Better Object Mapping in .NET with Dapper

Kevin Griffin
@1kevgriff

JetBrains .NET Days Online

Hello, I'm Kevin!

Kevin Griffin
Consultant

10-time Microsoft MVP
Focused on ASP.NET / ASP.NET Core / Azure

twitter.com/1kevgriff
kevin@consultwithgriff.com

consultwithgriff.com

twitch.tv/1kevgriff

Where are the slides?

Swift Kick

https://consultwithgriff.com/dapper

What is Object Mapping?

Swift Kick

Results Messages

	Id	FirstName	LastName	PhoneNumber	UserName	DateOfBirth	City	State	ZipCode
1	1	Stefanie	Buckley	681293-8600	LOMZAPEPOWER\Karl.Gallegos	1971-08-12	Montgomery	Maryland	13285
2	2	Sandy	Mc Gee	187-5405534	REDUDEWAX\Jaime.Barron	1968-08-23	St. Paul	Utah	49217
3	3	Lee	Warren	2505485864	SUPHUPEDANTOR\Gloria.Neal	1967-06-19	Bakersfield	North Dakota	65029
4	4	Regina	Forbes	3540801721	HAPDIMOMMOR\Rosa.Mc Neil	1987-05-25	Yonkers	Arizona	48838
5	5	Daniel	Kim	759-363-6906	BARFROPEDGANTOR\Stefanie.Robertson	1978-06-29	Glendale	Maine	68675
6	6	Dennis	Nunez	7850174429	EMERIMAN\Morgan.Ayers	2004-01-23	Des Moines	North Dakota	55141
7	7	Myra	Zuniga	727995-7122	TIPMUNICEX\Geoffrey.Herring	1961-01-18	Anaheim	Arizona	87469
8	8	Teddy	Ingram	074-316-5818	LOMMUNINOWER\Johanna.Paul	2005-04-13	Cincinnati	West Virginia	11805
9	9	Annie	Larson	933-4217334	TIPROBILANTOR\Shanda.Wilkinson	1985-10-15	Aurora	Texas	23341
10	10	Hernan	Anderson	885791-5573	ENDGLIBAN\Morgan.Conrad	1959-06-18	Omaha	Texas	08571
11	11	Jenifer	Livingston	186-3875584	RAPFROPOMMAR\Stephanie.Peck	2006-08-28	Akron	Minnesota	86406
12	12	Stefanie	Perez	9827245231	PARDIMIMOR\Latanya.Matthews	1973-05-26	Jacksonville	Washington	91433
13	13	Chastity	Garcia	957-869-0090	EMBANIMEX\Victor.Boyd	1956-11-05	Richmond	Minnesota	37911
14	14	Evelyn	Stokes	1167318493	RAPPICKOMMIN\John.Monroe	1956-11-22	Louisville	Colorado	66274
15	15	Jeannie	Daniel	146-3882806	RAPQUESTEFENTOR\Thomas.Pierce	1979-07-06	Memphis	Vermont	81333
16	16	Rickey	Santos	633-8555322	VARSAPEPISTOR\Oscar.Hopkins	1961-05-31	Jackson	Massachus...	10578

```
public class User
{
    public long Id { get; set; }
    public string FirstName { get; set; }
    public string LastName { get; set; }
    public string PhoneNumber { get; set; }
    public string UserName { get; set; }
    public DateTime DateOfBirth { get; set; }
    public string City { get; set; }
    public string State { get; set; }
    public string ZipCode { get; set; }
}
```

What is Object Mapping?

Swift Kick

Because raw database results are f-ing useless

How do we do object mapping?

Swift Kick

- DataReader -> Object
- DataTable
- Entity Framework
- nHibernate
- Massive / PetaPoco / etc.

Dapper

Swift Kick

Dapper - a simple object mapper for .Net

build passing

Release Notes

Located at stackexchange.github.io/Dapper

Packages

MyGet Pre-release feed: https://www.myget.org/gallery/dapper

Package	NuGet Stable	NuGet Pre-release	Downloads	MyGet
Dapper	nuget v2.0.30	nuget v2.0.30	downloads 29M	dapper v2.0.32
Dapper.Contrib	nuget v2.0.30	nuget v2.0.30	downloads 2M	dapper v2.0.32
Dapper.EntityFramework	nuget v2.0.30	nuget v2.0.30	downloads 32k	dapper v2.0.32
Dapper.EntityFramework.StrongName	nuget v2.0.30	nuget v2.0.30	downloads 3.9k	dapper v2.0.32
Dapper.Rainbow	nuget v2.0.30	nuget v2.0.30	downloads 31k	dapper v2.0.32
Dapper.SqlBuilder	nuget v2.0.30	nuget v2.0.30	downloads 519k	dapper v2.0.32
Dapper.StrongName	nuget v2.0.30	nuget v2.0.30	downloads 616k	dapper v2.0.32

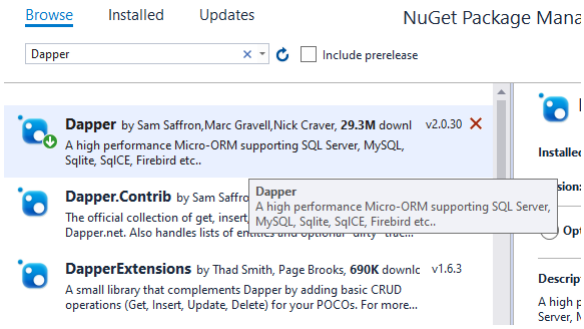
Dapper

Swift Kick

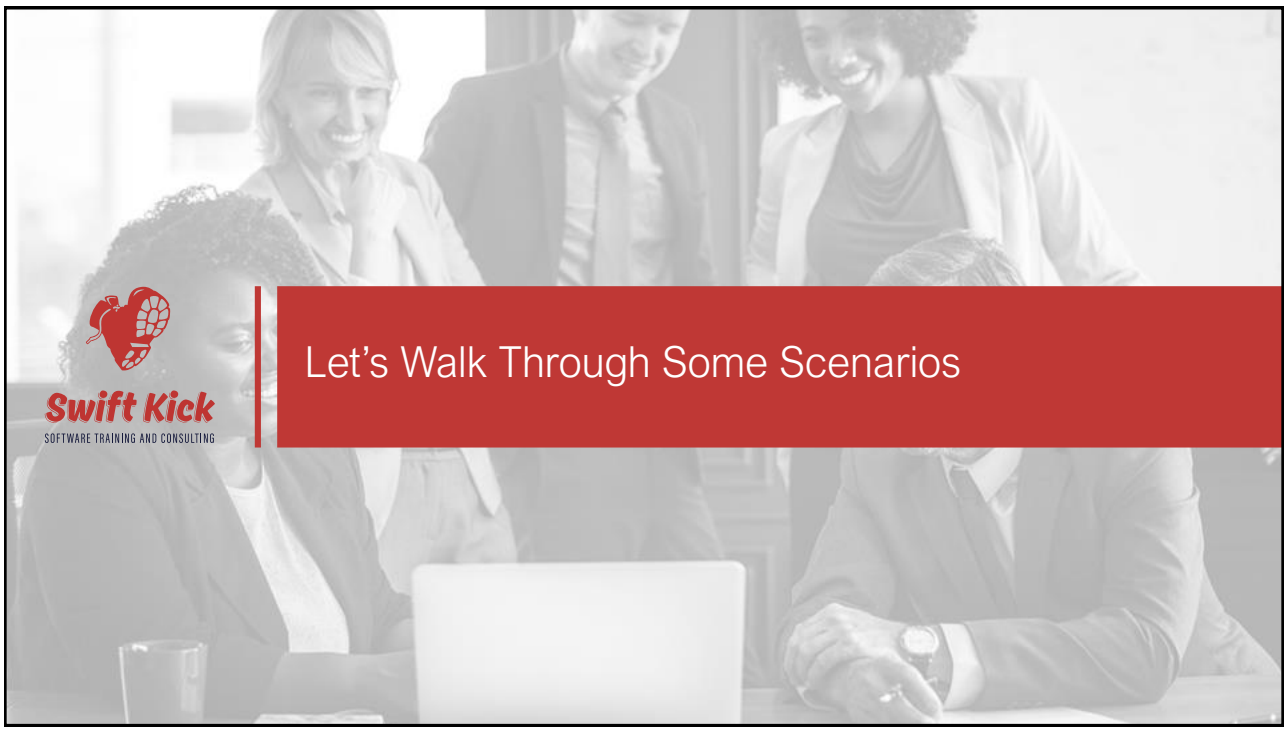
- "Micro" ORM
- No SQL generation (ala Entity Framework), but you're better than that anyway
- No Database generation (ala Entity Framework), but again – better this way.
- Provides fluent interface for mapping DataReaders to objects
- No DB specific implementation
  - Works across SQL Server, SQLite, SQL CE, Firebird, Oracle, MySQL, and PostgresSQL
- Developed by the great folks at StackOverflow!

How it works

Swift Kick




```
await using var connection = new SqlConnection(sqlConnectionString);
var sql = @"SELECT [Id]
                  ,[FirstName]
                  ,[LastName]
                  ,[PhoneNumber]
                  ,[UserName]
                  ,[DateOfBirth]
                  ,[City]
                  ,[State]
                  ,[ZipCode]
              FROM [dbo].[MassiveUserList]";
var result = await connection.QueryAsync<User>(sql);
```

QueryAsync<>

Swift Kick

```
private static async Task DemoWithQueryAsync(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"SELECT [Id]
                      ,[FirstName]
                      ,[LastName]
                      ,[PhoneNumber]
                      ,[UserName]
                      ,[DateOfBirth]
                      ,[City]
                      ,[State]
                      ,[ZipCode]
                  FROM [dbo].[MassiveUserList]";
    var result = await connection.QueryAsync<User>(sql);
    WriteList(result);
}
```


QueryAsync<> with parameters

Swift Kick

```
private static async Task DemoWithQueryAsyncWithParameters(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"SELECT [Id]
                      ,[FirstName]
                      ,[LastName]
                      ,[PhoneNumber]
                      ,[UserName]
                      ,[DateOfBirth]
                      ,[City]
                      ,[State]
                      ,[ZipCode]
                  FROM [dbo].[MassiveUserList]
                  WHERE DateOfBirth > @dateOfBirth";

    var result = await connection.QueryAsync<User>(sql,
        param: new { dateOfBirth = new DateTime( year: 2005,  month: 10,  day: 01) });
    WriteList(result);
}
```

QueryAsync<> with Dynamic Parameters


```
private static async Task DemoWithQueryAsyncWithDynamicParameters(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"SELECT [Id]
                      ,[FirstName]
                      ,[LastName]
                      ,[PhoneNumber]
                      ,[UserName]
                      ,[DateOfBirth]
                      ,[City]
                      ,[State]
                      ,[ZipCode]
                  FROM [dbo].[MassiveUserList]
                  WHERE DateOfBirth > @dateOfBirth";

    var dynamicParam = new DynamicParameters();
    dynamicParam.Add(name: "dateOfBirth", new DateTime(year: 2005, month: 10, day: 01));

    var result = await connection.QueryAsync<User>(sql, dynamicParam);

    WriteList(result);
}
```


QueryFirst or QueryFirstOrDefault


```
private static async Task DemoWithQueryFirst(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"SELECT TOP 1 [Id]
                      ,[FirstName]
                      ,[LastName]
                      ,[PhoneNumber]
                      ,[UserName]
                      ,[DateOfBirth]
                      ,[City]
                      ,[State]
                      ,[ZipCode]
                  FROM [dbo].[MassiveUserList]";

    var result = await connection.QueryFirstAsync<User>(sql);
    WriteList(result);
}
```

ExecuteAsync for INSERT/UPDATE/DELETE


```
private static async Task DemoExecuteAsyncWithInsert(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"INSERT INTO [dbo].[MassiveUserList]
                       ([FirstName]
                       ,[LastName]
                       ,[PhoneNumber]
                       ,[UserName]
                       ,[DateOfBirth]
                       ,[City]
                       ,[State]
                       ,[ZipCode])
                 VALUES
                       (@firstName, @lastName, @phoneNumber,
                        @userName, @dateOfBirth,
                        @city, @state, @zipCode)";

    var newUser = new User()
    {
        FirstName = "Jim", LastName = "Bob", City = "Chesapeake", State = "Virginia", ZipCode = "23320",
        DateOfBirth = DateTime.Parse("1999-01-01"),
        PhoneNumber = "757-867-5309", UserName = "COMPUTER/JimBob"
    };

    var result = await connection.ExecuteAsync(sql, newUser);
}
```


ExecuteAsync for INSERT/UPDATE/DELETE


```
private static async Task DemoExecuteAsyncWithUpdate(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"UPDATE [dbo].[MassiveUserList]
                SET DateOfBirth = @dateOfBirth
                WHERE FirstName = @firstName";

    var result = await connection.ExecuteAsync(sql,
        param: new {dateOfBirth = DateTime.Parse("2009-01-01"), firstName = "Jim"});
}
```

ExecuteAsync for INSERT/UPDATE/DELETE


```
private static async Task DemoExecuteAsyncWithDelete(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"DELETE FROM [dbo].[MassiveUserList]
                WHERE LastName = @lastName";

    var result = await connection.ExecuteAsync(sql, param: new {lastName = "Bob"});
}
```


Stored Procedures


```
private static async Task DemoStoredProcedure(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = "dbo.LastNameList";

    var results = await connection.QueryAsync<string>(sql, commandType: CommandType.StoredProcedure);
}
```

Transactions

- Do this one live – it's a ton of code.

Dynamic Results


```
private static async Task DemoDynamicResults(string sqlConnectionString)
{
    await using var connection = new SqlConnection(sqlConnectionString);
    var sql = @"SELECT [Id]
                      ,[FirstName]
                      ,[LastName]
                      ,[PhoneNumber]
                      ,[UserName]
                      ,[DateOfBirth]
                      ,[City]
                      ,[State]
                      ,[ZipCode]
                  FROM [dbo].[MassiveUserList]";

    var result = await connection.QueryAsync(sql);
    WriteList(result);
}
```

Thanks for joining me!

Swift Kick

Kevin Griffin
Owner, Swift Kick

twitter.com/1kevgriff
kevin@swiftkick.in

kevgriffin.com
swiftkick.in