

Sponsors and Partners

Strategic Sponsors



Gold Sponsors



Silver Sponsors





.NET

DEVELOPER

DAYS

Entity Framework Core

Tomas Herceg

Microsoft MVP
Microsoft Regional Director
CEO of RIGANTI

Entity Framework Core

- Lightweight
- Extensible
- Cross Platform
 - Full .NET (Windows Only)
 - .NET Core (All Platforms)
- Renamed from „Entity Framework 7“
- Revolution instead of evolution

Why New Version?

- The codebase of EF was pretty old
 - And not easily portable
- Two APIs
 - DbContext,ObjectContext
- Monolithic design
 - Difficult to extend (e.g. Hierarchy ID support)
- High memory footprint
 - Problem on mobile devices

New Features

- Batching
 - INSERT, UPDATE & DELETE done in batches
- Alternate Keys
 - Can be used as one end of the relationship
- More Complex Querying Mechanisms
 - In EF6, the whole LINQ query must be translated to SQL
 - In EF Core, the provider can decide what will be translated and what cannot
- Non-relational Data Store Support
 - Azure Table Storage
 - Redis
- In-Memory Store for Testing

Client Evaluation

```
from p in db.Products
where p.Discontinued == false
select new {
    Name = p.Name,
    ItemsOnStock = GetDisplayText(p.ItemsOnStock)
}
```

- EF Core can evaluate the **GetDisplayText** on the client
- Careful: It can slow down the query! Client eval produces warnings.
- <https://docs.efproject.net/en/latest/querying/client-eval.html>

Code First

- The name was not chosen right
- „Code-Based Modeling“
- The model can be written from scratch, or generated from database
- The only supported way for Entity Framework Core
 - No EDMX
 - No designer in Visual Studio
 - Console scaffolding tools only
 - `dotnet ef dbcontext scaffold [connectionString] [provider]`

What's Missing

- Dropped
 - OldObjectContext API
- Not Implemented Yet
 - Inheritance types (TPT, TPC)
 - Many to many without join entity
 - Complex types
 - Spatial data
 - Lazy Loading, Explicit Loading
 - Stored Procedures

Things That Remain The Same

- LINQ
- DbSet<TEntity>
- DbSet.Add
- DbSet.Remove
- DbContext
- DbContext.Database
- DbContext.ChangeTracker

```
using (var db = new BlogContext())
{
    db.Blogs.Add(new Blog
    {
        Url = "...";
    });
    db.SaveChanges();

    var blogs = from b in db.Blogs
                .Include(b => b.Posts)
                orderby b.Name
                select b;

    foreach (var blog in blogs)
    {
        Console.WriteLine(blog.Name);
        foreach (var post in blog.Posts)
        {
            Console.WriteLine(post.Title);
        }
    }
}
```



.NET

DEVELOPER

DAYS

DEMO

Entity Framework Core

Database Schema Changes

- EDMX: Update Model From Database
 - Not very reliable
 - Sometimes, your manual changes to the model just disappear
- Code First Migrations in EF6
 - Problems in team environment
 - Merge conflicts
- EF Core Migrations
 - `dotnet ef migrations add {name}`
 - `dotnet ef database update`



.NET

DEVELOPER

DAYS

DEMO

Entity Framework Core – Migrations

Providers

- Microsoft SQL Server
- SQLite
- PostgreSQL
- MySQL (official and Pomelo)
- Microsoft SQL Server Compact
- IBM Data Servers
- InMemory
- Devart (MySQL, Oracle, PostgreSQL, SQLite, DB2, SQL Server and more)
- Oracle (Coming Soon)



.NET

DEVELOPER

DAYS

DEMO

Entity Framework Core on Linux

EF Core 1.1

- Improved LINQ translation
- DbSet.Find
- Explicit Loading
- EntityEntry API: Reload, GetModifiedProperties, GetDatabaseValues
- Connection Resiliency
- Pluralization
- RTM tooling

- Also, Entity Framework 6.2 is coming

To upgrade or not to upgrade?

- New platforms (UWP, .NET Core on Linux)
 - Try it
- Existing applications with EF6
 - No need to upgrade
- Review the feature comparison
 - <https://docs.efproject.net/en/latest/efcore-vs-ef6/features.html>

Sponsors and Partners

Strategic Sponsors



Gold Sponsors



Silver Sponsors

