Building Reusable Components

Deborah Kurata
http://msmvps.com/blogs/deborahk/
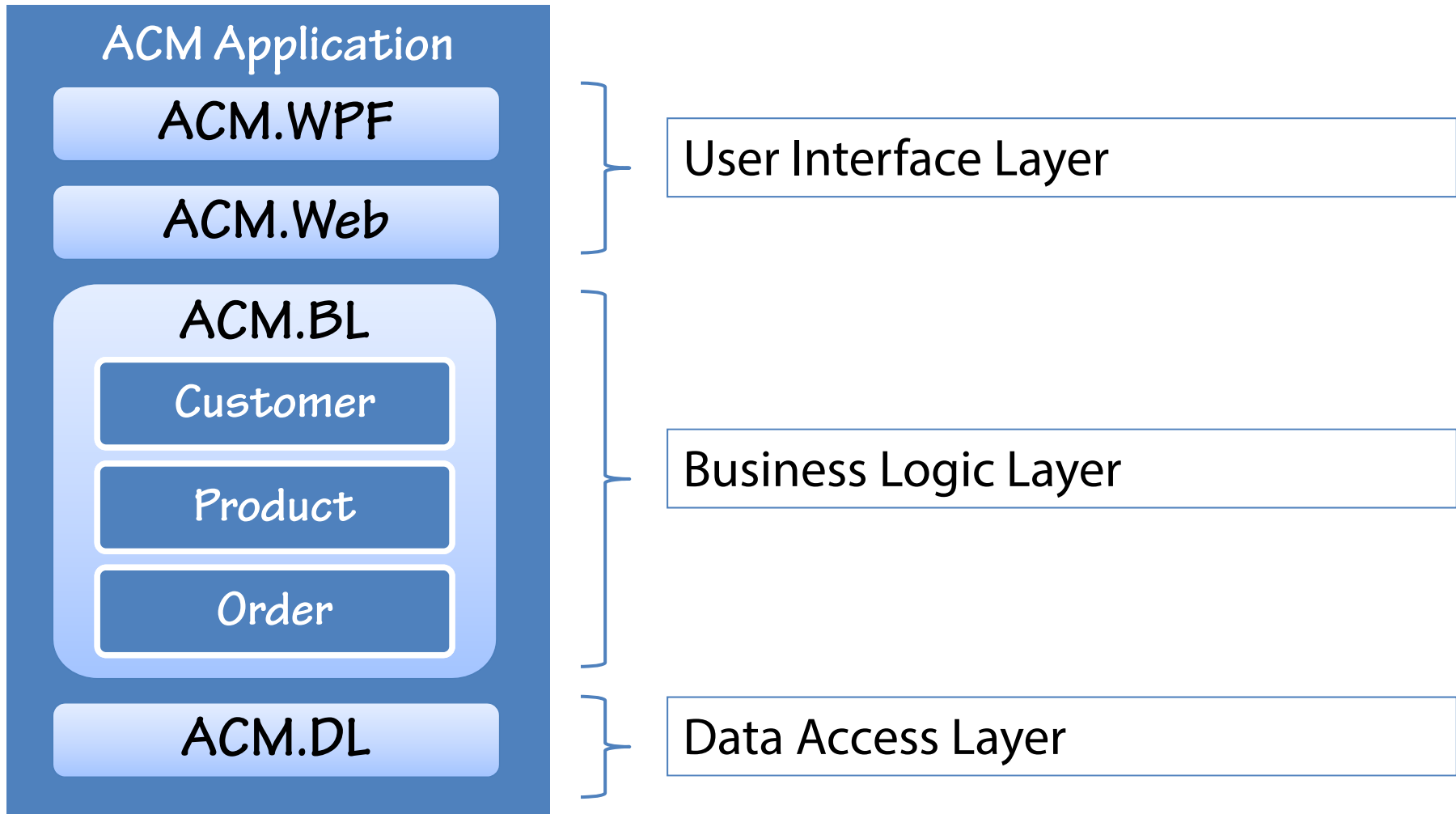@DeborahKurata
deborahk@insteptech.com
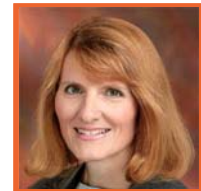
pluralsight
hardcore dev and IT training

Components

ACM Application		
ACM.WPF	⎱	User Interface Layer
ACM.Web	⎰	
ACM.BL Customer Product Order	⎱⎰	Business Logic Layer
ACM.DL	⎱⎰	Data Access Layer

Reusable Library Component

ACM Application

ACM.WPF

ACM.Web

ACM.BL
- Customer
- Product
- Order

ACM.DL

Acme.Common
- StringHandler
- LoggingService
- EmailService
- …
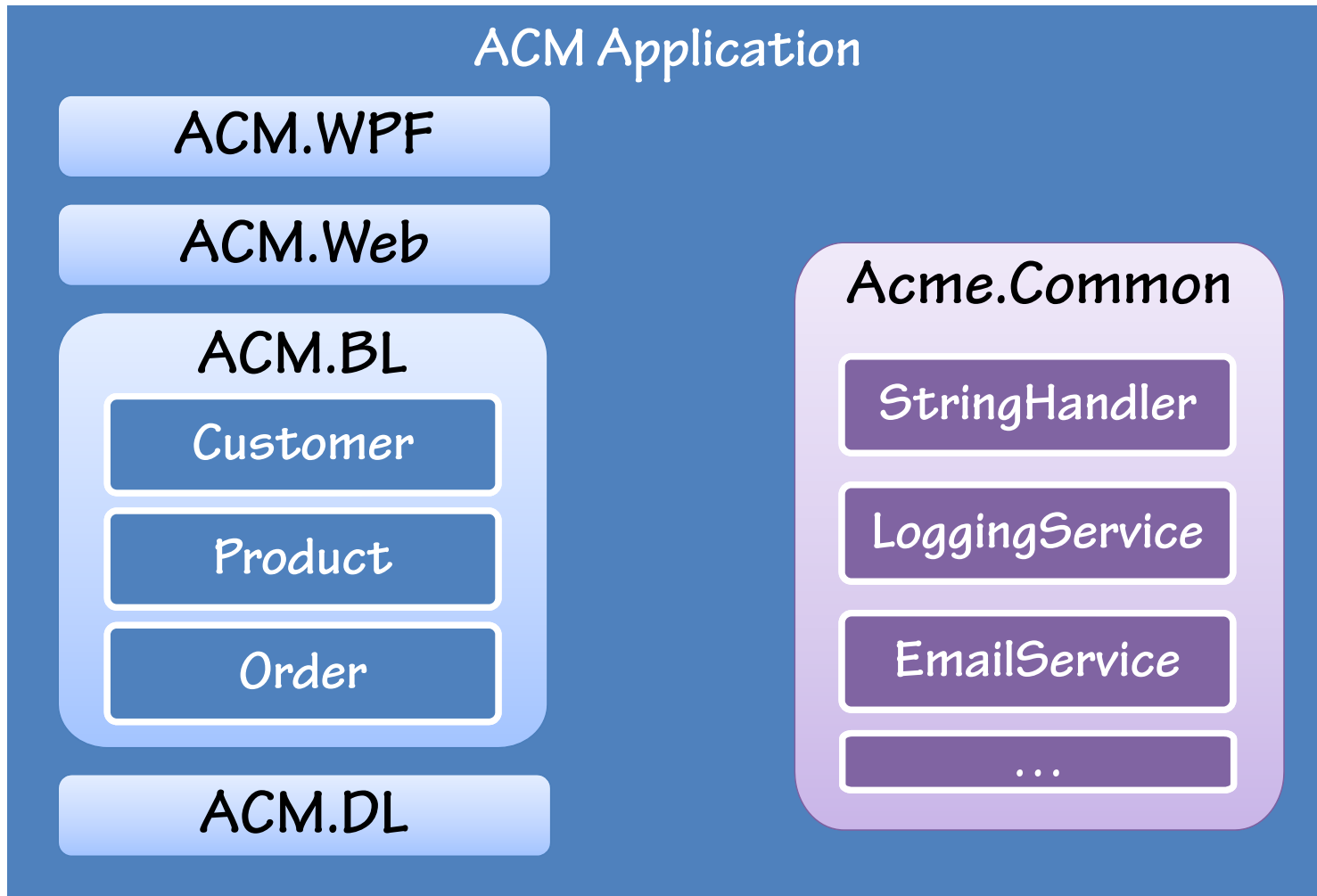
Module Outline

Building a Reusable Component

Testing the Reusable Component

Using the Reusable Component

Static Classes

Extension Methods

Scenario

Third Party System → `SonicScrewdriver` → Product Data

Product Data → `SonicScrewdriver` → Our System

Our System → Sonic Screwdriver →
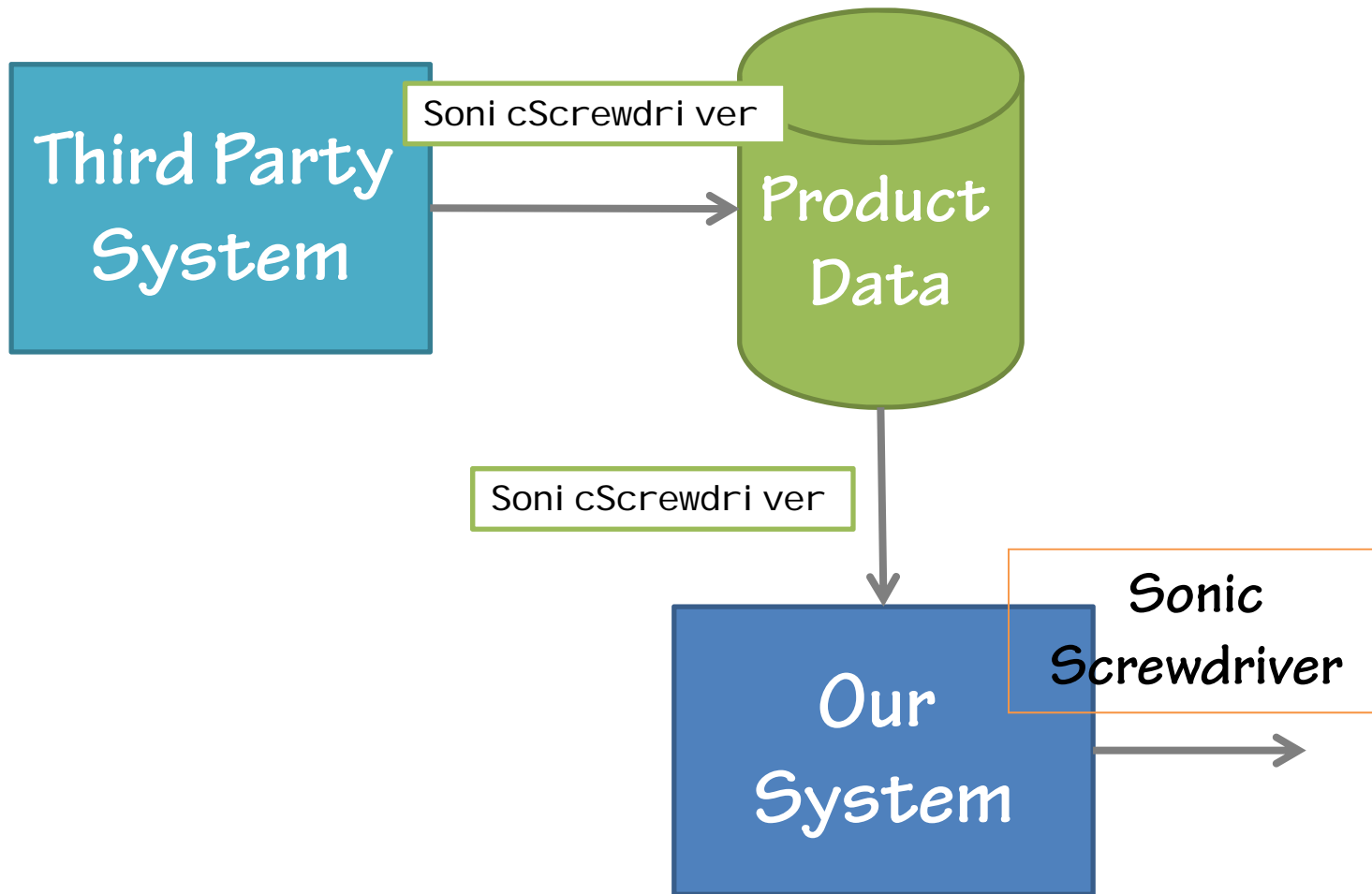
Static Class

```
public class StringHandler{
    public string InsertSpaces(string source)
```

```
StringHandler handler = new StringHandler();
handler.InsertSpaces(productName);
```

```
public static class StringHandler{
    public static string InsertSpaces(string source)
```

```
StringHandler.InsertSpaces(productName);
```

Extension Methods

```
public class StringHandler{
    public string InsertSpaces(string source)
```

```
StringHandler handler = new StringHandler();
handler.InsertSpaces(productName);
```

```
public static class StringHandler{
    public static string InsertSpaces(string source)
```

```
StringHandler.InsertSpaces(productName);
```

```
public static class StringHandler{
    public static string InsertSpaces(this string source)
```

```
productName.InsertSpaces();
```

Static Method vs Extension Method

Instance parameter? → **Operate on that Instance?** → **Appear in Intellisense?**

```
public class StringHandler{
    public string InsertSpaces(string source)
```

Summary

Building a Reusable Component

Testing the Reusable Component

Using the Reusable Component

Static Classes

Extension Methods

Acme.Common

StringHandler

LoggingService

EmailService

…