

Exploring Mappings and Migrations



Julie Lerman

MOST TRUSTED AUTHORITY ON ENTITY FRAMEWORK

@julielerman thedatafarm.com



Overview



Understanding many-to-many in EF Core

Creating a many-to-many relationship

Creating a one-to-one relationship

Scripting and migrating the database
after the model changes

Scaffolding a model from an existing
database



Let's Fix That, and Add Something Else, Too



Many (Samurais) to Many (Battles)



One (Samurai) to One (SecretIdentity)



Changes for EF Core

.

**is not supported
in EF Core 1.0 or 1.1
*without a join entity***

1:1

**got easier
in EF Core
than previous EF**

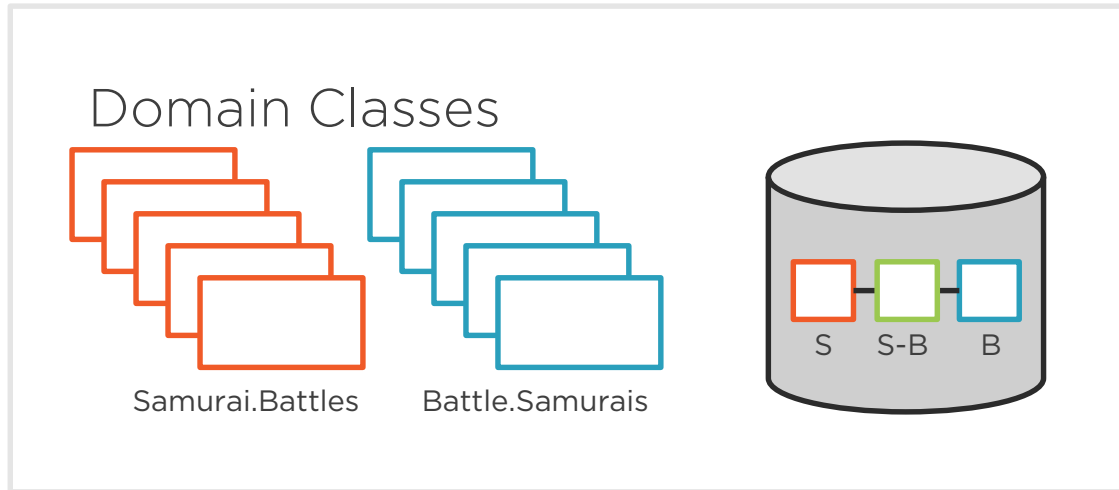
github.com/aspnet/EntityFramework/issues/1368



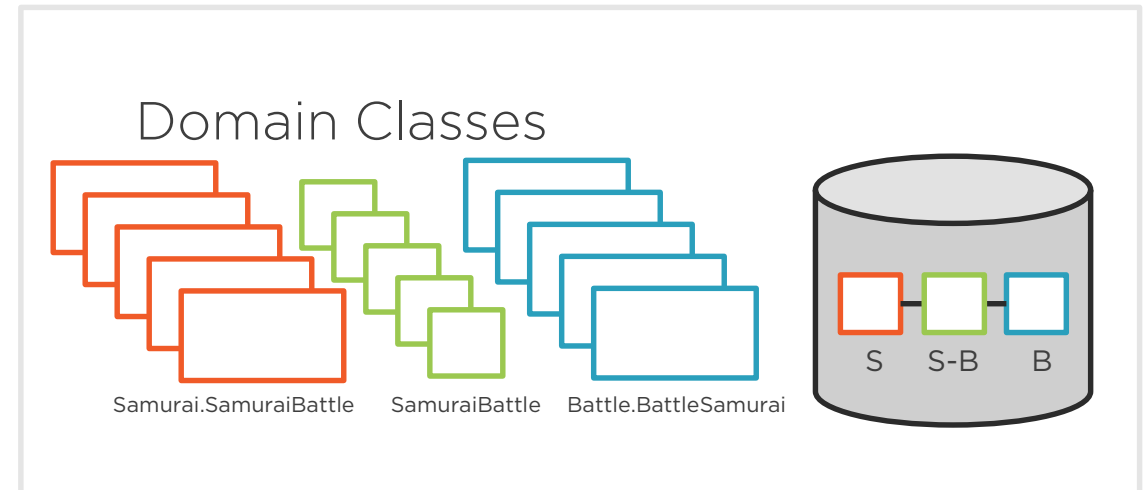
Understanding Many-to-Many Relationships



Many-to-Many Through EF6

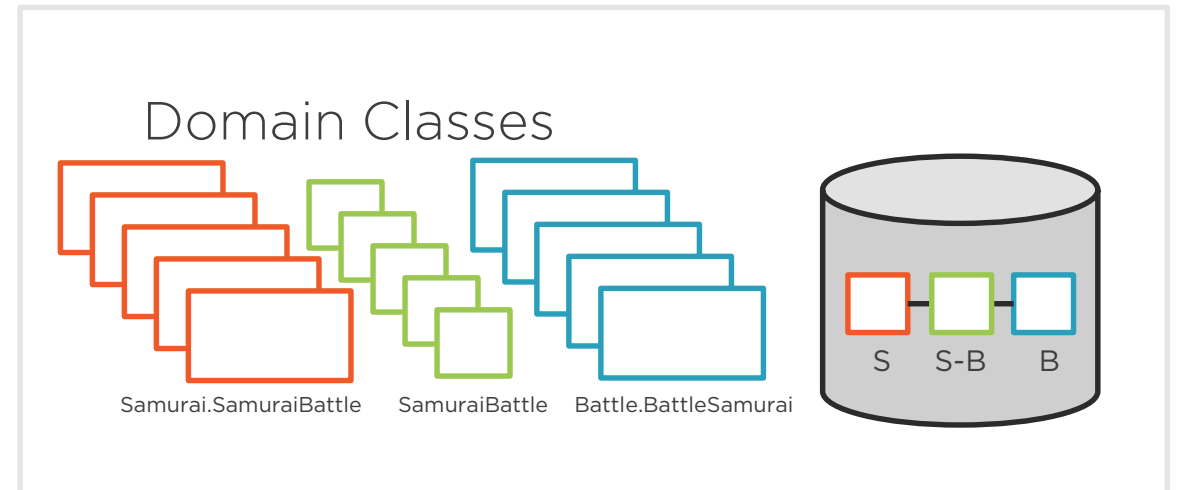


EF can infer the database **join** table
EF performs magic at runtime



You define a **join** entity
You control relationship in code

Many-to-Many in EF Core 1.0 & 1.1



You define a join entity
You control relationship in code



Creating a Many-to-Many Relationship



Primary Keys

```
public class Samurai
{
    public int Id { get; set; }
}
```

◀ Property named Id

```
public class Samurai
{
    public int SamuraiId { get; set; }
}
```

◀ Property named [type]Id



Overriding EF's Default Mapping Conventions

Data Annotations in class

```
[Key]
public int SamuraiKey { get; set; }

[Column("Spoken_Quote")]
public string Text { get; set; }
```

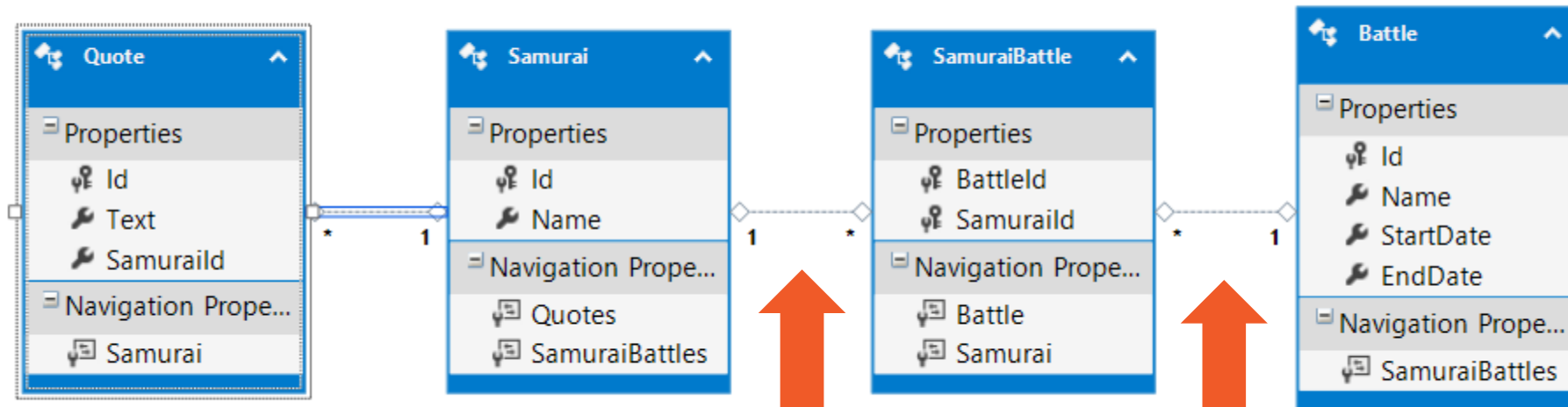
Fluent API via DbContext

```
modelBuilder.Entity<Samurai>()
    .HasKey(s => s.SamuraiKey);

modelBuilder.Entity<Quote>()
    .Property(q=>q.Text)
    .HasColumnName("Spoken_Quote");
```



The Data Model with Its Join Entity



One Samurai to Many SamuraiBattle

One Battle to Many SamuraiBattle



Creating a One-to-one Relationship



One-to-one Relationship Rules

Through EF6

Requires navigation properties on both ends

Primary key of dependent is also FK to principal

Mark dependent navigation as required, otherwise 1:0..1

Or: Fluent API to specify *principal* and *dependent*

In EF Core

Requires navigation properties on both ends

FK property used to infer *principal* and *dependent**

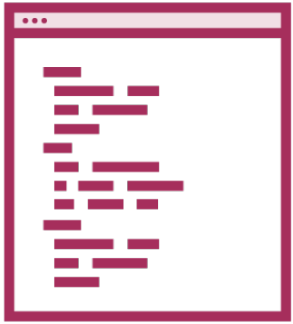
Mark dependent navigation as required, otherwise 1:0..1

Or: Simpler Fluent API: HasOne, WithOne

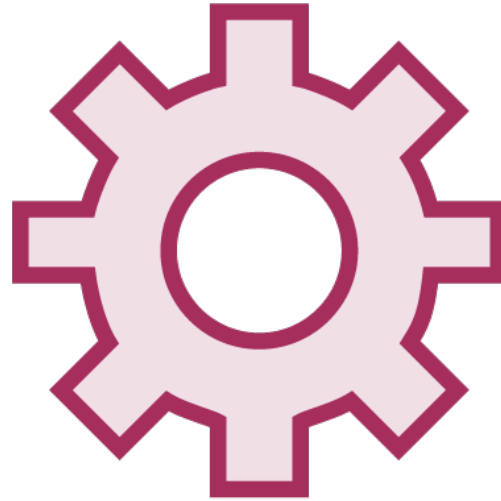
* Unique Constraint/FK support introduced in EF Core



Mappings Are for EF and Database



Business Logic



EF Core Logic



Database Constraints

Scripting and Migrating the Database After the Model Changes



Remove-Migrations Command

Does:

- Delete latest migration
- Updates snapshot

Does Not:

- Undo migration in database
- Modify your model's code
(domain classes, DbContext)



Remove Migrations with dotnet CLI

```
dotnet ef migrations remove
```

```
dotnet ef migrations remove --startup-project ../CoreUI
```



Understanding EF Core Migrations' Scaffolding Command



Scaffold with dotnet CLI

```
dotnet ef dbcontext scaffold "connectionstring" provider
```



```
--startup-project ../CoreUI
```



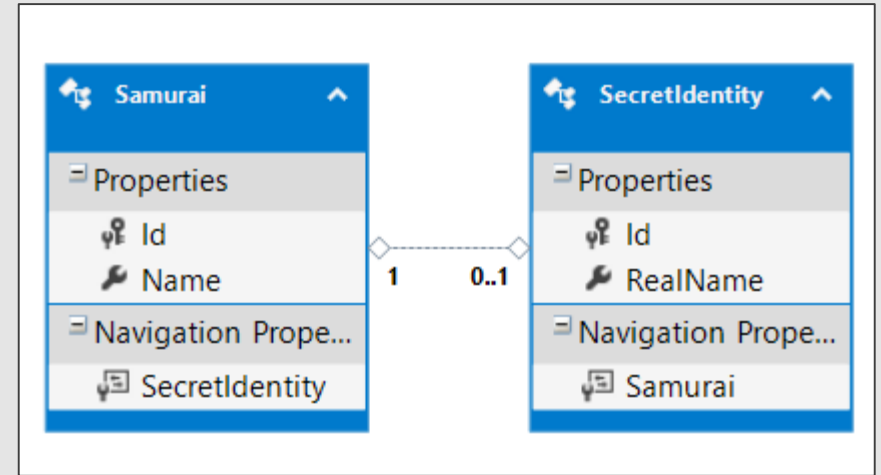
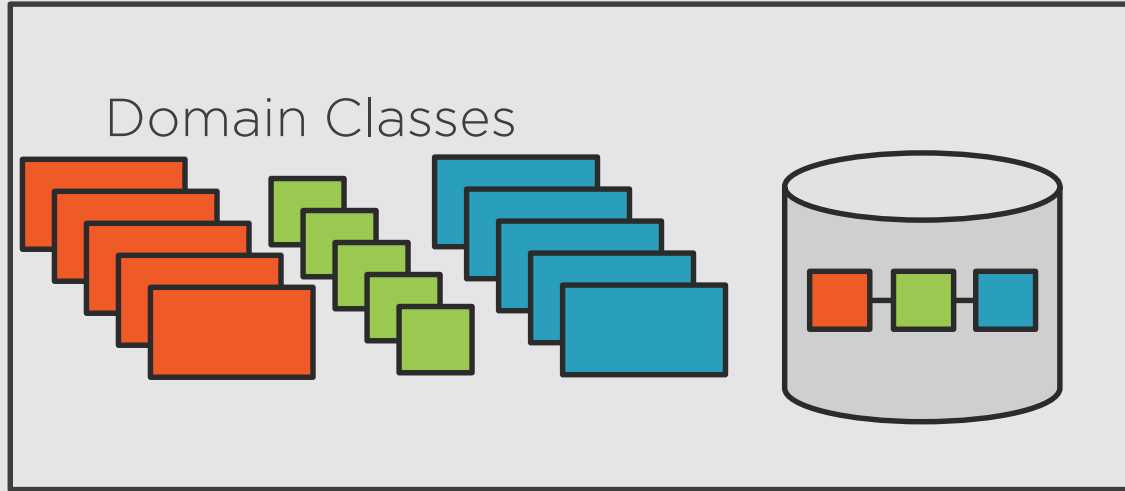
Scaffolding a Model from an Existing Database



Scaffolding is a one-time
function to provide a
stake in the ground



Review



Package Manager Console

Package source: **nuget.org**

```
PM> script-migration
PM> script-migration -idempotent
PM> script-migration -from JoinTable
PM> script-migration -from initial
```

Package Manager Console

Package source: **nuget.org** | Default project: **src\ScaffoldFromDatabase**

NAME

Scaffold-DbContext

SYNOPSIS

Scaffolds a DbContext and entity types for a database.

SYNTAX

```
Scaffold-DbContext [-Connection] <String> [-Provider] <String> [-OutputDir <String>] [-Context <String>] [-Schemas <String[]>] [-Tables <String[]>] [-DataAnnotations] [-Force] [-Environment <String>] [-Project <String>] [-StartupProject <String>] [<CommonParameters>]
```



Resources

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

EF Core Roadmap bit.ly/efcoreroadmap

EF Core Documentation docs.efproject.net

Run EF Core on Both .NET Framework and .NET Core bit.ly/2dw4Vab
MSDN Magazine, Data Points Oct 2016, Julie Lerman

EF7 Migrations: Not New but Definitely Improved bit.ly/1KEfMLF
MSDN Magazine, Data Points Jan 2016, Julie Lerman



Exploring Mappings and Migrations



Julie Lerman

MOST TRUSTED AUTHORITY ON ENTITY FRAMEWORK

@julielerman thedatafarm.com

