

Using EF Core in Client Applications



Julie Lerman

MOST TRUSTED AUTHORITY ON ENTITY FRAMEWORK

@julielerman thedatafarm.com



Overview



Adding real world logic to the model

EF Core in a client side WPF App

Patterns for EF Core when connected

Tips for many-to-many when connected

EF Core, SQLite and UWP on a phone

Exploring the UWP app's EF Core code



Adding Some Real World Logic





“Please store data modification dates”



Features Added to the Model

New to EF Core

Shadow state properties

Honorable mention:

**EF Core can track via
INotification**

Also added

IsDirty flag (not EF specific)

Ignore properties in model

Override SaveChanges



“Remind users when they
have changes they need to save”

“Let’s use an IsDirty flag”

“Don’t forget! WPF needs to be notified”



EF Core in a Client Side App: Using WPF



Things to Note About WPF & EF Core

**WPF data-binding
is no different
than earlier EF
versions**

**You *can* use
EF Core with
full .NET apps**

**EF Core currently
focused on
working with
.NET Core &
ASP.NET Core**



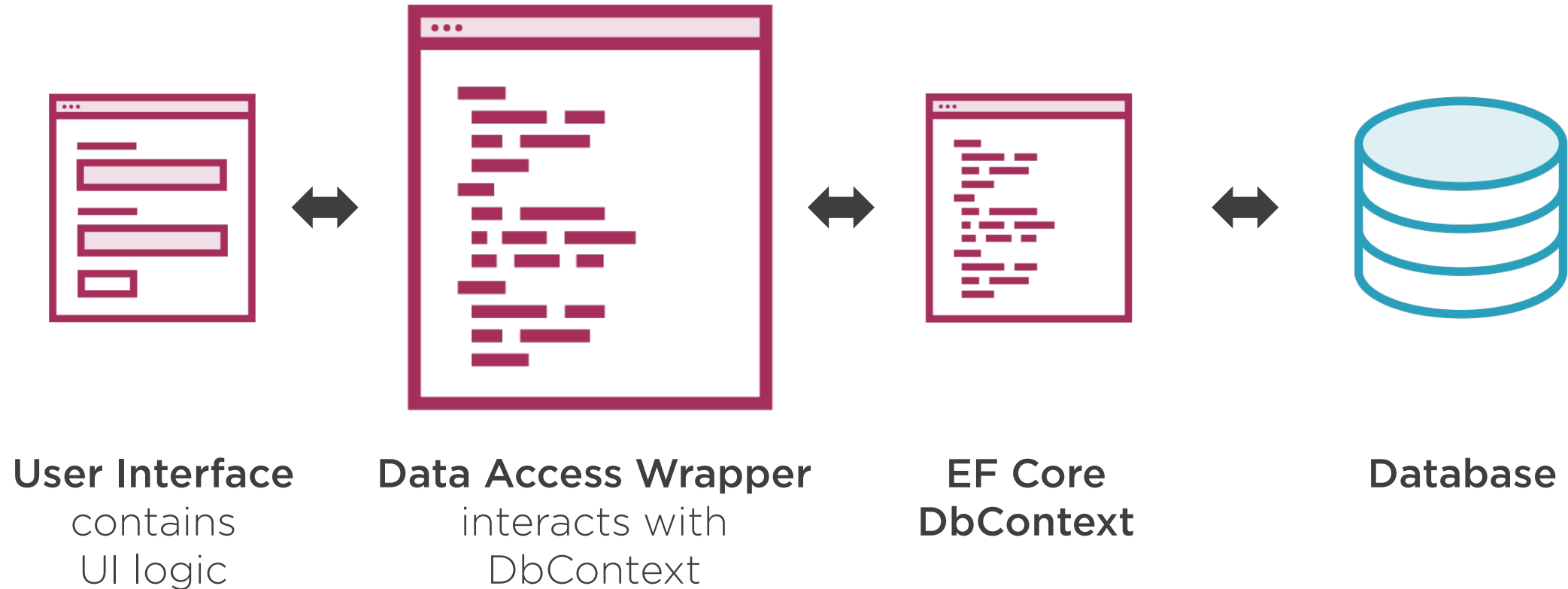
This is not
a step-by-step
walkthrough



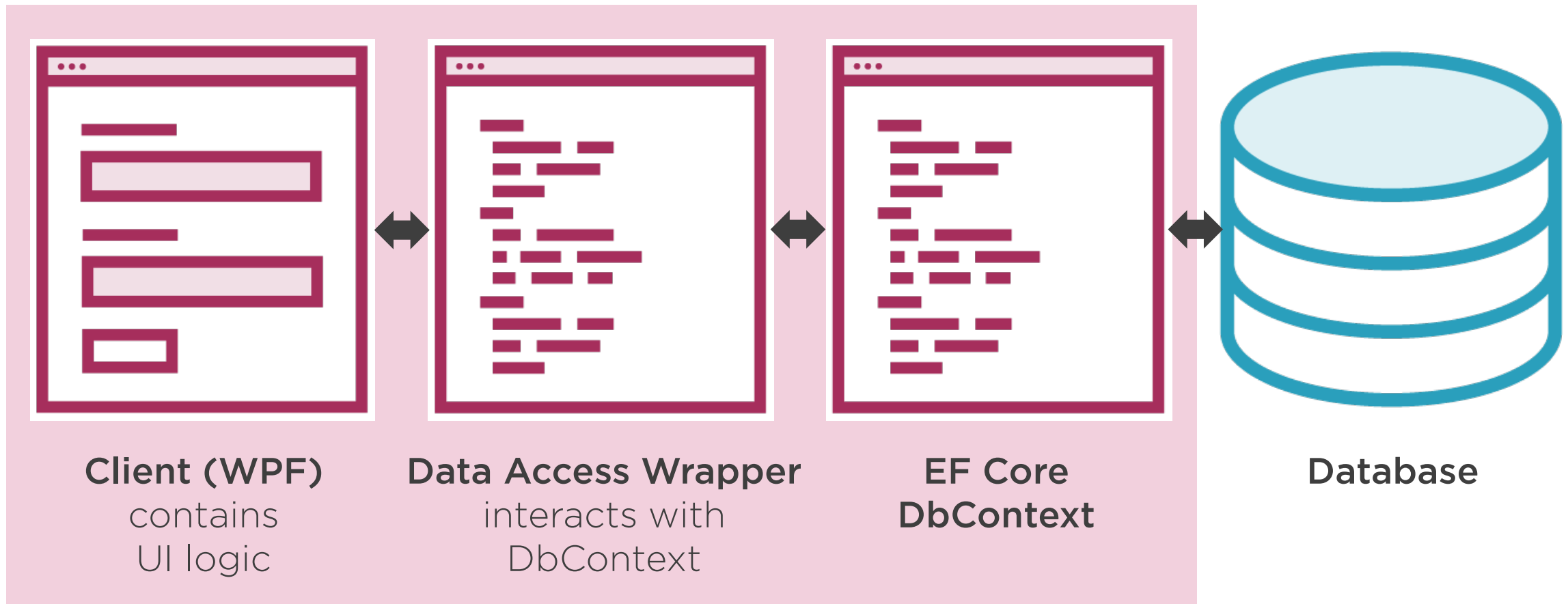
You're here to learn about
EF Core



Data Access Wrapper Contains EF Calls



Connected: DbContext Is Aware of Changes



Patterns for EF Core in Connected Apps



Entity Framework in the Enterprise

← → ↺

https://app.pluralsight.com/library/courses/entity-framework-enterprise-update/table-of-contents

🔍 ☆

Julie

📺 📁 📖 📌 👤 ⋮

PLURALSIGHT

What do you want to learn?

LIBRARY

Home

Browse

Paths

Mentors

Channels

Bookmarks

Notes

Entity Framework in the Enterprise

by Julie Lerman

Entity Framework is the most widely-used ORM for .NET software development. This course will show you how to incorporate Entity Framework into your advanced software architecture.

Resume Course

Bookmark

Add to Channel

Table of contents

Description

Transcript

Exercise files

Discussion

Recommended

Course Overview

Architecting a Data Layer

Understanding EF Encapsulation and the Great Repository Debates

Implementing Encapsulation Patterns with Entity Framework

Managing Complex Domains and Data Models

Expand all

1m 52s

29m 41s

21m 53s

46m 17s

4h 41m

Course author

Julie Lerman

Julie Lerman is a Microsoft MVP, .NET mentor and consultant who lives in the hills of Vermont. You can find Julie presenting on data access and other Microsoft .NET topics at user groups and...

Course info

Level

Intermediate

Rating

★★★★★ (70)

Duration

4h 41m

Released

23 Sep 2016

Share course

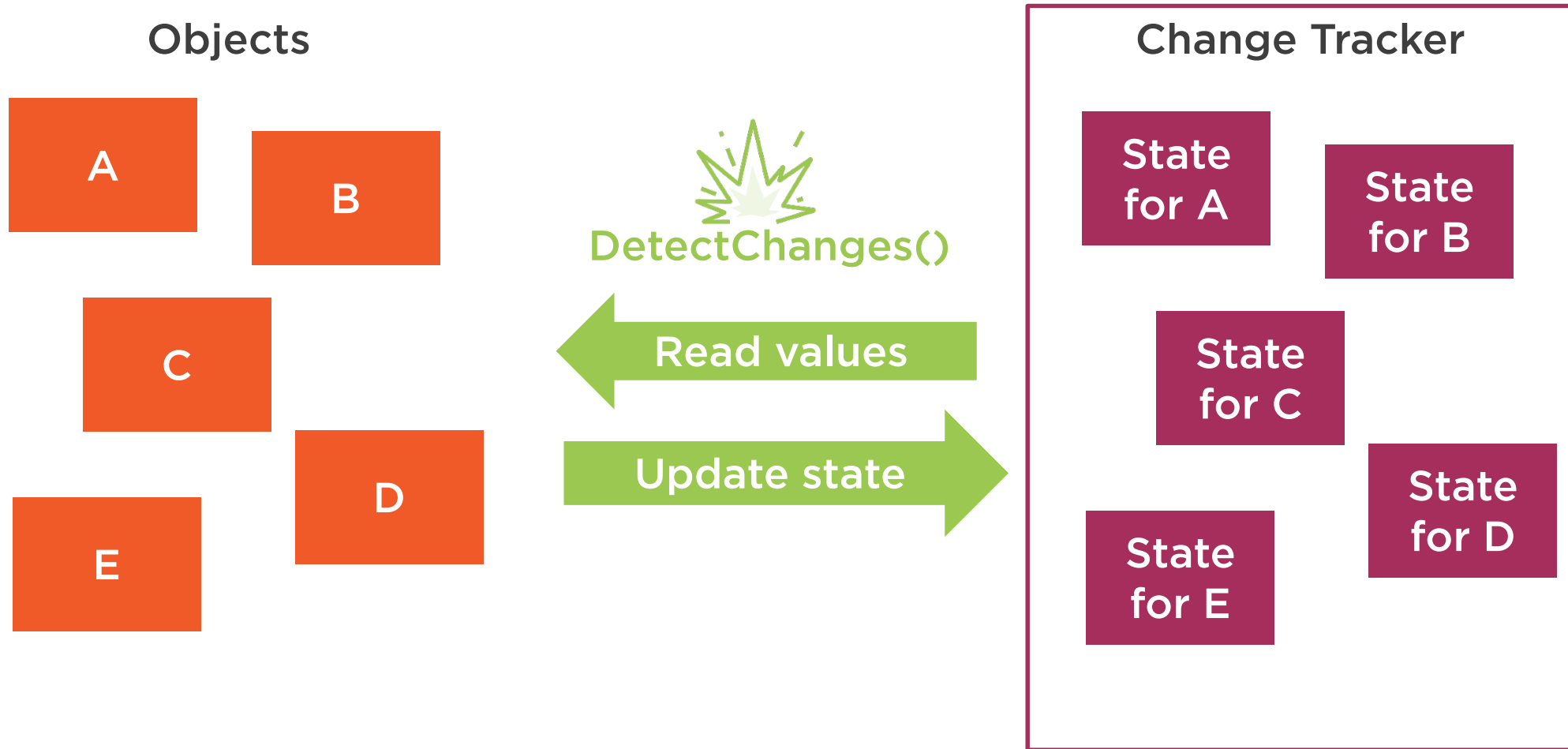
f

🐦

g+

in

DbContext Fixes up State Periodically



Tips for Implementing Many-to-many When Connected



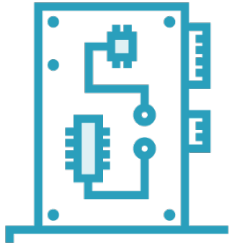
Universal Windows Platform Apps with EF Core



Universal Windows Platform Devices



Persist Data Locally



Devices + IoT



Mobile



PC



Xbox



Surface Hub



Hololens



EF Core



SQLite



Exploring EF Core Usage in the UWP App



Summary

New to EF Core

Shadow state properties

Honorable mention:
EF Core can track via
INotification

Also added

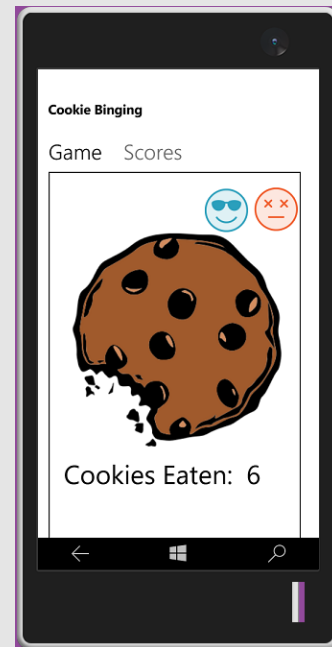
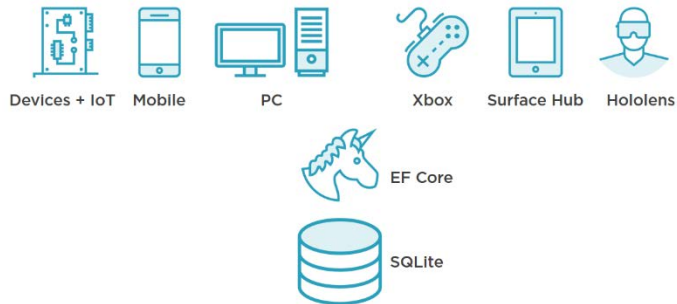
IsDirty flag (not EF specific)

Ignore properties in model
Override SaveChanges

The left screenshot shows a 'MainWindow' with a list of samurais (Julie, Giantpuppy, Kikuchiyo) and a 'Create New Samurai' button. Below the list are input fields for 'Name' (Julie), 'Real Name' (Julia Lerman), and a 'Quotes' text area containing 'I like chocolate!' and 'Hey is there any chocolate here?'. A 'Save' button is at the bottom right.

The right screenshot shows a 'Battles' window with a list of battles (Battle of Shiroyama, Siege of Osaka, Boshin War) and a 'Create New Battle' button. Below the list are input fields for 'Name' (Battle of Shiroyama), 'Start Date' (9/24/1877), and 'End Date' (9/24/1877). There are two lists: 'All Samurais' (Julie, Giantpuppy) and 'Samurais in this Battle' (Kikuchiyo). A 'Save' button is at the bottom right.

Persist Data Locally



```
namespace EFCoreUWP {  
  
    public class BingeContext : DbContext {  
        public DbSet<CookieBinge> Binges { get; set; }  
  
        protected override void OnConfiguring(DbContextOptionsBuilder options) {  
            var databaseFilePath = "CookieBinge.db";  
            try {  
                databaseFilePath = Path.Combine(  
                    ApplicationData.Current.LocalFolder.Path, databaseFilePath);  
            }  
            catch (InvalidOperationException) {  
            }  
  
            options.UseSqlite($"Data source={databaseFilePath}");  
        }  
    }  
}
```



Resources

Entity Framework Core on GitHub github.com/aspnet/entityframework

EF Core Documentation docs.microsoft.com/ef

Arthur Vickers Notification Entities Post:

blog.oneunicorn.com/2016/11/16/notification-entities-in-ef-core-1-1

Data Points - EF Core Change-Tracking Behavior: Unchanged, Modified and Added:

msdn.microsoft.com/magazine/mt767693

Guidance on Storing Data in Windows 10 Devices

blogs.windows.com/buildingapps/2016/05/10/getting-started-storing-app-data-locally

EnsureCreated vs Migrate Methods in EF Core:

thedatafarm.com/data-access/ef7-ensurecreated-vs-migrate-methods

Download Windows 10 SDK developer.microsoft.com/en-us/windows/downloads/windows-10-sdk



Using EF Core in Client Applications



Julie Lerman

MOST TRUSTED AUTHORITY ON ENTITY FRAMEWORK

@julielerman thedatafarm.com

