

# Refactoring Legacy Code to Follow OCP and LSP



**Steve Smith**

Founder and Principal Architect, NimblePros

@ardalis | ardalisc.com

# Overview



## Identify Cross-Cutting Concerns

- Apply Patterns to Follow OCP

## Identify Misuse of Types

- Ensure substitutability to Follow LSP



## Wikipedia – Cross Cutting Concerns

**“... cross-cutting concerns are aspects of a program that affect several modules, without the possibility of being encapsulated in any [one] of them.”**



# Examples of Cross-Cutting Concerns

**Business Rules**

**Data Access / Persistence**

**Validation**

**Logging**

**Exception Handling**

**Monitoring**





## Refactoring

- Open/Closed Principle
- Liskov Substitution Principle
- Cleaning up CalculatePrice()



# Review After Refactoring to Apply OCP and LSP

**Pulled Error Handling  
Out**

**Pulled Data Access  
Out**

**Focus on Business  
Logic**



# More Files – More Focused Files

## Interfaces

```
► ▲ □ Interfaces
  ▷ ▲ C# ICabinetDataService.cs
  ▷ ▲ C# IFeatureDataService.cs
  ▷ ▲ C# IGetUserMarkup.cs
  ▷ ▲ C# IKitchenDataService.cs
  ▷ ▲ C# IOrderDataService.cs
  ▷ ▲ C# IPartCostDataService.cs
  ▷ ▲ C# IPartPricingService.cs
  ▷ ▲ C# IPriceCalculationStrategy.cs
  ▷ ▲ C# IPricingService.cs
  ▷ ▲ C# IWallDataService.cs
```

## Classes

```
► ▲ □ Services
  ▷ ▲ C# DefaultPriceCalculationStrategy.cs
  ▷ ▲ C# GlobalHelpers.cs
  ▷ ▲ C# NewOrderPriceCalculationStrategy.cs
  ▷ ▲ C# PriceReportCalculationStrategy.cs
  ▷ ▲ C# PricingService.cs
  ▷ ▲ C# PricingServiceDecorator.cs
  ▷ ▲ C# SqliteCabinetDataService.cs
  ▷ ▲ C# SqliteFeatureDataService.cs
  ▷ ▲ C# Sqlite GetUserMarkupService.cs
  ▷ ▲ C# SqliteKitchenDataService.cs
  ▷ ▲ C# SqliteOrderDataService.cs
  ▷ ▲ C# SqlitePartCostDataService.cs
  ▷ ▲ C# SqlitePartPricingService.cs
  ▷ ▲ C# SqliteWallDataService.cs
```



# Summary



## Additional refactoring steps

- Extracting and injecting data access
- Using a decorator to follow OCP
- Ensuring all interface implementations follow LSP

## Identifying cross-cutting concerns

### Remember:

- Interfaces tell us WHAT
- Implementations tell us HOW



**Up Next:**

# **Assessing and Testing SOLID Code**

---

