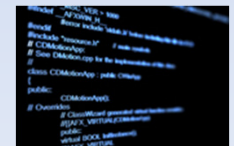


.NET and C# in a TRAVERSE world

.NET framework is a development environment that provides:

- A rich set of built-in functionality
- A single programming model between windows and internet based applications
- Easy deployment and extensibility
- Portability of compiled applications
- Handling program configuration via config files
- Memory management (garbage collection)
- Cross language interaction
- Thousands of types grouped into namespaces – examples:
 - System.Windows.Forms – building windows GUI applications
 - System.XML – processing XML schemas and data



.NET and C# in a TRAVERSE world

TRAVERSE uses the n-tiered approach supported by .NET framework to split application into 3 tiers or layers.

The presentation (or client) layer is responsible for:

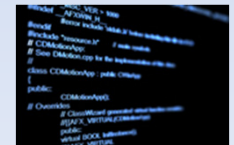
- Gathering information from the user
 - Sending the user information to the business layer for processing
 - Receiving the results of the business layer processing
 - Presenting those results to the user
- TRAVERSE.Client.xxxxx.dll

The business layer is responsible for:

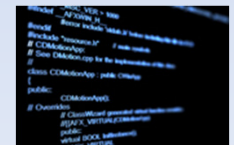
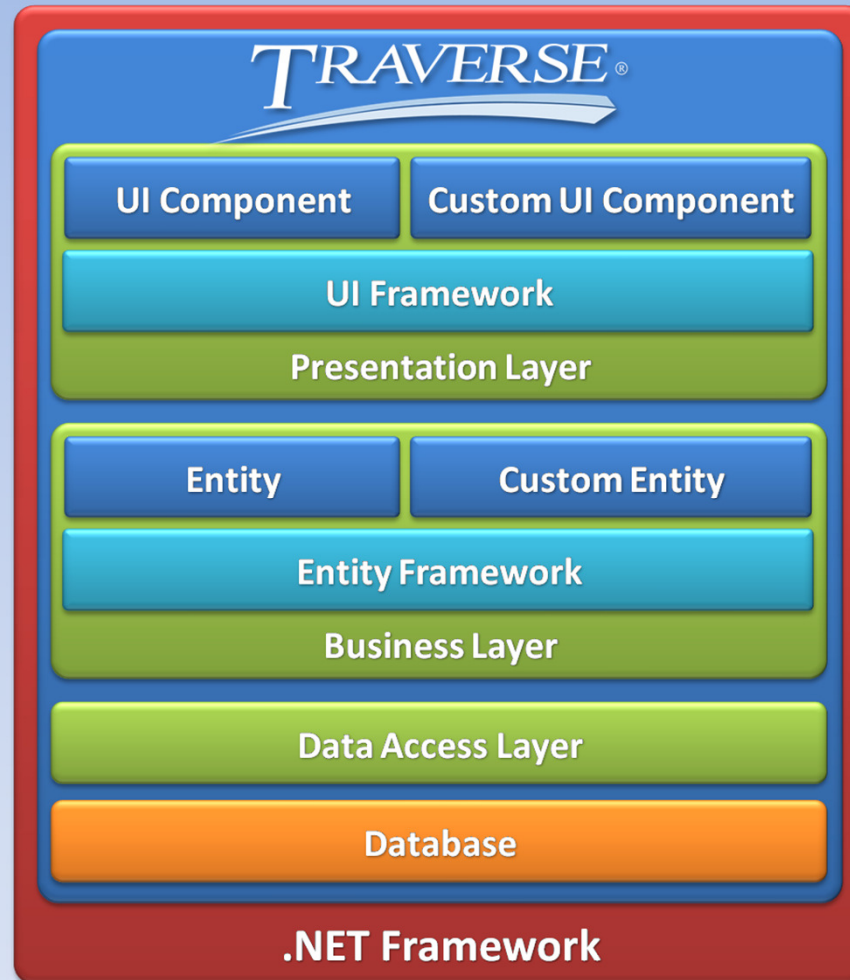
- Receiving input from the client layer
 - Interacting with the data layer to perform the business operations
 - Implementing business logic
 - Sending the processed results to the client layer
- TRAVERSE.Business.xxxxx.dll

The data layer is responsible for:

- Storage of data
 - Retrieval of data
 - Maintenance of data
 - Integrity of data
- TRAVERSE.Data.xxxxx.dll



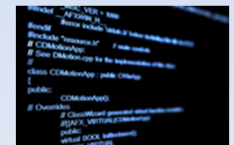
.NET and C# in a TRAVERSE world



.NET and C# in a TRAVERSE world

TRAVERSE v11 utilizes Object Oriented Programming:

- Defines units of data and functions (of an entity) in classes (modularizes code)
- Allows users to only need to interact with a class via methods provided by the class
- Allows for Inheritance – new instances of classes derive from base classes
- Allows for Enhancement and Specification – redefining or extending the functionality of the base class
- Allows for Polymorphism – enables an entity to accept different parameters at execution that will allow method to handle multiple types



.NET and C# in a TRAVERSE world

C# code

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Windows.Forms;
using TRAVERSE.Business;
using TRAVERSE.Client;
using TRAVERSE.Business.Contacts;
using TRAVERSE.Client.Contacts;

namespace OSI.MS(      .CustomCustomers
{
    class CustomCustomerControl:CustomerControl
    {
        public CustomCustomerControl(IPlugin host)
        {
            this.Load +=new EventHandler(CustomCustomerControl_Load);
        }

        void CustomCustomerControl_Load(object sender, EventArgs e)
        {
            //define field(s) to add
            CustomFieldsList fields = new CustomFieldsList();
            try
            {
                fields.Add(this.AllCustomFields["Logo"]);
            }
            catch { }

            //add field(s)
            this.CreateCustomFields(null, fields);
            this.CustomFieldControls["Logo"].DoubleClick += new EventHandler(CustomerLogo_DoubleClick);
        }
    }
}
```

```
using System;
using System.Collections.Generic;
using System.Text;
using TRAVERSE.Client.Contacts;

namespace      Client.CustomCustomers
{
    public class CustomCustomerPlugin : CustomerPlugin
    {
        public override void Initialize()
        {
            this.MainInterface = new CustomCustomerControl(this);
        }
    }
}
```

Inheritance

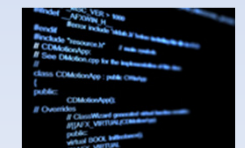
Extension

```
if (e.PropertyName.Equals (Transaction))
{
    ▲1 of 3 ▼ bool string.Equals (object obj)
    obj: An System.Object.
}
```

Polymorphism

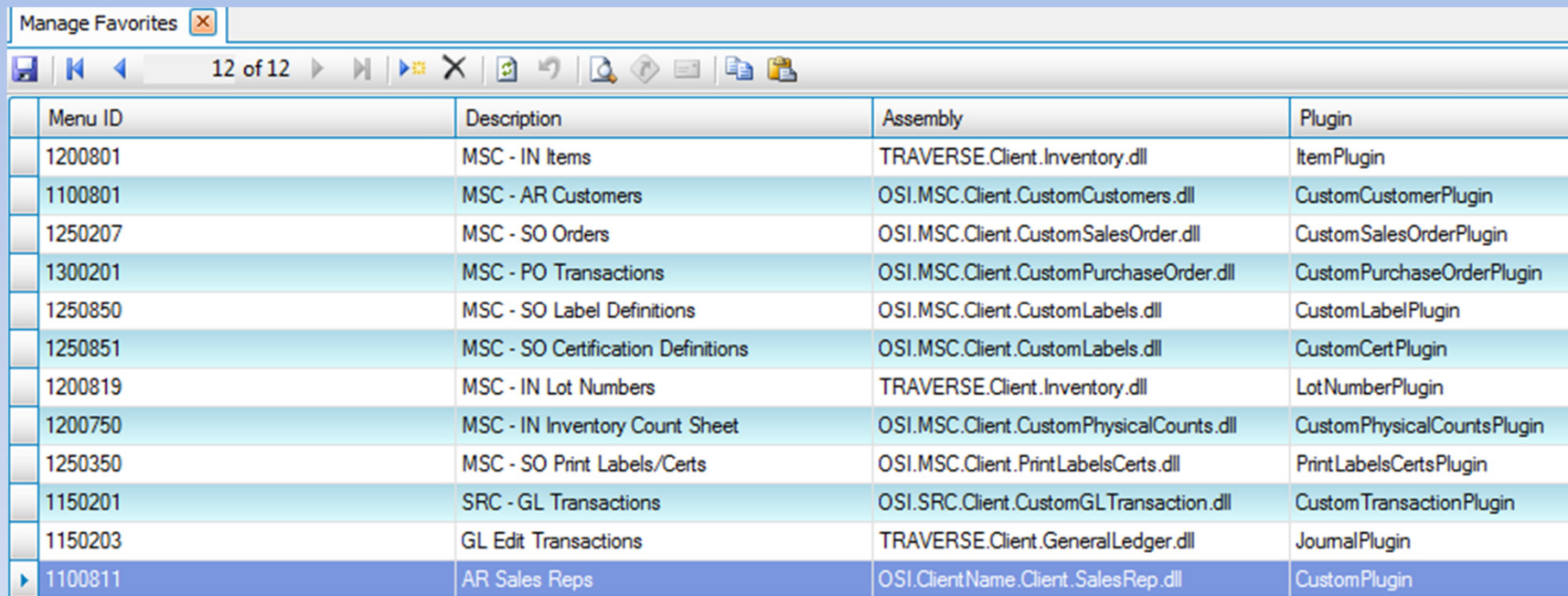
```
if (e.PropertyName.Equals (Transaction))
{
    ▲2 of 3 ▼ bool string.Equals (string value)
    value: A System.String.
}
```

```
if (e.PropertyName.Equals (TransactionDetail.Columns.ItemId))
{
    ▲3 of 3 ▼ bool string.Equals (string value, StringComparison comparisonType)
    value: A System.String object.
}
```

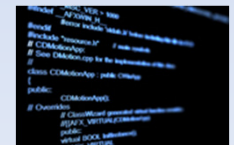


.NET and C# in a TRAVERSE world

TRAVERSE Menu – using Assembly Name and Plugin from modified C# project



Menu ID	Description	Assembly	Plugin
1200801	MSC - IN Items	TRAVERSE.Client.Inventory.dll	ItemPlugin
1100801	MSC - AR Customers	OSI.MSC.Client.CustomCustomers.dll	CustomCustomerPlugin
1250207	MSC - SO Orders	OSI.MSC.Client.CustomSalesOrder.dll	CustomSalesOrderPlugin
1300201	MSC - PO Transactions	OSI.MSC.Client.CustomPurchaseOrder.dll	CustomPurchaseOrderPlugin
1250850	MSC - SO Label Definitions	OSI.MSC.Client.CustomLabels.dll	CustomLabelPlugin
1250851	MSC - SO Certification Definitions	OSI.MSC.Client.CustomLabels.dll	CustomCertPlugin
1200819	MSC - IN Lot Numbers	TRAVERSE.Client.Inventory.dll	LotNumberPlugin
1200750	MSC - IN Inventory Count Sheet	OSI.MSC.Client.CustomPhysicalCounts.dll	CustomPhysicalCountsPlugin
1250350	MSC - SO Print Labels/Certs	OSI.MSC.Client.PrintLabelsCerts.dll	PrintLabelsCertsPlugin
1150201	SRC - GL Transactions	OSI.SRC.Client.CustomGLTransaction.dll	CustomTransactionPlugin
1150203	GL Edit Transactions	TRAVERSE.Client.GeneralLedger.dll	JoumalPlugin
1100811	AR Sales Reps	OSI.ClientName.Client.SalesRep.dll	CustomPlugin



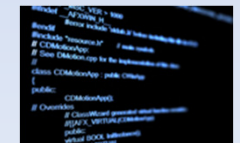
.NET and C# in a TRAVERSE world

TRAVERSE Program directory – showing N-Tiered assemblies: Data/Business/Client

The screenshot displays two overlapping Windows Explorer windows. The main window shows the directory structure of the TRAVERSE program, listing various DLLs categorized into Business and Client tiers. A sub-window provides a detailed view of the DataProvider folder, listing specific data-related DLLs.

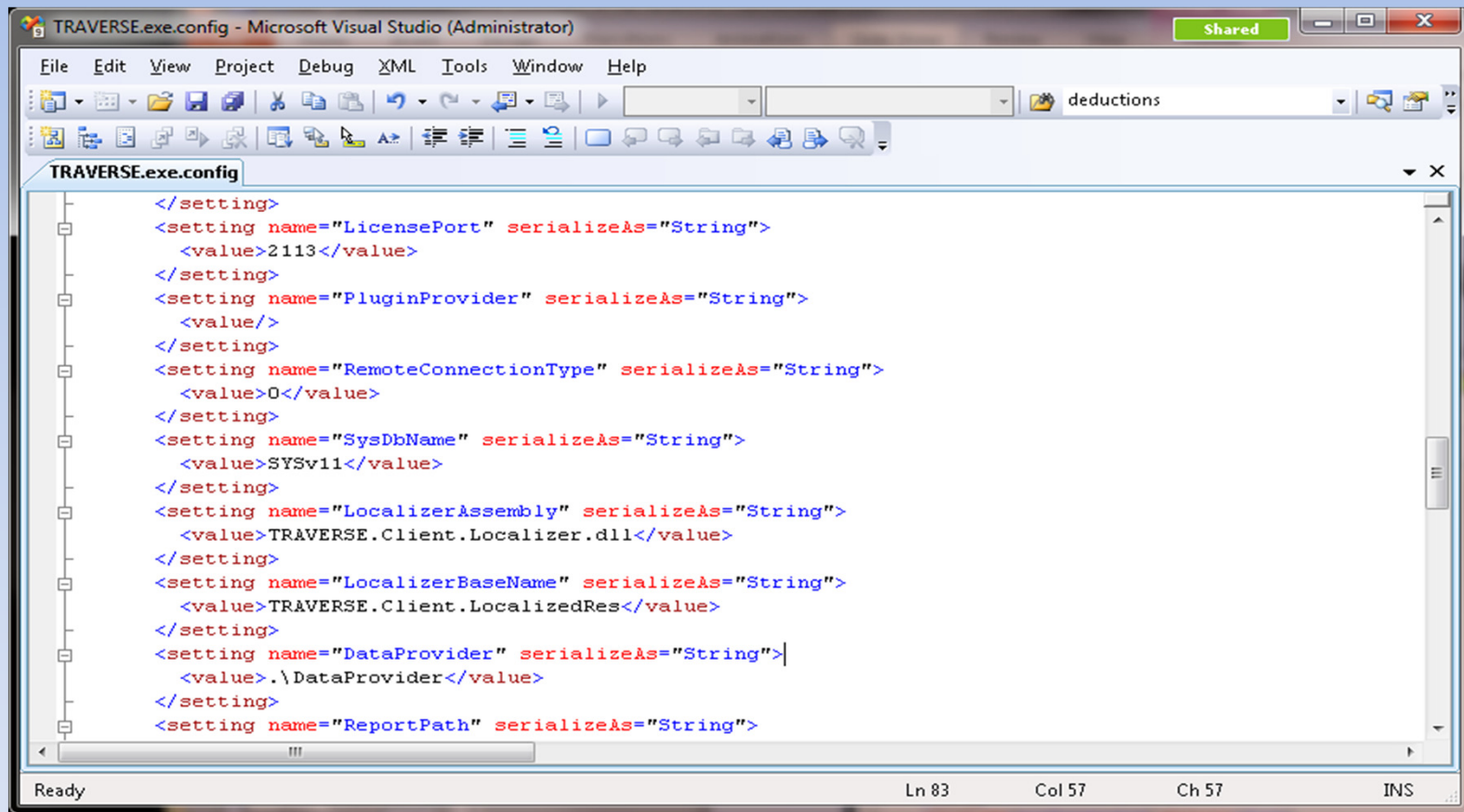
Name	Date modified	Type	Size
TRAVERSE.Business.SalesRep.dll	7/30/2010 11:18 AM	Application extens...	
TRAVERSE.Business.Shell.dll	7/27/2010 2:18 PM	Application extens...	
TRAVERSE.Business.Sys.dll			
TRAVERSE.Business.SystemAudit.dll			
TRAVERSE.Business.Tax.dll			
TRAVERSE.Business.TaxUpdate.dll			
TRAVERSE.Business.Ten99.dll			
TRAVERSE.Business.TimeTickets.dll			
TRAVERSE.Business.Transaction.dll			
TRAVERSE.Business.TransLink.dll			
TRAVERSE.Business.UserStore.dll			
TRAVERSE.Business.Utility.dll			
TRAVERSE.Business.Vendor.dll			
TRAVERSE.Business.Views.dll			
TRAVERSE.Business.VoidCheck.dll			
TRAVERSE.Client.AccountsPayable.dll			
TRAVERSE.Client.AccountsReceivable.dll			
TRAVERSE.Client.APChecks.dll			
TRAVERSE.Client.APHistory.dll			
TRAVERSE.Client.APTransaction.dll			
TRAVERSE.Client.ARHistory.dll			

Name	Date modified	Type	Size
TRAVERSE.Data.AccountsPayable.dll	7/30/2010 11:17 AM	Application extens...	
TRAVERSE.Data.AccountsReceivable.dll	7/30/2010 11:18 AM	Application extens...	
TRAVERSE.Data.ActivityLog.dll	7/30/2010 11:16 AM	Application extens...	
TRAVERSE.Data.APHistory.dll	7/30/2010 11:18 AM	Application extens...	
TRAVERSE.Data.APTransaction.dll	7/30/2010 11:19 AM	Application extens...	
TRAVERSE.Data.ARHistory.dll	7/30/2010 11:18 AM	Application extens...	
TRAVERSE.Data.ARTTransaction.dll	7/30/2010 11:19 AM	Application extens...	
TRAVERSE.Data.BankRec.dll	7/30/2010 11:21 AM	Application extens...	
TRAVERSE.Data.Batch.dll	7/30/2010 11:17 AM	Application extens...	
TRAVERSE.Data.BillMaterial.dll	7/30/2010 11:19 AM	Application extens...	
TRAVERSE.Data.BudgetSchema.dll	7/30/2010 11:17 AM	Application extens...	
TRAVERSE.Data.CashFlow.dll	7/30/2010 11:17 AM	Application extens...	
TRAVERSE.Data.Commission.dll	7/30/2010 11:21 AM	Application extens...	
TRAVERSE.Data.CommissionSetup.dll	7/30/2010 11:17 AM	Application extens...	

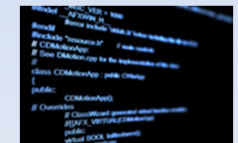


.NET and C# in a TRAVERSE world

TRAVERSE.exe.config file – stores settings such as sys database name, port, etc



```
</setting>
<setting name="LicensePort" serializeAs="String">
  <value>2113</value>
</setting>
<setting name="PluginProvider" serializeAs="String">
  <value/>
</setting>
<setting name="RemoteConnectionType" serializeAs="String">
  <value>0</value>
</setting>
<setting name="SysDbName" serializeAs="String">
  <value>SYSv11</value>
</setting>
<setting name="LocalizerAssembly" serializeAs="String">
  <value>TRAVERSE.Client.Localizer.dll</value>
</setting>
<setting name="LocalizerBaseName" serializeAs="String">
  <value>TRAVERSE.Client.LocalizedRes</value>
</setting>
<setting name="DataProvider" serializeAs="String">
  <value>.\DataProvider</value>
</setting>
<setting name="ReportPath" serializeAs="String">
```



.NET and C# in a TRAVERSE world

Assembly showing utilization of .NET base types (System.Windows.Forms)

