

# View-Based Navigation

Brian Lagunas

[www.pluralsight.com](http://www.pluralsight.com)



# Outline

- Basic Region Navigation
- View and ViewModel Participation
- Passing Parameters
- Navigating to Existing Views
- Confirming or Canceling Navigation
- Navigation Journal

# Basic Region Navigation

- View replaces view
- `INavigateAsync.RequestNavigate`

# RequestNavigate

- **Region**

```
IRegion region = ...;  
region.RequestNavigate(new Uri("MyView", UriKind.Relative));
```

- **RegionManager**

```
IRegionManager regionManager = ...;  
regionManager.RequestNavigate(RegionNames.ContentRegion,  
    new Uri("MyView", UriKind.Relative));
```

# Basic Region Navigation

- View replaces view
- `INavigateAsync.RequestNavigate`
- Based on URIs
- Type must register as Object

# Register Types as Object

- **Standard registration**

```
Container.RegisterType<HomeView>("HomeView");
```

```
Container.RegisterType<IHomeView, HomeView>("HomeView");
```

- **Navigation registration**

```
Container.RegisterType<object, HomeView>("HomeView");
```

```
Container.RegisterType(typeof(object), typeof(HomeView), "HomeView");
```

# Basic Region Navigation

- View replaces view
- `INavigationAsync.RequestNavigate`
- Based on URIs
- Type must register as Object
- View or ViewModel first

# Basic Region Navigation – cont.

- Navigation Callback

```
private void Navigate(string navigatePath)
{
    RegionManager.RequestNavigate(RegionNames.ContentRegion,
                                  navigatePath, NavigationCompleted);
}

private void NavigationCompleted(NavigationResult result)
{
    ...
}
```

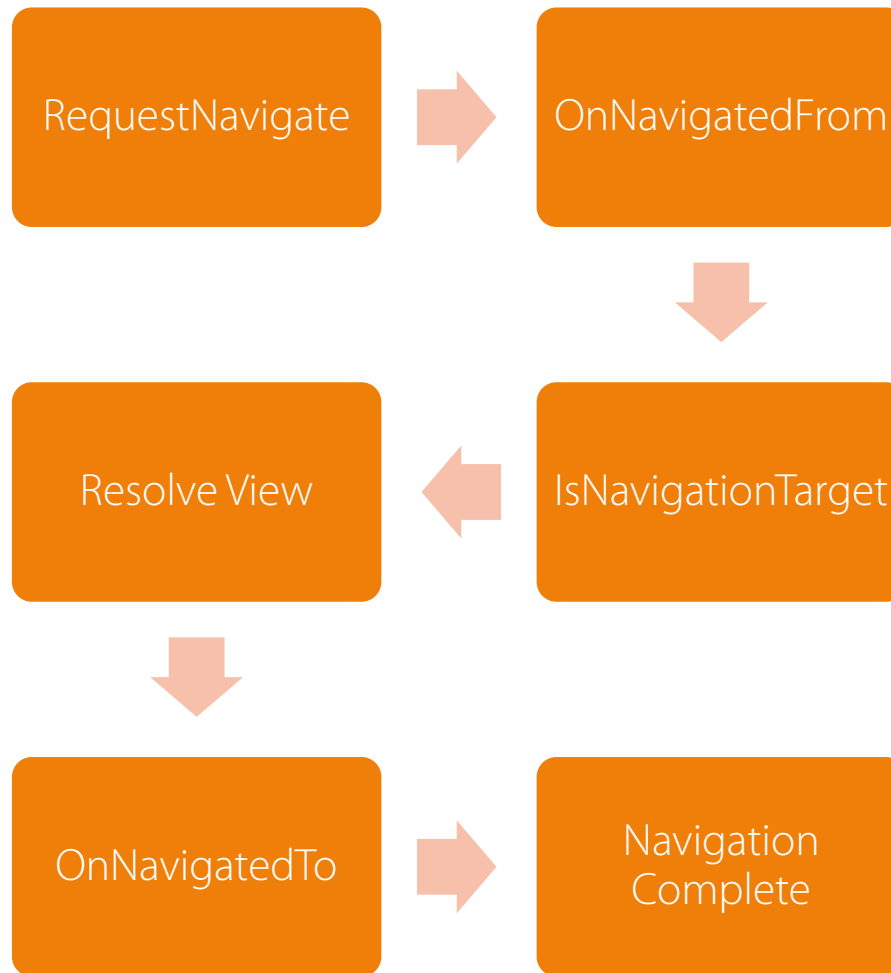


# View and ViewModel Participation

- INavigationAware

```
public interface INavigationAware
{
    bool IsNavigationTarget(NavigationContext navigationContext);
    void OnNavigatedFrom(NavigationContext navigationContext);
    void OnNavigatedTo(NavigationContext navigationContext);
}
```

# INavigationAware Process



# View and ViewModel Participation

- **INavigationAware**

```
public interface INavigationAware
{
    bool IsNavigationTarget(NavigationContext navigationContext);
    void OnNavigatedFrom(NavigationContext navigationContext);
    void OnNavigatedTo(NavigationContext navigationContext);
}
```

- **IRegionMemberLifetime**

```
public interface IRegionMemberLifetime
{
    bool KeepAlive { get; }
}
```

# Passing Parameters

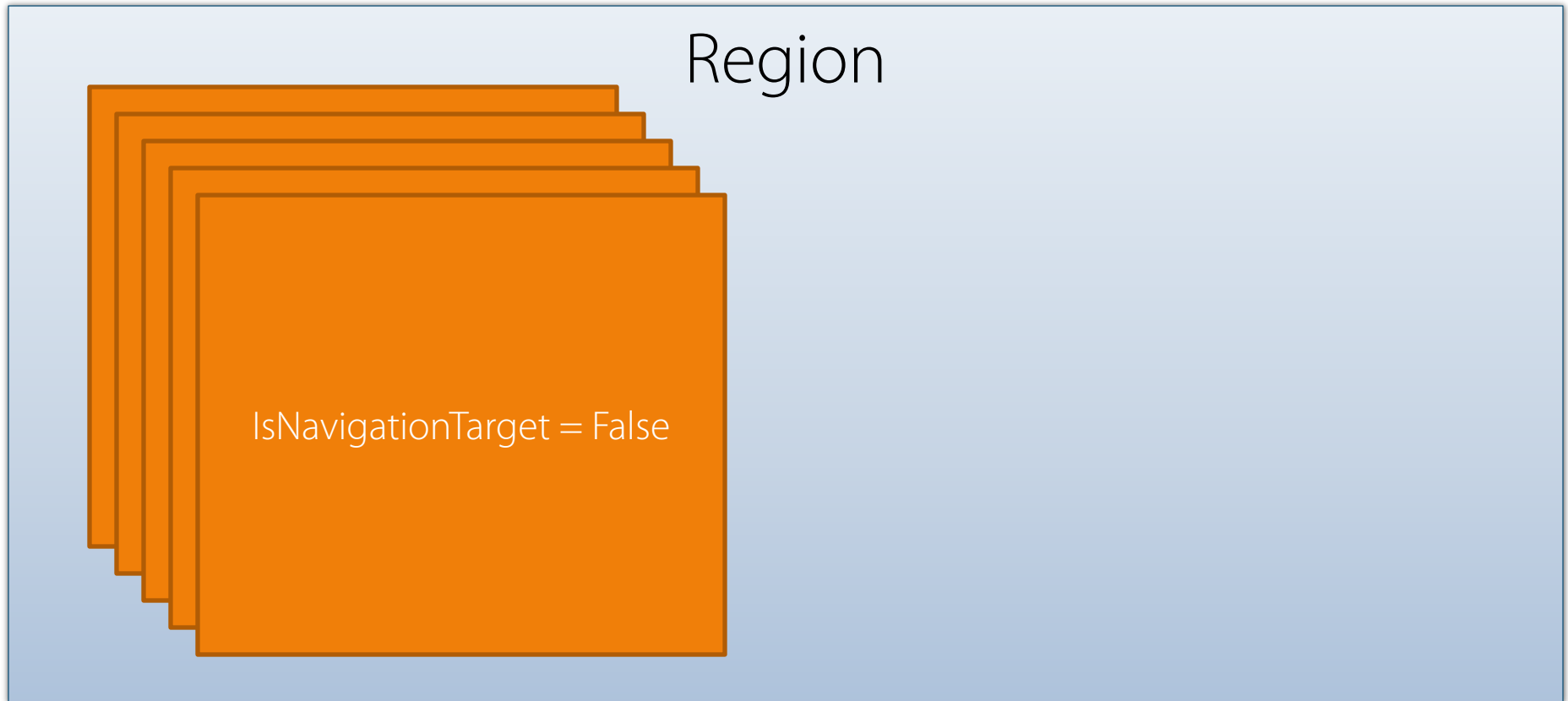
- Append to navigation URI
- UriQuery
- Accessed from INavigationAware methods
- **NavigationContext**
  - NavigationService
  - URI
  - Parameters

# Navigating to Existing Views

- **INavigationAware.IsNavigationTarget**
- **Parameters**

```
public bool IsNavigationTarget(NavigationContext navigationContext)
{
    string id = navigationContext.Parameters["Id"];
    return this.currentCustomer.Id.Equals(id);
}
```

# Navigating to Existing Views

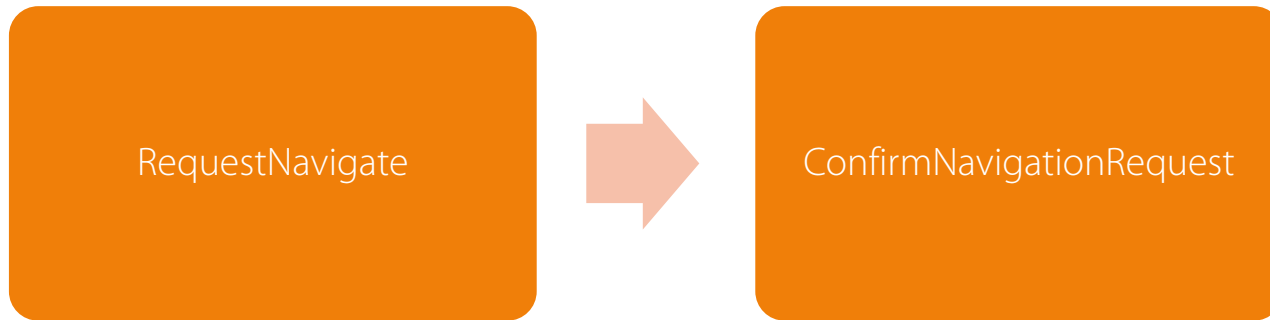


# Confirming or Canceling Navigation

- **IConfirmNavigationRequest**

```
public interface IConfirmNavigationRequest : INavigationAware
{
    void ConfirmNavigationRequest(NavigationContext navigationContext,
        Action<bool> continuationCallback);
}
```

# IConfirmNavigationRequest Process





# IConfirmNavigationRequest Process

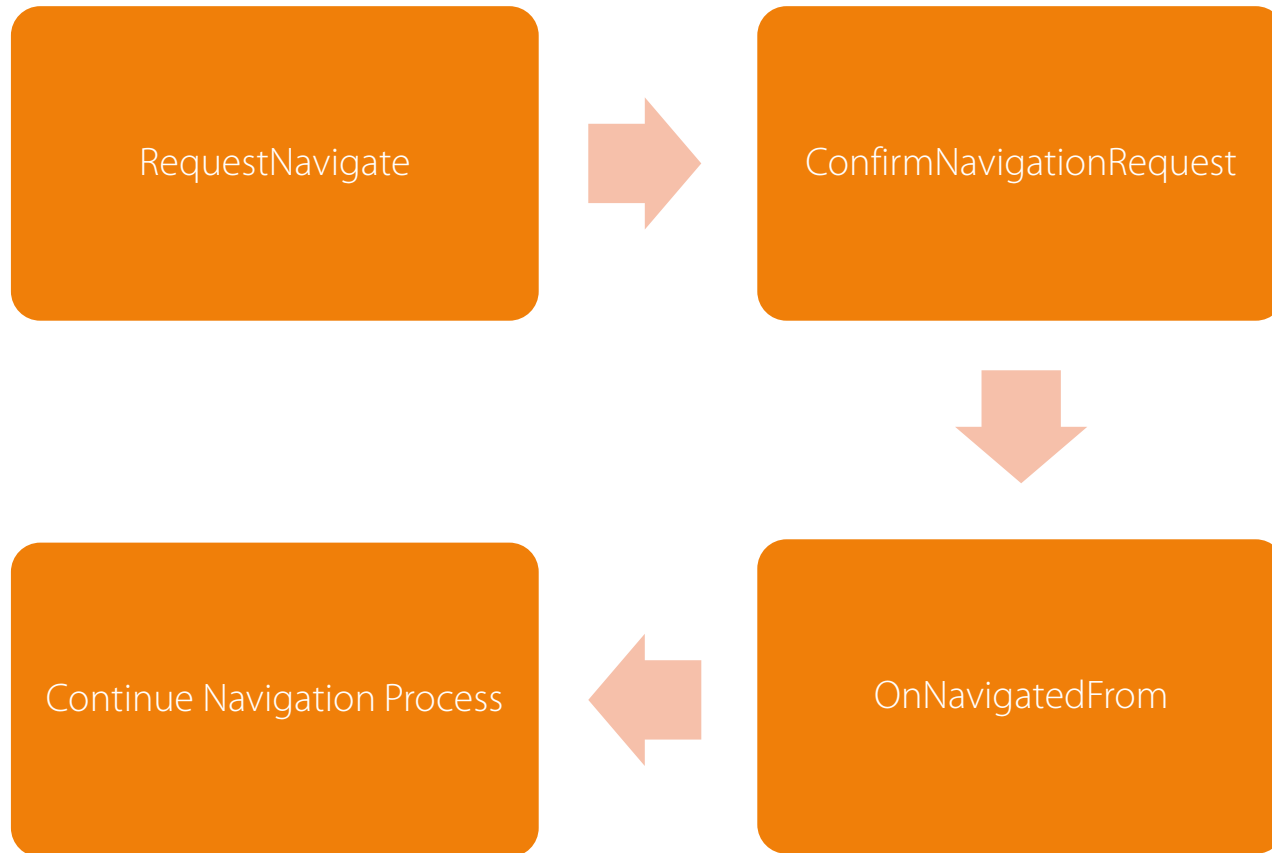
```
public void ConfirmNavigationRequest(NavigationContext navigationContext,
                                    Action<bool> continuationCallback)
{
    //by default we allow navigation
    bool result = true;

    //prompt user for desired action
    MessageBoxResult messageResult = MessageBox.Show("Save", "Save Object?",
                                                    MessageBoxButton.YesNoCancel);

    if (messageResult == MessageBoxResult.Yes)
        CurrentCustomer.Save(); //perform save
    else if (messageResult == MessageBoxResult.Cancel)
        result = false; //don't allow navigation

    continuationCallback(result);
}
```

# IConfirmNavigationRequest Process



# Navigation Journal

- **Stack-Based**
- **Works Only with Region Navigation Services**
- **Accessed from NavigationService.Journal property**
- **NavigationService accessed from NavigationContext**
- **GoBack(), CanGoBack()**
- **GoForward(), CanGoForward()**

# Summary

- Basic Region Navigation
- View and ViewModel Participation
- Passing Parameters
- Navigating to Existing Views
- Confirming or Canceling Navigation
- Navigation Journal

For more in-depth **online** developer **training** visit



**on-demand** content from authors you **trust**