

Why Entity Framework?

Overview of ADO.NET Entity Framework



Overview

- **Microsoft Moves Forward with Data Access in .NET**
- **Entity Framework and Your Application**
- **Coding 101 with Entity Framework**

Why Entity Framework?

ADO
Record
Set

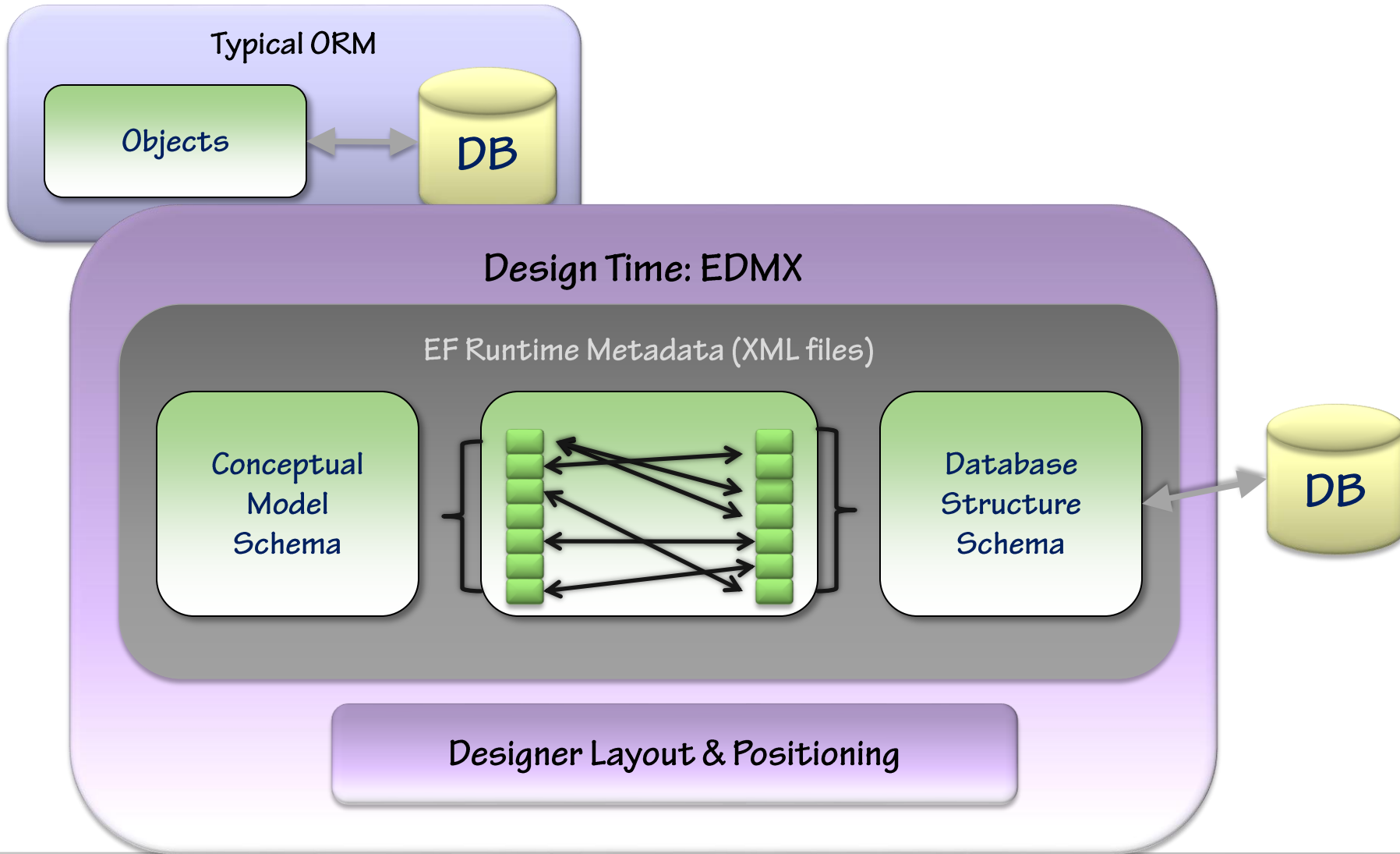
ADO.NET
Data
Set

Object Relational Mapper

Object
Spaces

Entity
Framework

EF Metadata & Other ORMs



ADO.NET Today & Tomorrow



DataTables



LINQ to SQL



Entity Framework

Model in Entity Framework

Database
First

Model
First

Legacy
Database

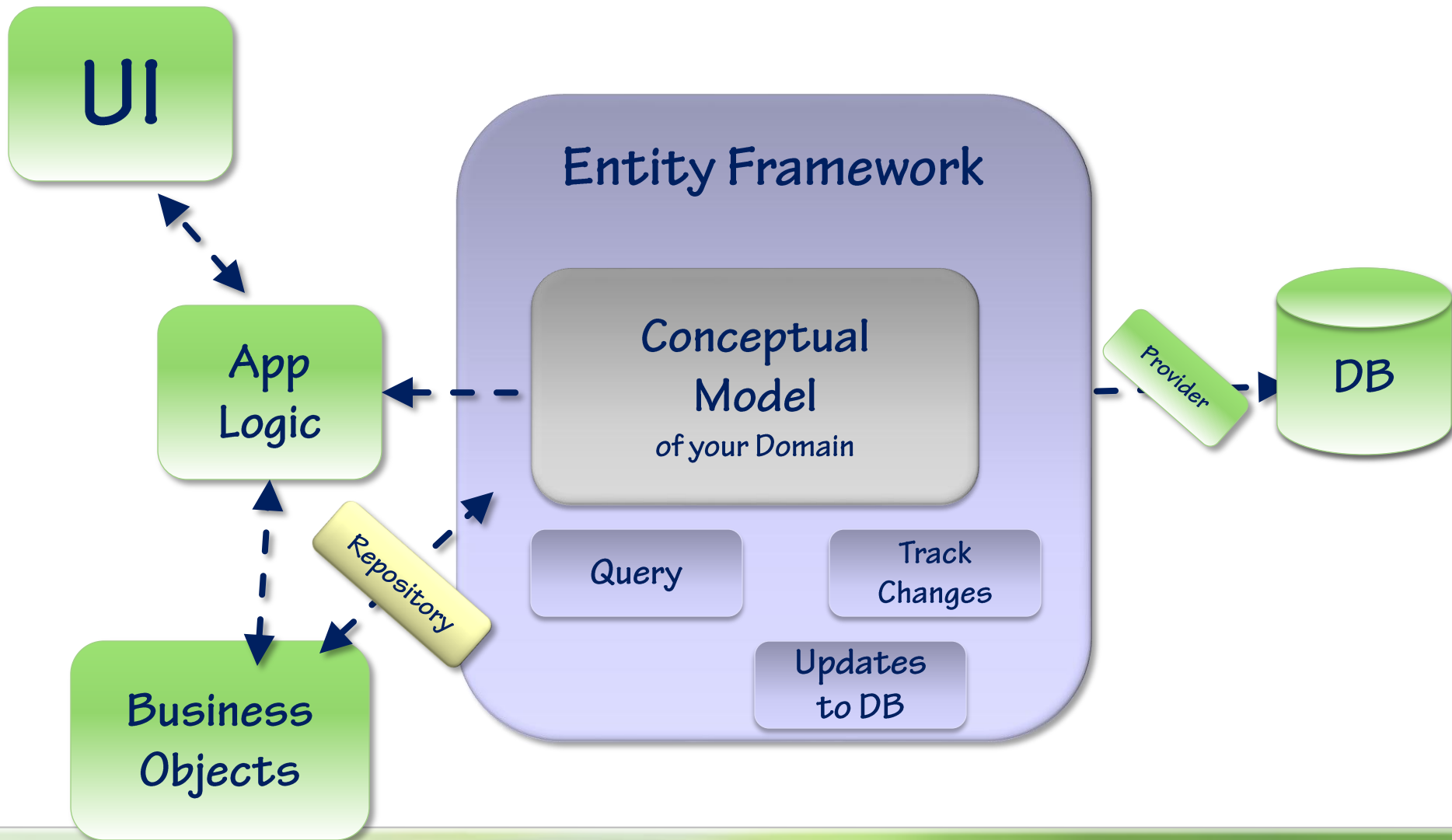
New
Database

Code
First

"M"

```
public class Contact
{
    public Contact()
    {
        Addresses = new List<Address>();
    }
    public int ContactId { get; set; }
    public string FirstName { get; set; }
    public string LastName { get; set; }
    public string Title { get; set; }
    public DateTime AddDate { get; set; }
    public DateTime ModifiedDate { get; set; }
    private string Gender { get; set; }
    public ICollection<Address> Addresses { get; set; }
}
```

Conceptual Model in your Apps



Entity Framework Everywhere

Client/n-Tier

- WinForms
- WPF

Distributed

- ASP.NET EntityDataSource
- ASP.NET Dynamic Data
- ASP.NET MVC
- WCF & Self-Tracking Entities
- WCF RIA Services (for Silverlight)
- WCF Data Services (RESTful data)

Cloud

- SQL Azure
- AppFabric (Velocity/caching)

Summary

- Work with a conceptual model of your business domain
- VS2010 Entity Data Model Designer + .NET 4 APIs
- Works with any database with an EF-savvy provider
- Use in RAD apps or sophisticated enterprise architectures
- Next generation of ADO.NET data access functionality
- Long term commitment from Microsoft

References

- **Pluralsight On Demand**
 - <http://www.pluralsight.com>
- **Programming Entity Framework, 2nd Edition, Julie Lerman**
 - O'Reilly Media, August 2010
 - <http://www.learnentityframework.com>
- **Microsoft Data Developer Center**
 - <http://msdn.com/data>