Hammer';
```

Product Detail

Name:	Steel Hammer
Code:	tbx-0048
Description:	Curved claw steel hammer
Availability:	May 21, 2017
Price:	$8.90
5 Star Rating:	★ ★ ★ ★ ★

Edit

Change Detection

Product Shell Detail Template

```
Name: {{product.productName}}
Code: {{product.productCode}}
Description: {{product.description}}
```

Product Shell Detail Component

```
get product(): IProduct {
  return this.productService.currentProduct;
}
```

Product Service

```
export class ProductService {
    currentProduct: IProduct;
}
```

Product Detail: Hammer

Name:	Hammer
Code:	tbx-0048
Description:	Curved claw steel hammer
Availability:	May 21, 2017
Price:	$8.90
5 Star Rating:	★★★★★

Edit

Product Shell List Component

```
onSelected(product: IProduct) {
    this.productService.currentProduct =
                              product;
}
```

Change Detection

Product Shell Detail Template

```
Name: {{product.productName}}
Code: {{product.productCode}}
Description: {{product.description}}
```

Product Shell Detail Component

```
get product(): IProduct {
  return this.productService.currentProduct;
}
```

Product Service

```
export class ProductService {
    currentProduct: IProduct;
}
```

Product Shell List Component

```
onSelected(product: IProduct) {
    this.productService.currentProduct =
                        product;
}
```

Product Detail: Video Game Controller

Name: Video Game Controller
Code: gmg-0042
Description: Standard two-button
video game controller
Availability: October 15, 2017
Price: $35.95
5 Star ★★★★⯪
Rating:

Edit

Guidelines & Summary: Communicating through a State Management Service

Property Bag vs. State Management Service

Property Bag Service

```
@Injectable()
export class ParamService {
    showImage: boolean;
    filterBy: string;
}
```

o **Retain view settings**

o **Pass data between components**

State Management Service

```
@Injectable()
export class ProductService {
    private products: IProduct[];

    getProducts() {}
    getProduct(id: number) {}

    createProduct(product: IProduct) {}
    deleteProduct(id: number) {}
    updateProduct(product: IProduct) {}
}
```

o **Retrieve, manage, store state**

o **Share state between components**

State Management Service

o **Encapsulates retrieve and store operations**

o **Retains and shares state values**

o **Minimizes hits to the backend server**

    o **Improves performance**

o **Provides change notifications for bound values using a getter**

Considerations:

o **Stale data**

o **No explicit change notifications**

o **State is not immutable**

```
getProducts(): Observable<IProduct[]>
{
    if (this.products) {
        return of(this.products);
    }
    ...
}
```

Getter

Benefits:

- Keeps bound data in sync
  - Define a property in a service
  - Bind that property in a template
  - Use a getter in the component class

- Simple

- Easy to understand

Considerations:

- Only works with bound data

```
get product(): IProduct | null {
    return this.productService.currentProduct;
}
```

Timer

Benefits:

o **Poll for changes**

Considerations:

o **Too short -> more processing than needed**

o **Too long -> delayed reaction**

o **Possible race conditions**

```
ngOnInit() {
    timer(0, 1000).subscribe(t =>
        console.log(this.prod));
}
```

Summary

State Management Service
- Sharing Entity State

Displaying Concurrent Components

Keeping State in Sync

Change Detection
- Communicating State Changes

Managing State

Property Bag

Basic State Management

State Management with Notifications

ngrx/Redux

```
@Injectable()
export class ProductService {
    private products: IProduct[];

    getProducts() {}
    getProduct(id: number) {}
    createProduct(product: IProduct) {}
    ...
}
```