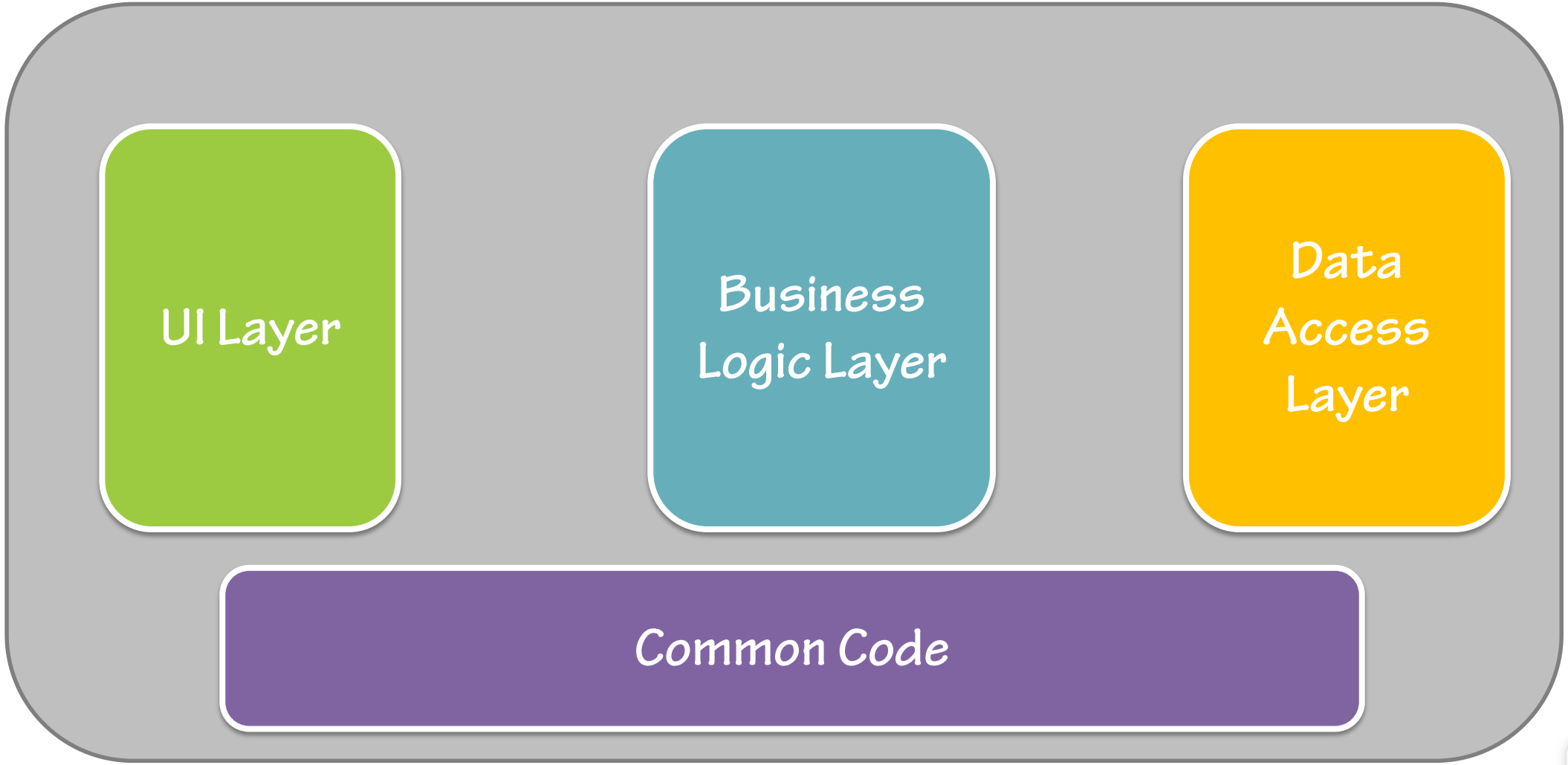
Defining Proper Classes

Deborah Kurata

@deborahkurata | blogs.msmvps.com/deborahk/

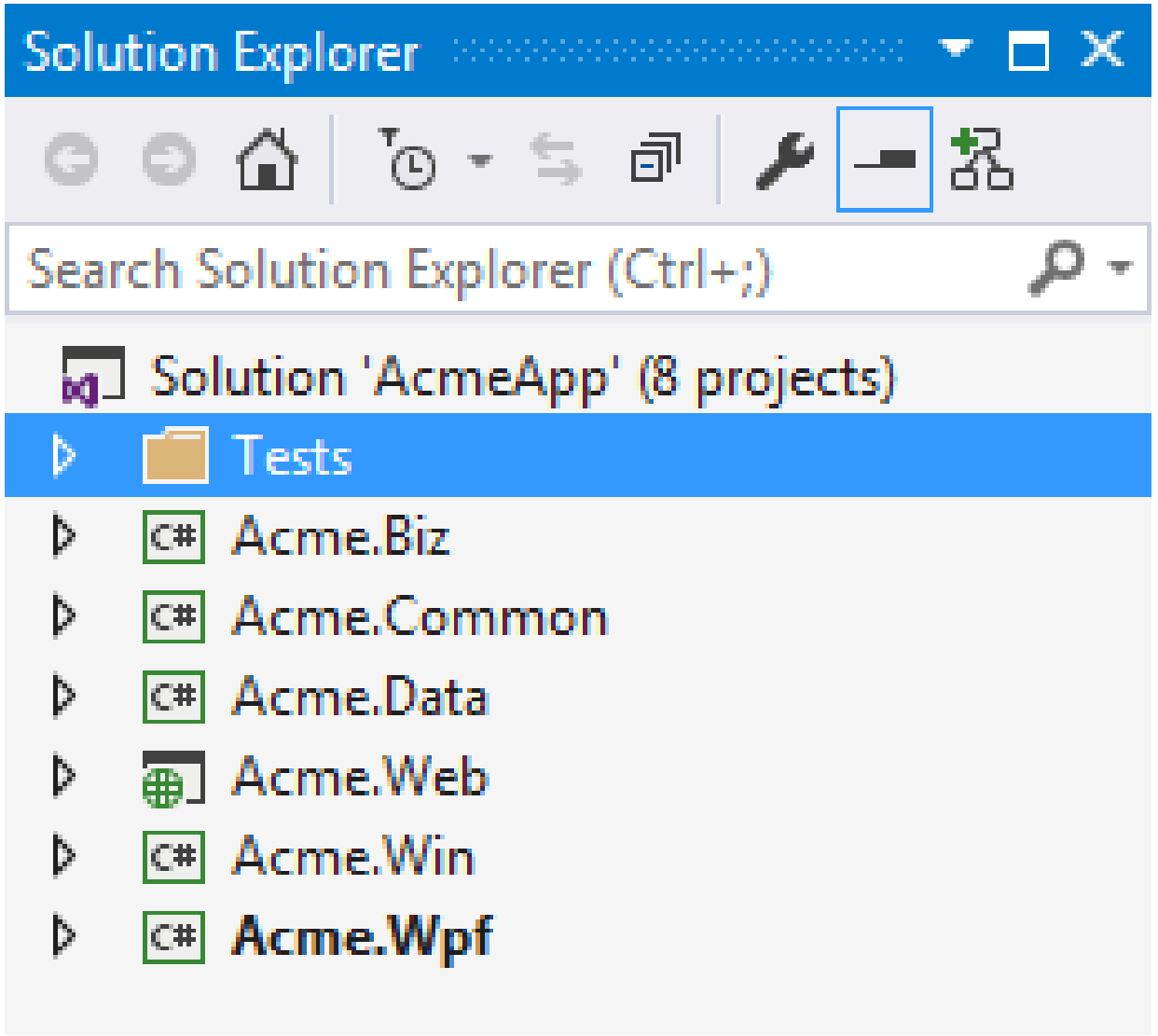
Application Architecture

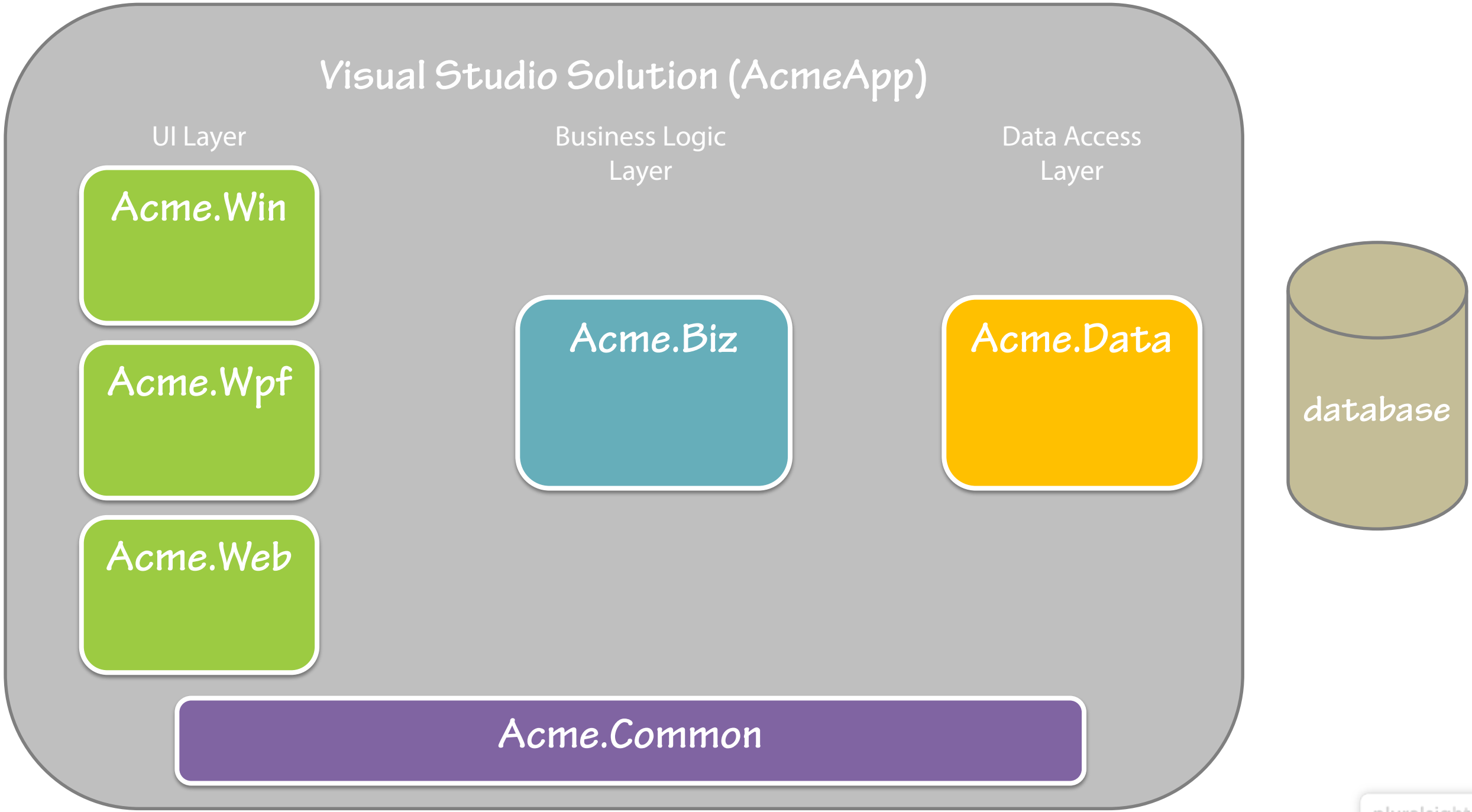


Module Overview

- Application Architecture
- What is a Class?
- Types of Classes
- Unit Testing
- FAQ

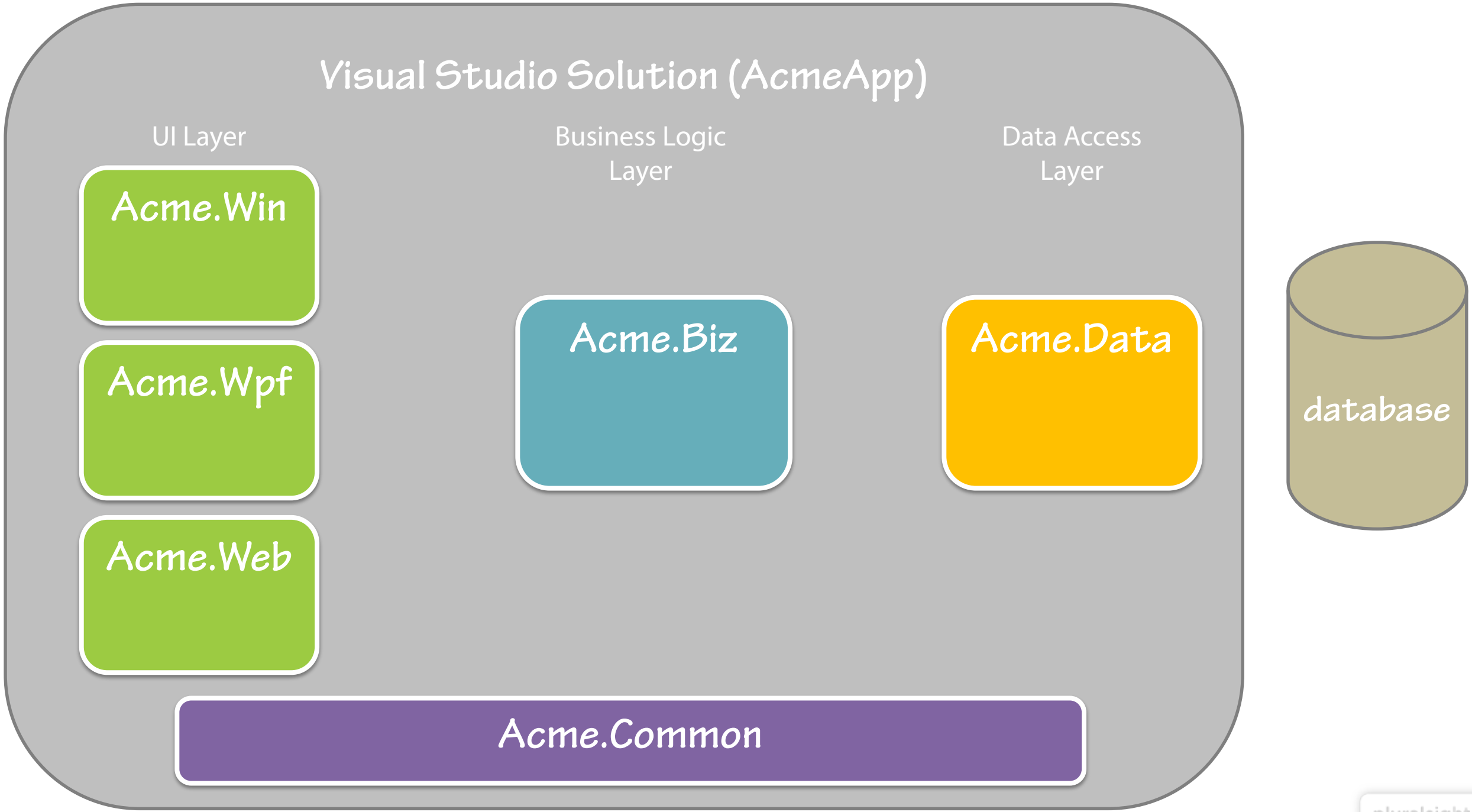
Application Architecture



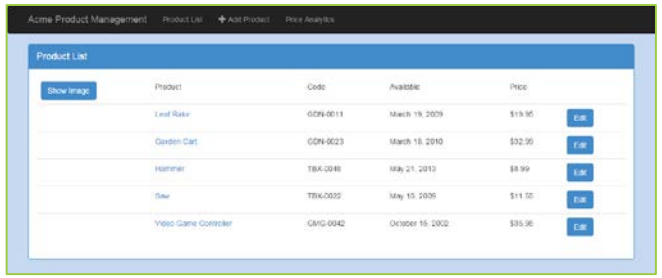


Recommended Viewing

"Getting Started with Entity Framework 6"



C# Application "Things"



Products

Logging

DataSet

Vendors

Email Generation

DbContext

Orders

What Is a Class?

Reference: Wikipedia (German)
https://de.wikipedia.org/wiki/Spritzgie%C3%9Fen

A template or mold

Specifies the traits or data

Specifies the behavior or operations

What Is a Class?

A template or mold

Specifies the traits or data

Specifies the behavior or operations

What Is a Class?

```
namespace Acme.Wpf.Views
{
    /// <summary>
    /// Interaction logic for VendorDetailView.xaml
    /// </summary>
    2 references
    public partial class VendorDetailView : Page
    {
        0 references
        public VendorDetailView()
        {
            InitializeComponent();
        }

    }
}
```

A template or mold

Specifies the traits or data

Specifies the behavior or operations

What Is a Class?

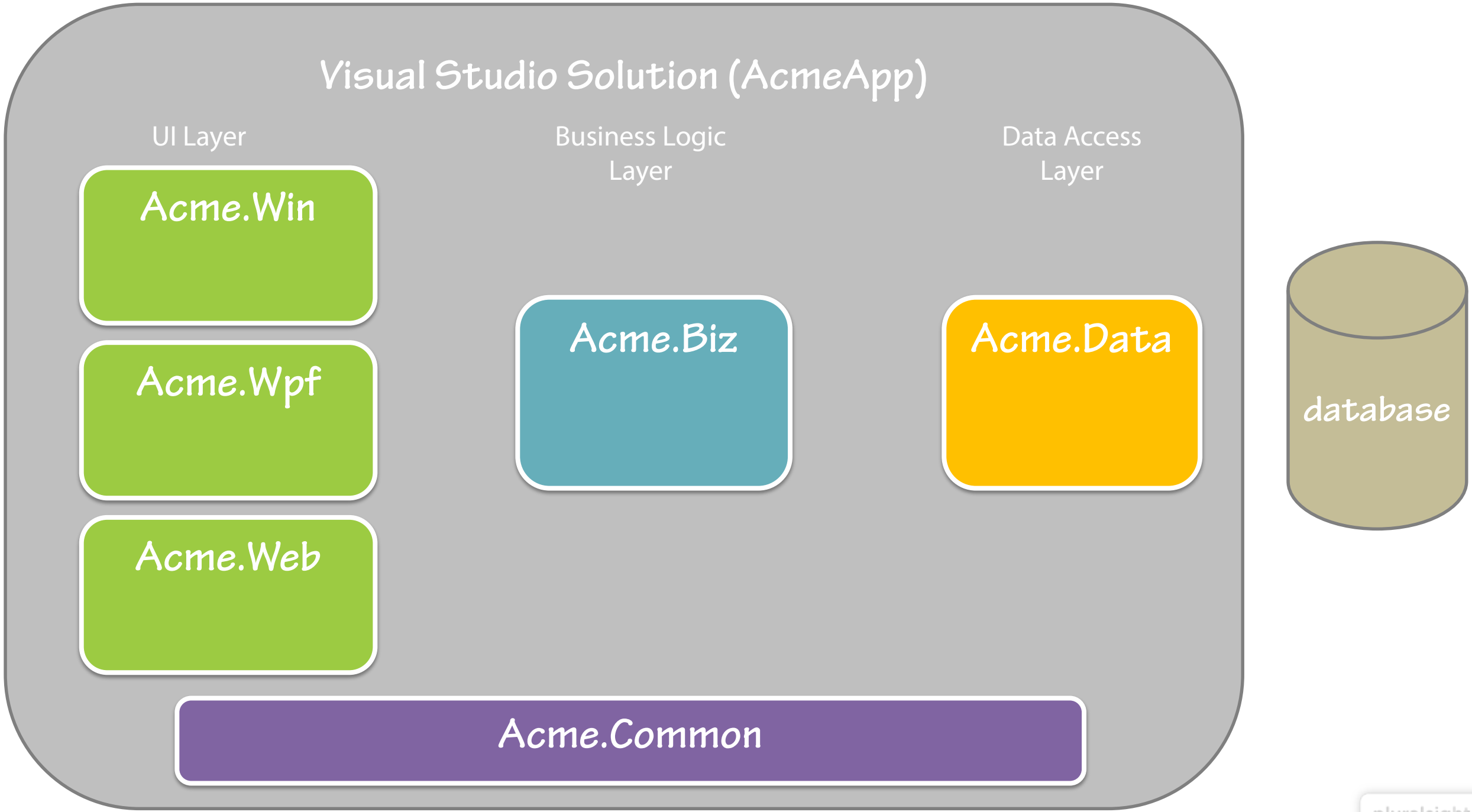
```
namespace Acme.Biz
{
    /// <summary>
    /// Manages the vendors from whom we purchase our inventory.
    /// </summary>
    9 references
    public class Vendor
    {
        1 reference
        public int VendorId { get; set; }
        7 references
        public string CompanyName { get; set; }
        4 references
        public string Email { get; set; }

        /// <summary>
        /// Sends an email to welcome a new vendor.
        /// </summary>
        /// <returns></returns>
        3 references
        public string SendWelcomeEmail(string message)...
    }
}
```

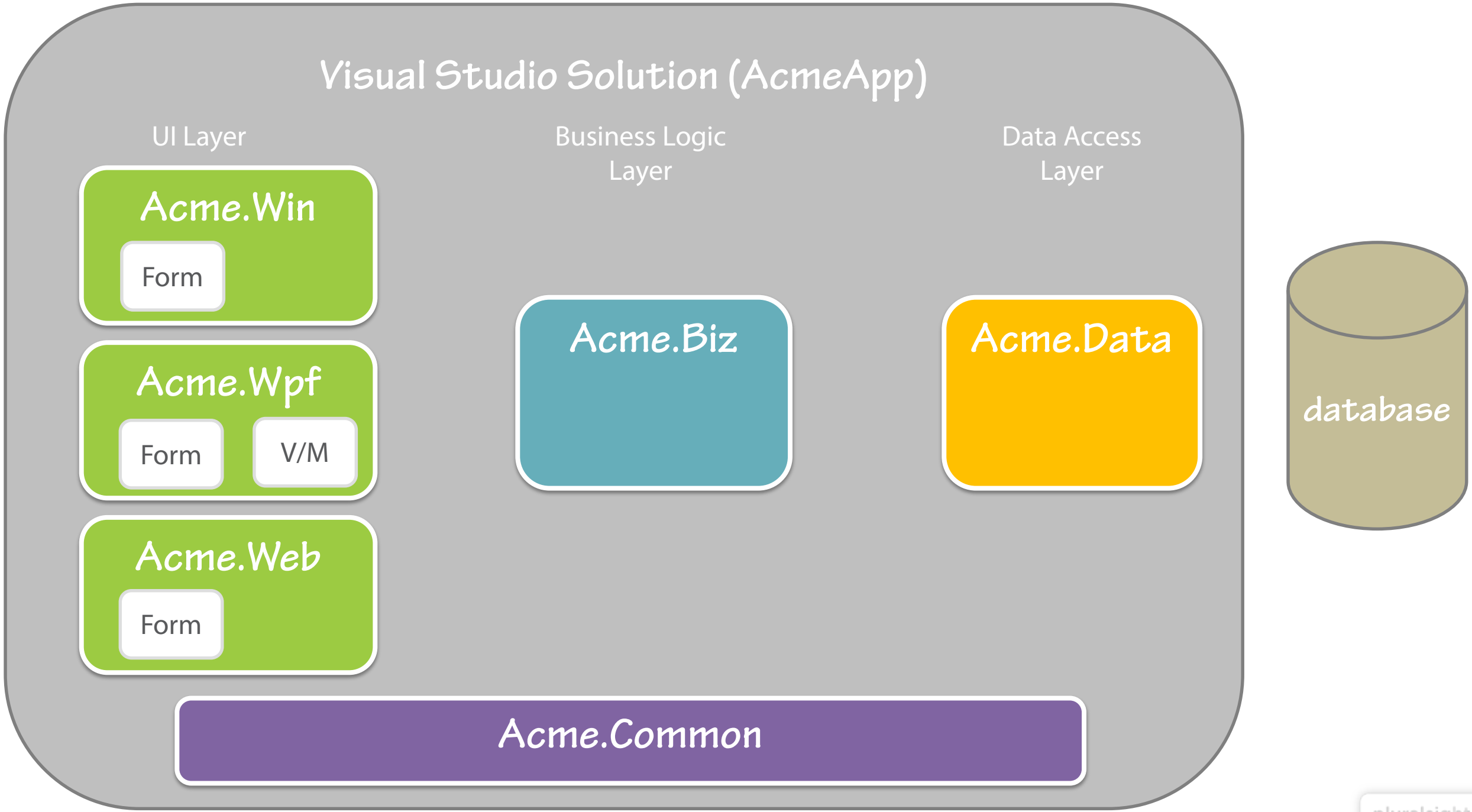
A template or mold

Specifies the traits or data

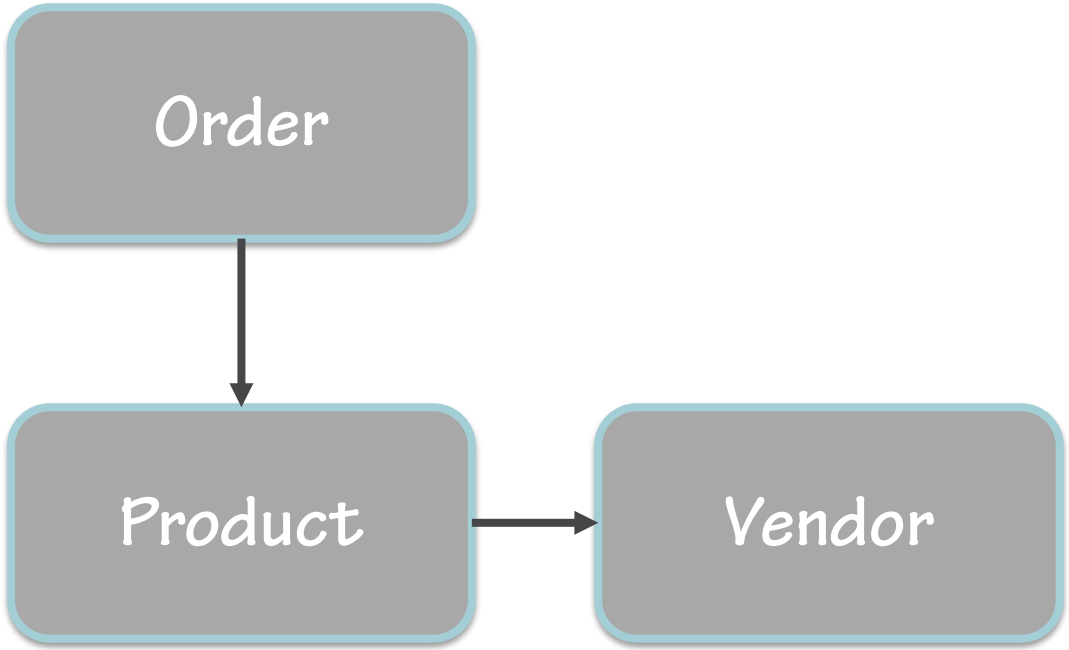
Specifies the behavior or operations

Types of Classes

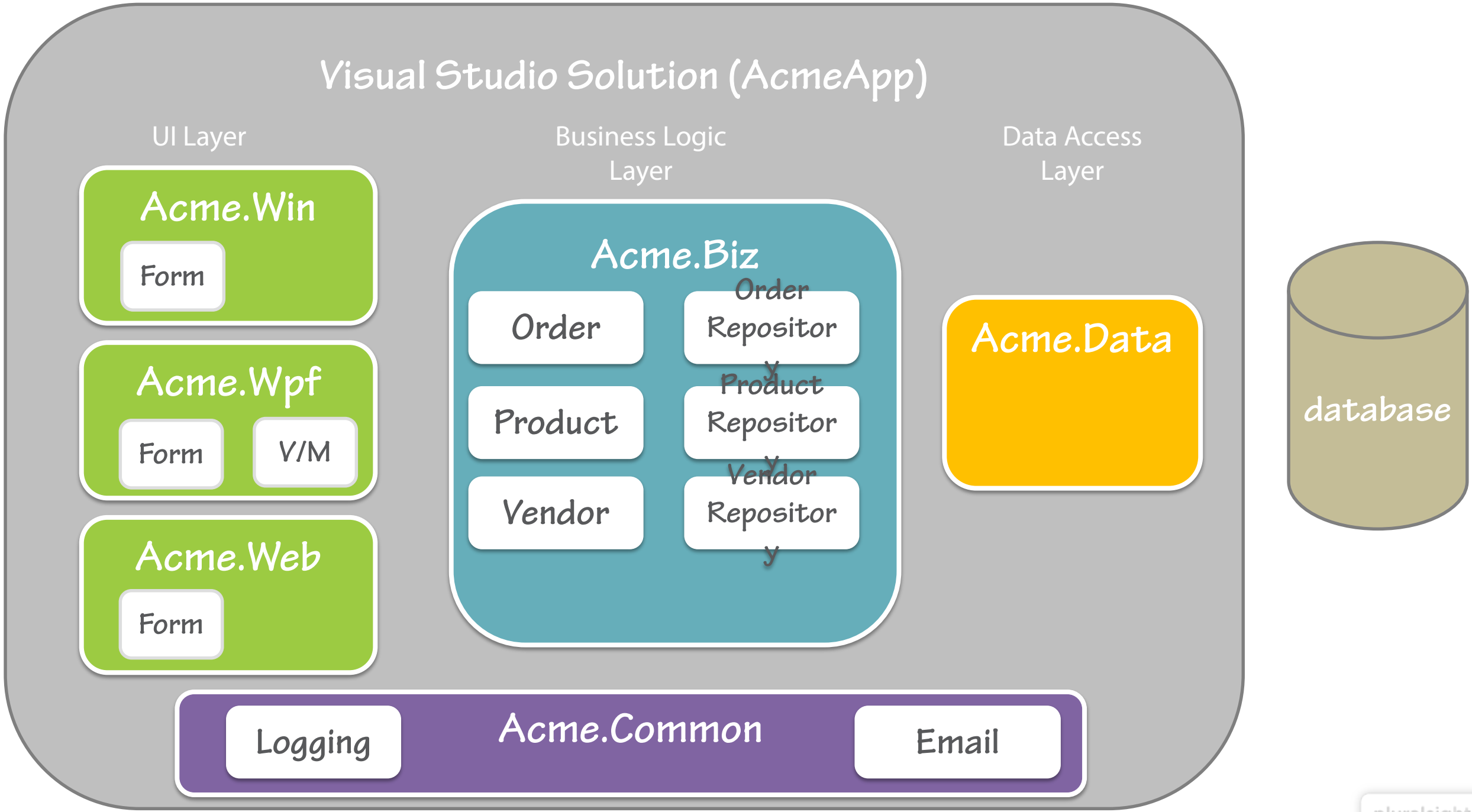



Visual Studio Solution (AcmeApp)
UI Layer
Business Logic Layer
Data Access Layer
Acme.Win
Form
Acme.Wpf
Form
V/M
Acme.Web
Form
Acme.Biz
Acme.Data
database
Acme.Common

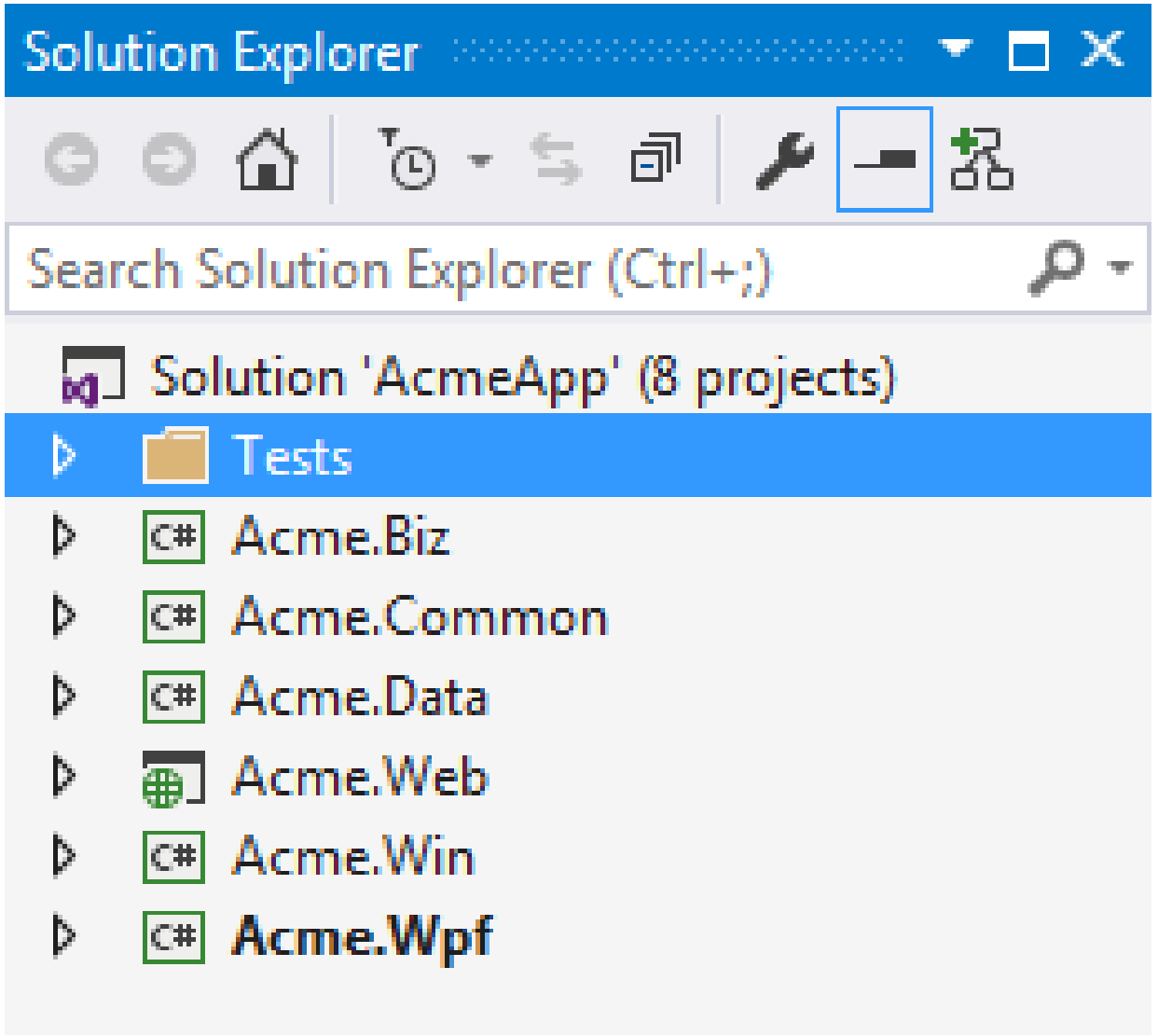
Business Logic Component(s)



Business Domain Model



Application Architecture



Unit Testing

- Tests the behavior of a unit of code
  - Often a method
- Automated
- Defined with code
- Identifies errors

```
public int CalculateDiscount(int percent) {
   return this.total - (this.total * percent/100)
}
```

Tests call CalculateDiscount(x) where x is

- 10
- 100
- 200
- 0
- null

Higher Code Quality

Testable Methods

Test Cases Clarify Scenarios

Tests Identify Failures

Higher Code Quality

Faster and Easier Debugging

Higher Code Quality

Faster and Easier Debugging

Repeatable

Unit Testing in Visual Studio

- MSTest
- Steps:
  - Define the test scenarios
  - Generate the tests
  - Execute the tests

Defensive Coding in C#

This course will show you how to write clean, maintainable, and testable code, and how to keep that code great using defensive coding techniques.

Frequently Asked Questions

- Why is a layered architecture important?
  - Logical components are easier to create, change, extend, and maintain
  - Code is easier to reuse
- What is a class?
  - A template for the objects created at runtime
  - Specifies the data and operations for each entity
- What are the benefits of unit testing?
  - Higher quality code, faster and easier debugging, and they are repeatable over the life of the application

This Module Covered

- Application Architecture
- What is a Class?
- Types of Classes
- Unit Testing