# .NET Assembly API Demos

THESE DEMOS WILL BE QUICK, BECAUSE ALL OF THE CODE IS AVAILABLE IN CODEEXCHANGE

## Sending An Email (Manual API)

* Program
  + Collects an email address
  + Calls SendReport(file, recipient)
* SendReport
  + Just talk through the code
* Can’t run this one, but thought you may find it useful!

# .NET Assembly API Demos

## SQL Server Reporting Client (Manual API)

* Program
  + Presents a list of available reports
  + User selects one
  + Calls ViewReport(ReportPath)
* ViewReport
  + Notice URL of SQL Server report server
  + Little more complex, this time we’re using 5 assemblies
  + First creates a WinForm
  + Then adds a ReportViewer control to the form
  + Identifies the report to show
  + Shows the form
  + This one has a call-back
    - Declares delegate
    - Adds delegate to event Form.Load
    - Event code calls reports RefreshReport method to execute report
* Run the demo

# .NET Assembly API Demos

## Building a WinForm (Manual API)

* This demo uses three assemblies and lots of different classes
* Builds and processes a Windows form
* Demo
  + Open assemblies
  + Create a windows form
  + Create various controls and add to form
  + Two event handlers
    - OK button click
    - TrackBar value changed
  + Synergy code is in Synergy class FormHandlers
  + Loads data into form controls
  + Processes form by calling “ShowDialog” method
  + Extracts data from form properties
* Run it
  + Change some data
  + Explain, synergy event handler updates text when TrackBar changes
  + Synergy event handler Closes form when OK button clicked.

# .NET Assembly API Demos

## XSL Transformation (GENNET)

* This demo uses .NET framework classes to perform an XSL Transformation
  + Takes an XML data file
  + Merges it with an XSL template (defines target output)
  + Produces output (often HTML, but doesn’t have to be)
* Review SalesData.xml
* Review SalesReportTemplate.xsl
* Review rebuild.bat
* Open a DOS box and execute REBUILD
* Review test program
  + Now using native .NET types, not @DotNetAssembly and @DotNetObject
  + Basically used just like in C# or VB
  + This synergy code is Synergy for .NET compatible
* Run

# .NET Assembly API Demos

## Building a WinForm (GENNET)

* This demo is again building a Windows form – VERY different approach
* Review BuildWrappers.bat
  + Don’t run it, it takes a while
  + Look at the PROTO folder – 1979 “entities” (mainly classes) generated!!!!
  + The OLB is 83 MB in size and contains over 60,000 routines!!!!!!
* Review FrmEmployee.dbl
  + This is Synergy code which uses the generated wrappers
    - VB or C# programmers might recognize the type of code
  + Just walk through the code
* Run demo