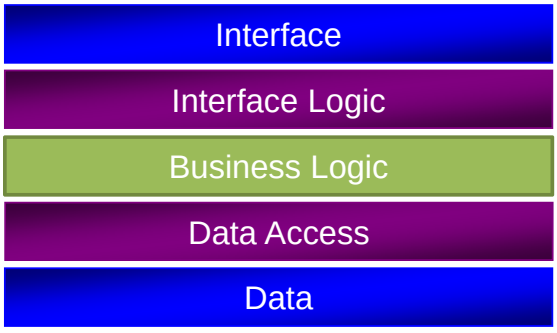
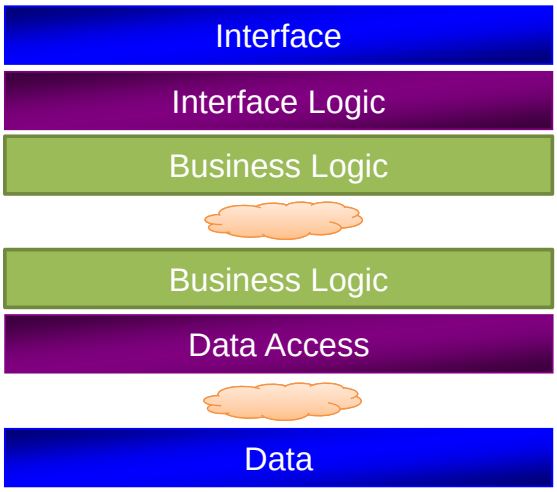
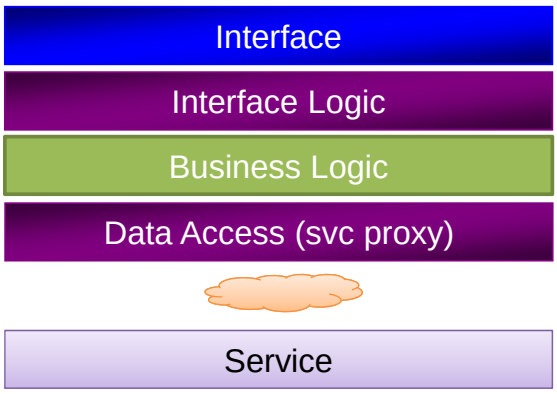
N-Layer App Architecture



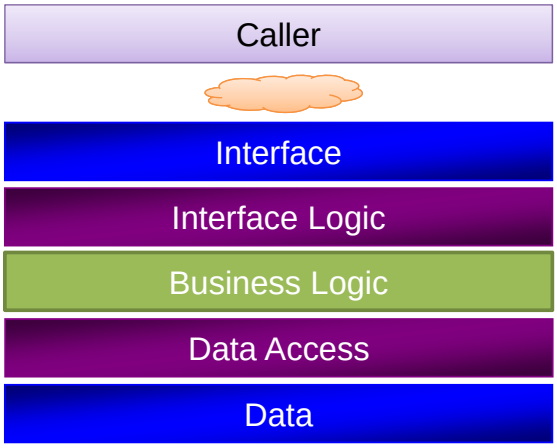
3-Tier App Architecture (shared logic)



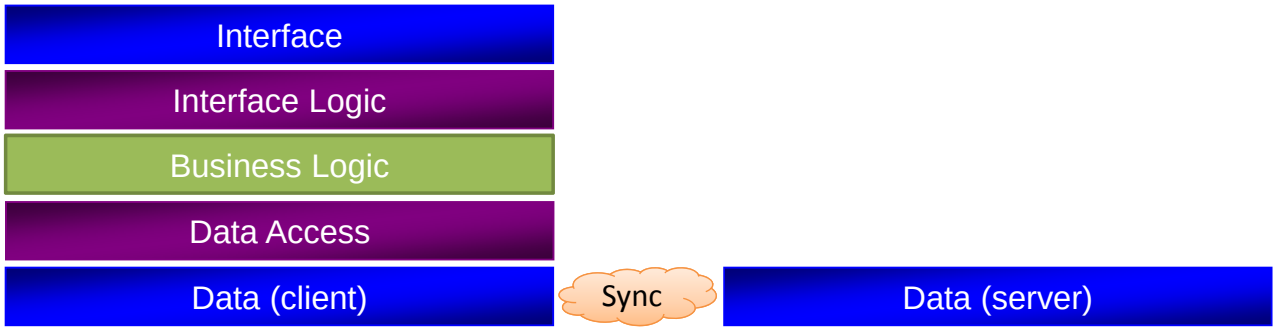
Edge App Architecture



Service Architecture



Disconnected Edge App Architecture



CSLA .NET Development Framework

Lower development, maintenance cost

- Standardize code structure
- Focus on business, not "plumbing"

Built-in support for growth

- Flexible deployment models
- Scalable use of server resources

Abstract .NET

- Leverage, but hide, .NET complexity
- Shield app code from .NET changes

About CSLA .NET

- Cross-platform open-source .NET development framework
- Used by

•Retail •Manufacturing •Telecom •Finance •Shipping •Medical •HR and Payroll •Sports •Security •Media •Web sales	•National •State/Province •Local •City •Police •Fire	•Military •International •NGO •Education

cslanet.com

github.com/marimerllc/csla

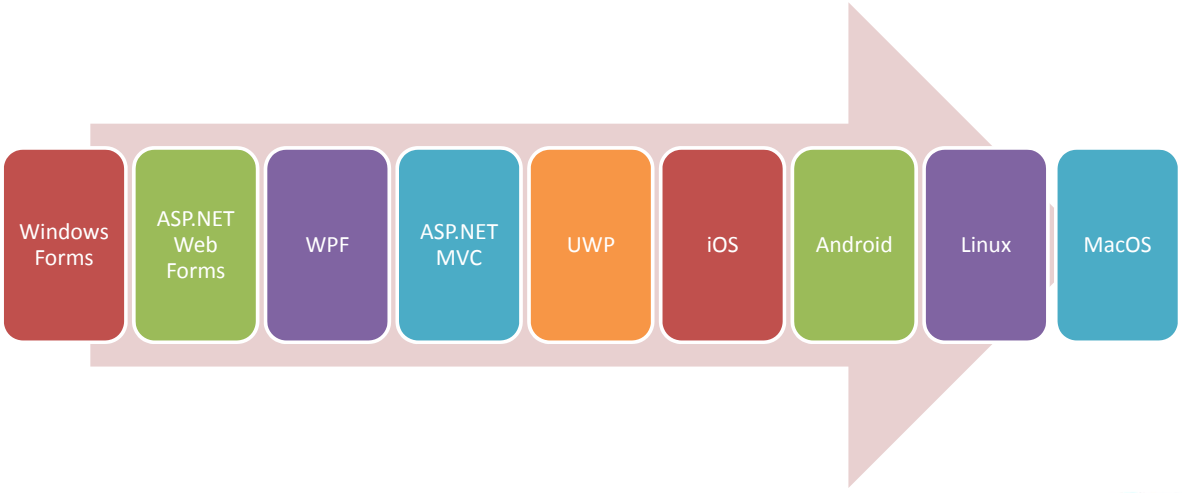
nuget.org/packages?q=csla+.net

21 years!

A Brief History

- 1997: CSLA COM
- 2001: CSLA .NET 1.0
- 2005: Support .NET 2.0, generics
- 2007: Support .NET 3.0: WCF, WPF, WF
- 2008: Support Silverlight, Entity Framework
- 2010: Support for .NET 4.0, Silverlight 4, WP 7
- 2011: Support for mono, Android, WP 7.5 “Mango”
- 2012: Support for .NET 4.5, Windows 8 WinRT, WinPRT
- 2013: Support for iOS (and Android) via Xamarin
- 2015: Support for .NET 4.6, UWP (Windows 10)
- 2017: Support for .NET Core and NetStandard

Supported Platforms



CSLA .NET Goals

Minimize/standardize coding

- Decrease maintenance costs

Require software layering

- Enforce layering/separation of concerns
- Decrease maintenance costs

Enable responsibility-driven business layer

- Responsibility-driven business objects
- Encapsulate validation/authorization/logic
- Support data binding (Windows, Web, XAML)

Abstract mobile object architecture

- Codeless switch between 1, 2, 3, 4-tier models
- Transport neutral (HTTP, WCF, Remoting, Enterprise Services)
- Gain scalability and security

Non-goals

- UI framework
- Object-relational mapping (ORM)
- Support for data-centric OO designs
- Async messaging (pure SOA)
- Languages beyond VB/C#

CSLA .NET Primary Features

Business rules	• Business, validation, authorization
Data binding	• XAML, Xamarin, Windows, Web
Mobile objects	• 1-, 2-, 3- and 4-tier deployment models
Abstract persistence	• ADO.NET, Entity Framework, others
LINQ	• Synced lists, data binding
N-level undo	• Smart client platforms

Working with Xamarin

DEMO

rocky@lhotka.net
www.lhotka.net
www.cslanet.com

THANK YOU

Magenic Microsoft Regional Director MVP Microsoft Most Valuable Professional CSLA .NET