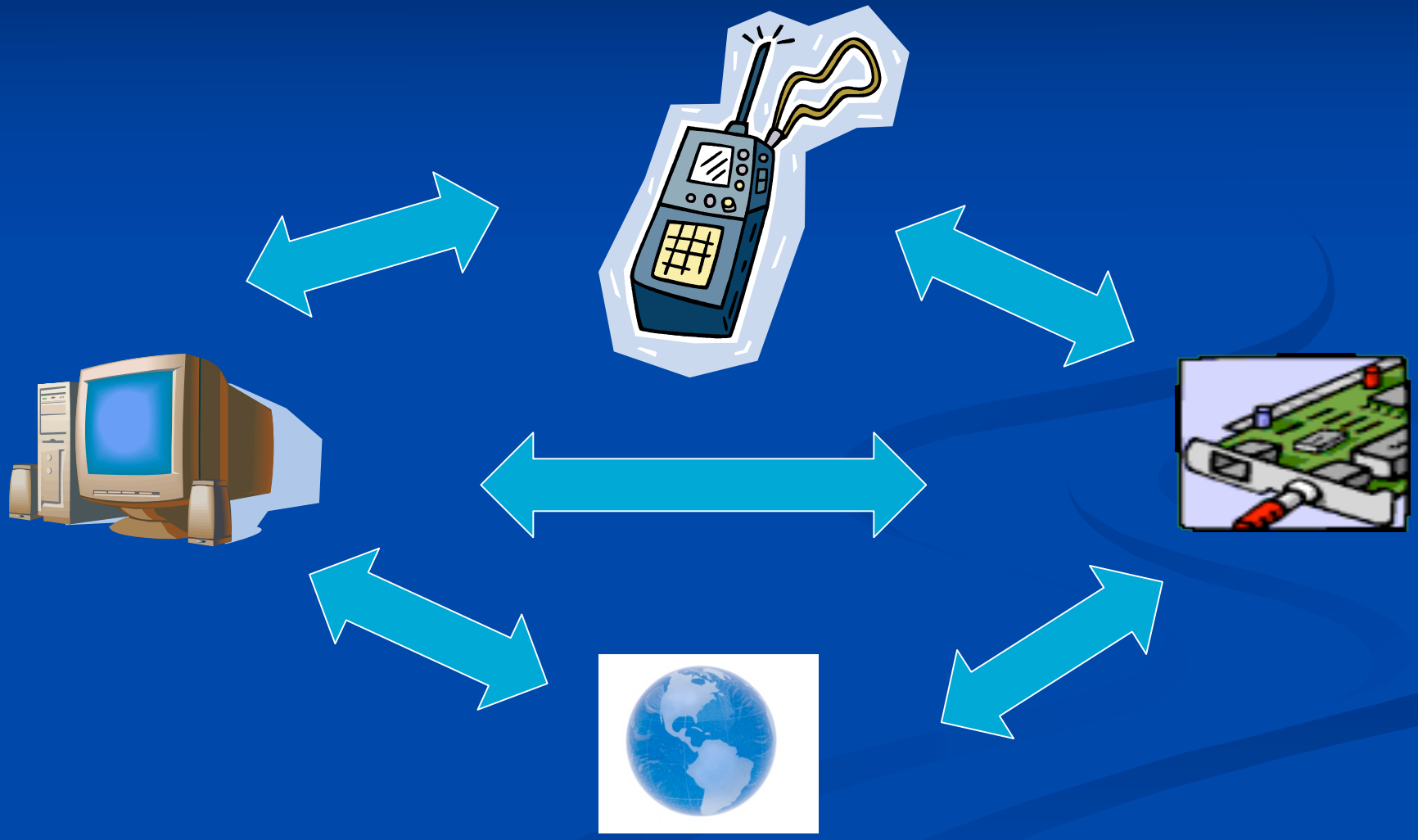
.Net API for Ham Radio

Dennis Harding- K7DEN

How?



For Today

- Mapping
  - MapPoint
  - Live Maps
- GPS Devices
- C#
- .Net
- Sorry, no LINUX

Mapping

A Look At Fundamentals

Find APIs

Contains Find related APIs

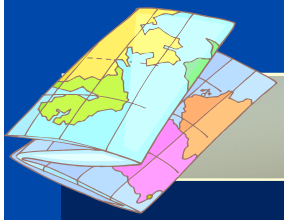
- Find Place
- Find Address
- Find Nearby

Route APIs

Contains Route related APIs

- Calculate Route
- Calculate Driving Direction

Map Control



Render APIs

Contains Render related APIs

- Render Maps Styles
- Render Pushpins
- Render Graphics

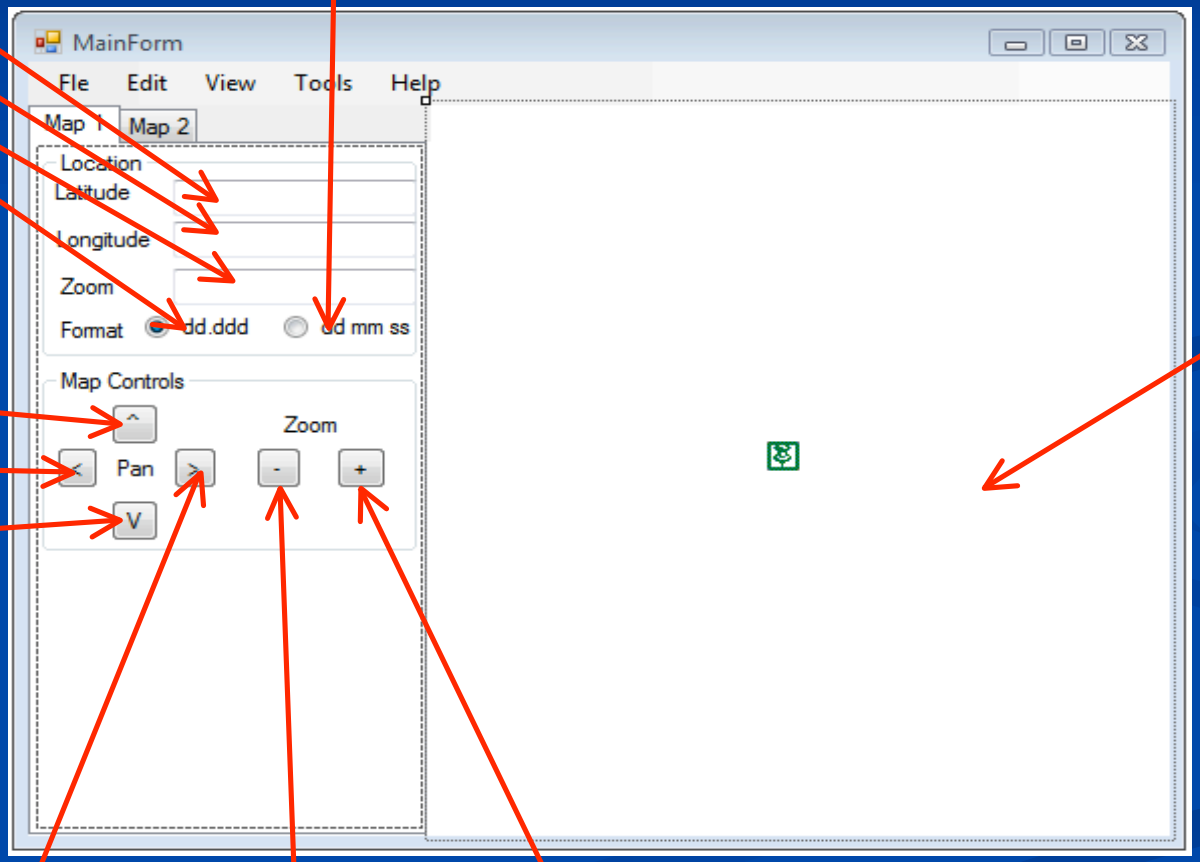
Layer APIs

Contains Layer based APIs

First Map Application

- Display a Map
- Show Latitude and Longitude of the center of the map
- Pan and Zoom
- Add a Push Pin at the center location

Basic Form


txtLatitude
txtLongitude
txtZoom
rbnDecimalDegree
rbnDegreeMinSec
MainForm
Fle
Edit
View
Tools
Help
Map 1
Map 2
Location
Latitude
Longitude
Zoom
Format
dd.ddd
dd mm ss
Map Controls
Zoom
Pan
-
+
V
MapPoint Control
Or
Web Control
btnScrollUp
btnScrollLeft
btnScrollDown
btnScrollRight
btnZoomIn
btnZoomOut

Adding the controls

- MapPoint
  - Step 1 – Add Reference to MapPoint Control Library
  - Step 2 – Add MapPoint Control the ToolBox
  - Step 3 – Add the control to the form
  - Step 4 – Initialize the control
- Live Maps
  - Step 1 – Add Web control
  - Create Map Web Page
    - Add the web page to the Project
    - Reference the VE Map Control
    - Scripts to control the map
- <browser>.Document
- HtmlDocument.InvokeScript()
- Windows.external.<function>

MapPoint Events

Event	Description
AfterRedraw	Occurs when MapPoint is finished repainting the map on the screen
AfterViewChange	Occurs when the view of the map has changed and the map is done repainting to that viewpoint
BeforeClick	Occurs after the user clicks on the map but before MapPoint has processed the action.
BeforeDblClick	Occurs after the user double-clicks on the map but before MapPoint has processed the action
DataMapChange	Occurs after data mapping properties are changed for a data set
MouseDown	Occurs when a mouse button is pressed while the pointer is over the map
MouseMove	Occurs when the mouse is moved while the pointer is over the map
MouseUp	Occurs when a mouse button is released while the pointer is over the map
NewDataSet	Occurs after a new data set is created
ReadyStateChange	This event occurs when the state of the MapPoint Control has changed
RouteAfterCalculate	Occurs after the route has been calculated
RouteAfterOptimize	Occurs after the stops on the route have been optimized
SelectionChange	Occurs when a selection on the map changes

Live Map Events

Event	Description
onchangeview	Occurs whenever the map view changes.
onclick	Occurs when the user clicks on the map.
oncontextmenu	Occurs when the user right-clicks on the map
onendcontinousspan	Occurs when a pan of the map ends.
onendzoom	Occurs when the map zoom ends.
onerror	Occurs when there is a map control error
onchangemapstyle	Occurs when the map style changes
onLoadMap	Occurs when the map is first loaded
onmouseup	Occurs when the user releases a mouse click on the map
onobliquechange	Occurs only when the bird's eye image scene ID is changed. This event fires only if the map is currently displaying a bird's eye image and that image is changed.
onobliqueenter	Occurs when switching to bird's eye imagery from another map style.
onobliqueleave	Occurs when switching from bird's eye imagery to another map style.
onresize	Occurs when the map is resized.
onstartcontinousspan	Occurs when a pan of the map begins.
onstartzoom	Occurs when the map zoom begins.

Thing you should know about MapPoint

- MapPoint and Streets and Tips use the same data. To update MapPoint use latest Streets and Tips data.
  - Program Files/ Microsoft Streets & Tips/Data => Program Files/Microsoft MapPoint/Data
- You change altitude to zoom the map.*
- Zoom effects Pan
- MapPoint uses a COM interface, you need to Start and Unload the MapPoint application*
- MapPoint Control Events
  - BeforeClick
  - BeforeDoubleClick

Things you should know about Live Maps

- Latest version is 6.0
- There were major changes between 4.0 and 5.0, making them incompatible
- More Info at
  - http://msdn2.microsoft.com/en-us/library/bb429619.aspx
  - http://dev.live.com/virtualearth/sdk/#
- IE 6/7 on XP requires - <html xmlns="http://www.w3.org/1999/xhtml">
- Page header defines map
- Body element contains the map
- Events are linked to script functions
  - VEMap.AttachEvent(event, function);

Panning and Zooming

MapPoint

- Zooming is controlled by setting altitude (in miles)
- Panning is controlled by the PanFactor, Direction.
  - PanFactor of 1 at 1mi altitude = 0.2 miles, at 50 mi = 10 miles
  - Direction is controeld by GeoPanCmd enumerator
  - For uniform panning, multiply altitude by 4.88568304395
- ActiveMap.Pan(direction, panFactor)

Live Maps

- Zooming in controlled by the zoom factor (value 1-19)
- Panning is controlled by the number of pixels to move
- VEMap.Pan(dX, dY)
- VEMap.PanToLatLong(VELatLong)
- VEMap.ZoomIn()
- VEMap.ZoomOut()
- VEMap.SetCenterAndZoom(VELatLong, zoom)

Things you should know about WebBrowser Control

- Adds a IE Browser to your app
- WebBrowser.Document -> HtmlDocument
- window.external.<method>
- WebBrowser.Document.InvokeScript()

MapPoint

MAPPOINT DEMO 1

Starting the Map

```
private void EnsureMapPointApplication()
      {
         centerPosition.IsMapPoint = true;
         if (this.app == null)
         {
axMappointControl1.NewMap(MapPoint.GeoMapRegion.geoMapNorthAmerica);
            app = axMappointControl1.ActiveMap.Application;
            // centerPosition contains the information for where to start the
MapMapPoint.Location location  =
                app.ActiveMap.GetLocation(centerPosition.Latitude,
                centerPosition.Longitude, centerPosition.Zoom);
            location.GoTo();
            centerPosition.UpdateLabels(txtLatitude, txtLongitude, txtZoom);
         }
      }
```

Stopping

```
protected override void Dispose(bool disposing)
    {
       if (app != null)
       {
          // note user will be asked to save the map.
          // to avoid this question, set the ..ActiveMap.Save = true
          // do not ask the user to save the map when terminating
          app.ActiveMap.Saved = true;
          app.Quit();
          app = null;
       }
                    ..........
    }
```

Finding the Center

```
private void locateCenter()
    {
      int cX, cY;
      cX = app.ActiveMap.Width / 2;
      cY = app.ActiveMap.Height / 2;
      MapPoint.Location location = app.ActiveMap.XYToLocation(cX, cY);
      centerPosition.SetPosition(location.Latitude, location.Longitude);
      centerPosition.UpdateLabels(txtLatitude, txtLongitude, txtZoom);
    }
```

MapPoint

LIVE MAP DEMO 1

HTML Header …

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
   <head>
      <title>Virtual Earth</title>
      <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
      <script type="text/javascript"
…….
src="http://dev.virtualearth.net/mapcontrol/mapcontrol.ashx?v=6"></script>
      <style type="text/css" media="screen">
</script>
   </head>
   <body style="margin: 0px">
   <div style="overflow: hidden" id="myMap"></div>
   </body>
</html>
```

Starting the Map

```
function startVE(width_a, height_a, lat_a, lon_a, zoom_a)
        {
          if (map == null)
          {
            map = new VEMap('myMap');
            map.LoadMap(new VELatLong(47.6, -122.33),
                    zoom_a, 'r', false);
            map.AttachEvent("onendcontinuouspan", update_map_position);
            map.AttachEvent("onchangeview", update_view_position);
            map.AttachEvent("onendzoom", update_map_zoom_level);
           map.AttachEvent("ondoubleclick", update_map_position);
            map.AttachEvent("onclick", scroll_to_mouse);
            window.external.scriptLoadCompleted();
          }
        }
```

Linking C# to Scripts

```
private const string ZoomInScript = "zoomIn";
private const string ZoomOutScript = "zoomOut";
private const string PanScript = "pan";
private const string RemovePushpinScript = "removePushpin";
private const string AddPushpinScript = "addPushpin";
private const string FindAddressLocation = "findAddress";
private const string FindWhatWhereLocation = "findLocation";
private const string SetMapStyleScript = "setMapStyle";
private const string ResizeScript = "resize";
private const string ClearPushpinsScript = "clearPushpins";
private const string StartVE = "startVE";
private const string SetCenterAndZoom = "setCenterAndZoom";
private const string SetCenterPosition = "setCenterPosition";
private const string SetZoomLevel = "setZoom";
private void executeScript(string scriptName, params object[] parameters)
{    virtualEarthMapBrowser1.Document.InvokeScript(scriptName,
         parameters)  }
```

FINDING THINGS

Find Function

MapPoint

- FindPlaceResult(name)
- FindAddressResult(street,city ,othercity, region, postalcode, country)
- FindNearby(distance)
- FindResult(name)
- FindPushPin(name)

Live Maps

- VEMap.Find(what, where, type, index, num results, showResults, createResults, useDefaultDisambigous, setBestView,callback)
- Callback – receives 5 parameters
  - callback(shapeLayer, findResult, place, more, error)

Find/Geo-Coding

- Available functions.
  - VEMap.Find(what, where, type, index, num results, showResults, createResults, useDefaultDisambigous, setBestView,callback);
- VEMap.Find() returns a results object and map for all.
- To get geo-coded results use map.GetCenter()
- CallBack – callback(shapeLayer, findResult, place, more, error)

Second Map Application

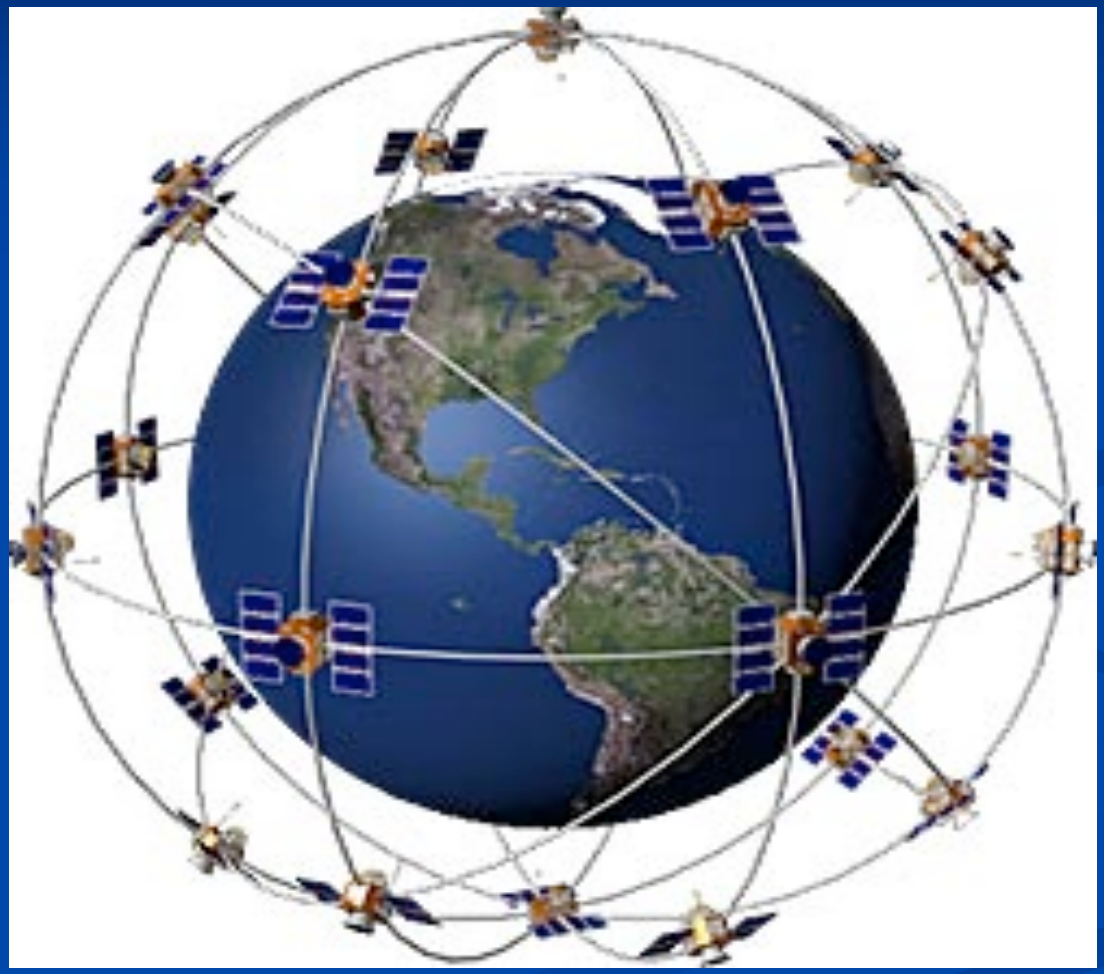
- Map a set of locations
  - Map a file of Latitude and Longitude
  - Map a file of addresses
  - Exporting Pin Locations
  - Map a set of findu locations

MapPoint

LIVE MAP DEMO2

GPS DEVICE

GPS System



GPS Devices

- USB Serial Devices
  - Use the .Net SerialPort
  - Devices usually are
    - Low Power Consumption
    - 8 parallel satellite-tracking channels for fast acquisition and reacquisition
    - Support for true NMEA-0183 data protocol
    - Enhanced algorithms provide superior tracking performance in urban, canyon, and foliage environments
    - Maximum navigation accuracy achievable with the Standard Positioning Service (SPS)
    - Meets rigid shock and vibration requirements
    - Automatic altitude hold mode from three-dimensional to two-dimensional navigation
    - Automatic cold start acquisition process
    - Built-in Antenna

NEMA Sentences Type

Type	Description
GPAPB	Auto Pilot B
GPBOD	bearing, origin to destination - earlier G-12's do not transmit this
GPGGA	fix data
GPGLL	Lat/Lon data - earlier G-12's do not transmit this
GPGSA	overall satellite reception data, missing on some Garmin models
GPGSV	detailed satellite data, missing on some Garmin models
GPRMB	minimum recommended data when following a route
GPRMC	minimum recommended data
GPRTE	route data, only when there is an active route. (this is sometimes bidirectional)
GPWPL	waypoint data, only when there is an active route (this is sometimes bidirectional)

NEMA Sentence

$GPGGA,213326.000,5343.3606,N,00641.2233,W,1,05,3.9,88.0,M,055,M,,*69
$GPGSA,A,3,07,20,25,04,11,,,,,,,,4.7,3.9,2.5*3A
$GPGSV,2,1,08,07,47,283,37,11,50,124,33,14,13,035,,20,84,203,35*71
$GPGSV,2,2,08,01,28,170,,28,06,243,,25,13,080,34,04,12,290,36*76
$GPRMC,213326.000,A,5343.3606,N,00641.2233,W,0.0,0.0,291203,08.5,W*58

Position Sentence (GPGGA)

$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47

Where:

- GGA — Global Positioning System Fix Data
- 123519 — Fix taken at 12:35:19 UTC
- 4807.038,N — Latitude 48 deg 07.038' N
- 01131.000,E — Longitude 11 deg 31.000' E
- 1 — Fix quality: 0 = invalid
  - 1 = GPS fix (SPS)
  - 2 = DGPS fix
  - 3 = PPS fix
  - 4 = Real Time Kinematic
  - 5 = Float RTK
  - 6 = estimated (dead reckoning) (2.3 feature)
  - 7 = Manual input mode
  - 8 = Simulation mode
- 08 — Number of satellites being tracked
- 0.9 — Horizontal dilution of position
- 545.4,M — Altitude, Meters, above mean sea level
- 46.9,M — Height of geoid (mean sea level) above WGS84 ellipsoid
- (empty field) time in seconds since last DGPS update
- (empty field) DGPS station ID number
- *47 — the checksum data, always begins with *

Converting Lat/Lon

- Format is ddmm.mmmmD -> need ±dd.dddddddd
  - Where D is N = North, S = South, E = East, W = West
- x = ((int)(x / 100.0)) +

  ((x - (((int)(x / 100.0)) * 100)) / 60);

GPS RECEIVER DEMO

References

- Slide 123
  - http://www.ku7m.net
- Slide 456
  - http://www.arrl.org

Resources

- Check out Virtual Earth for yourself: www.virtualearth.com
- Interactive SDK: http://dev.live.com/virtualearth/default.aspx?app=virtual_earth
- Microsoft MapPoint Developer Center http://msdn.microsoft.com/mappoint/
- Virtual Earth for Government Home Page http://www.microsoft.com/virtualearth/government/default.mspx

Resources

- Virtual Earth For Gov't Blog: http://virtualearth4gov.spaces.live.com/
- Extending an ESRI investment with Microsoft Virtual Earth http://msdn.microsoft.com/mappoint/mappointweb/mappointwstechart/default.aspx?pull=/library/en-us/dnmapnet30/html/ESRI_MWS.asp
- ViaVirtualEarth: Info for developers who want to integrate VE into their own applications
- Birds Eye Tourist: Catalog of nice BE maps

Resources

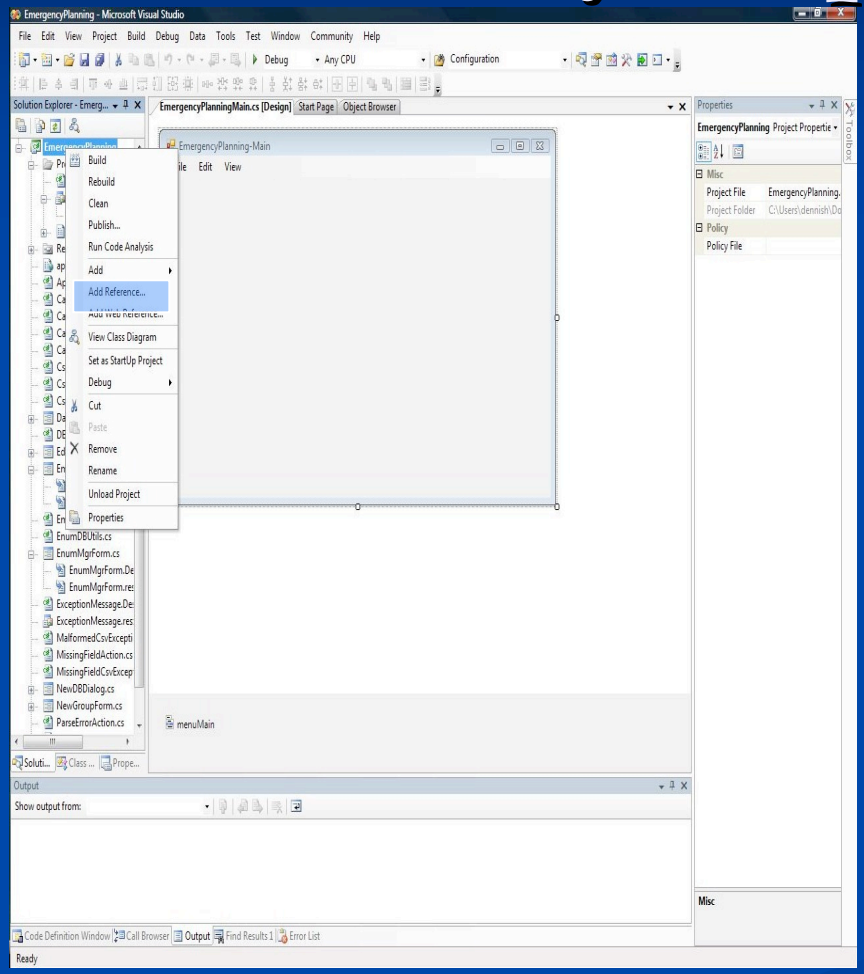
- Pass The POI: Share your WLL collections with the world or browse the creations of others
- maplic@microsoft.com to contact a Microsoft Virtual Earth Solution Specialist for more information and assistance
- http://www.viavirtualearth.com - Community site with great articles and examples, blogs, etc.

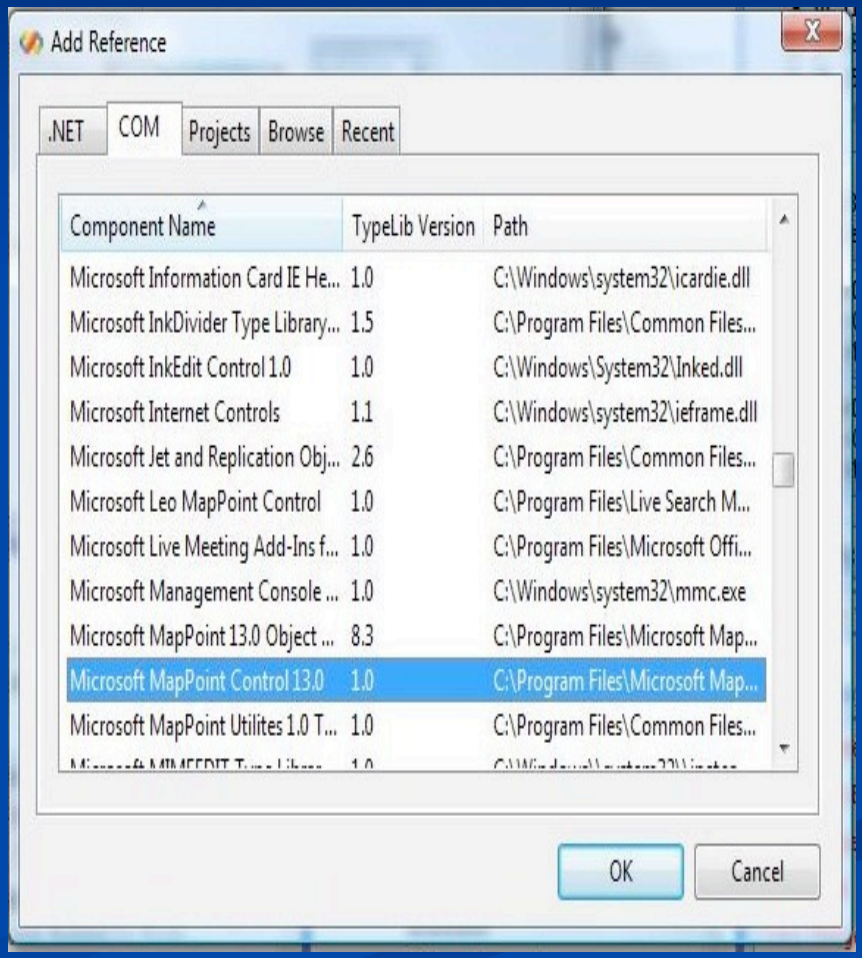
Resources

- blogs.msdn.com/virtualearth/ - A VE developer blog with updates and tips
- http://heptazane.spaces.live.com/ (David Buerer) - How to host the VE3D managed control in a desktop app (UNSUPPORTED)

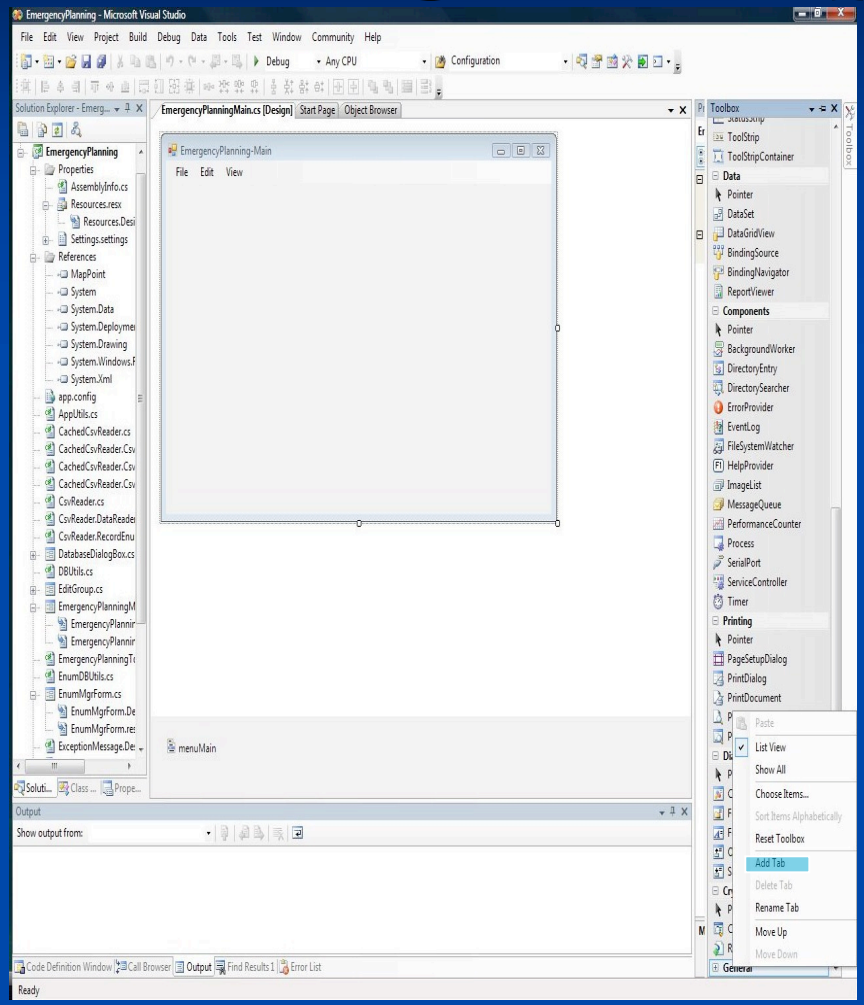

MicroHAMS
Where do you want to QSO today?

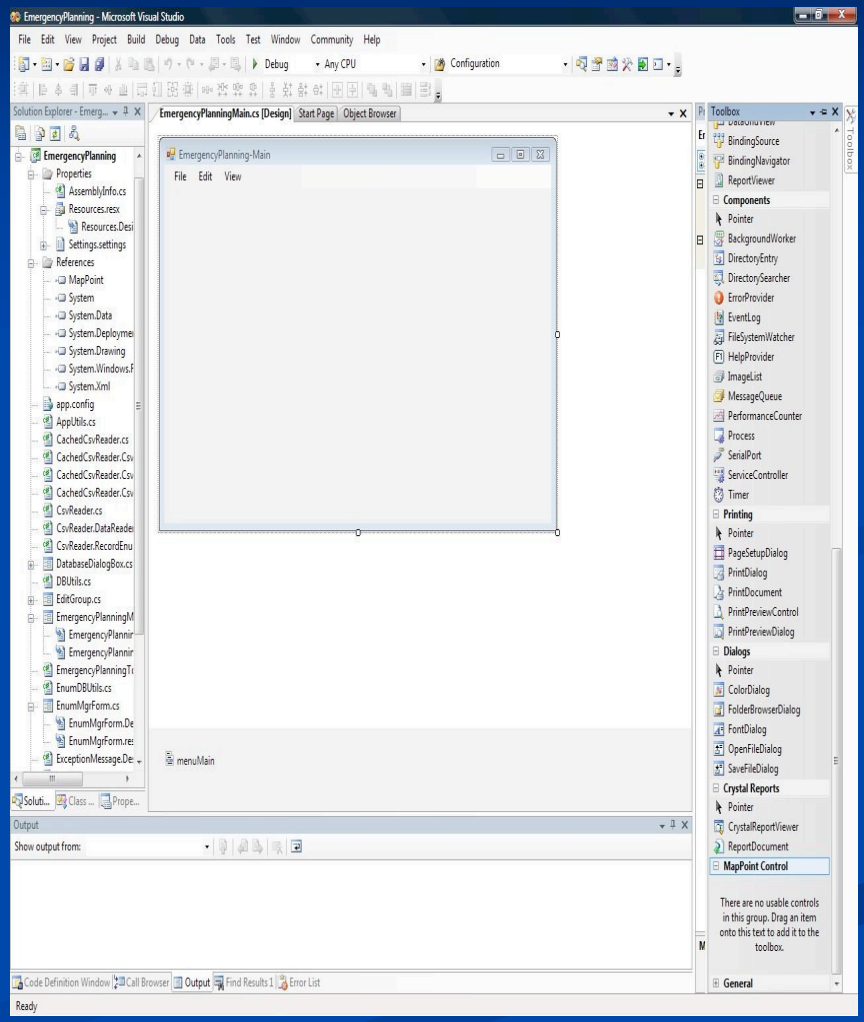
Adding MapPoint Library DLL to your project



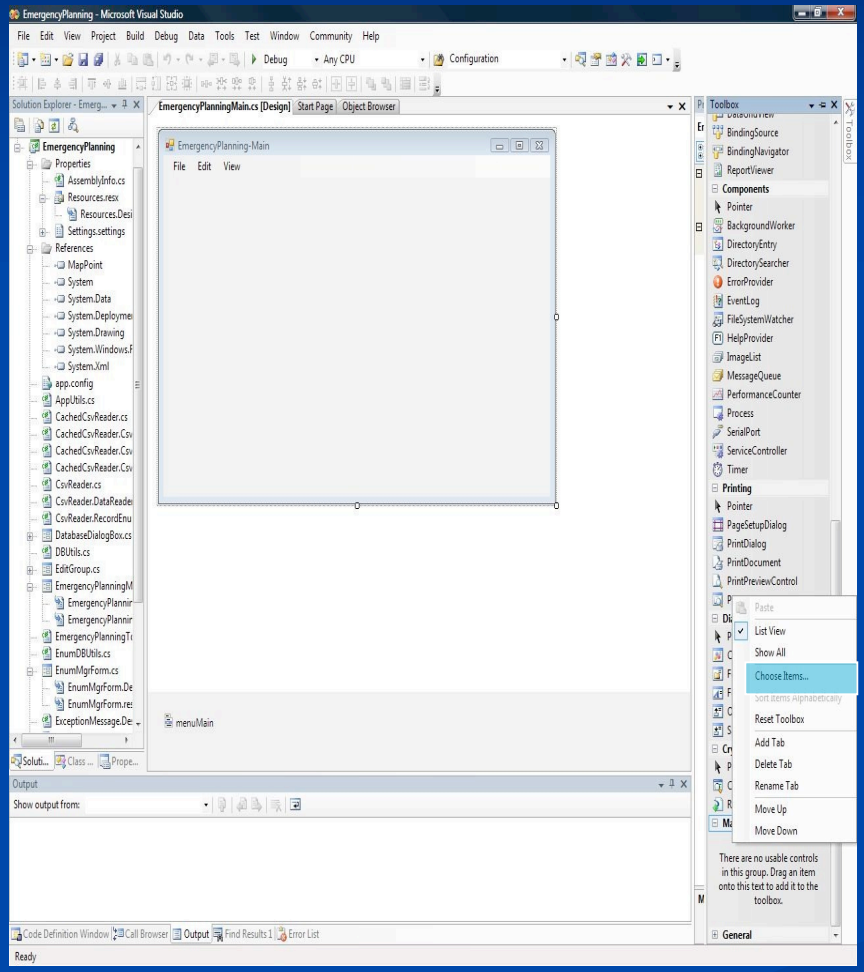


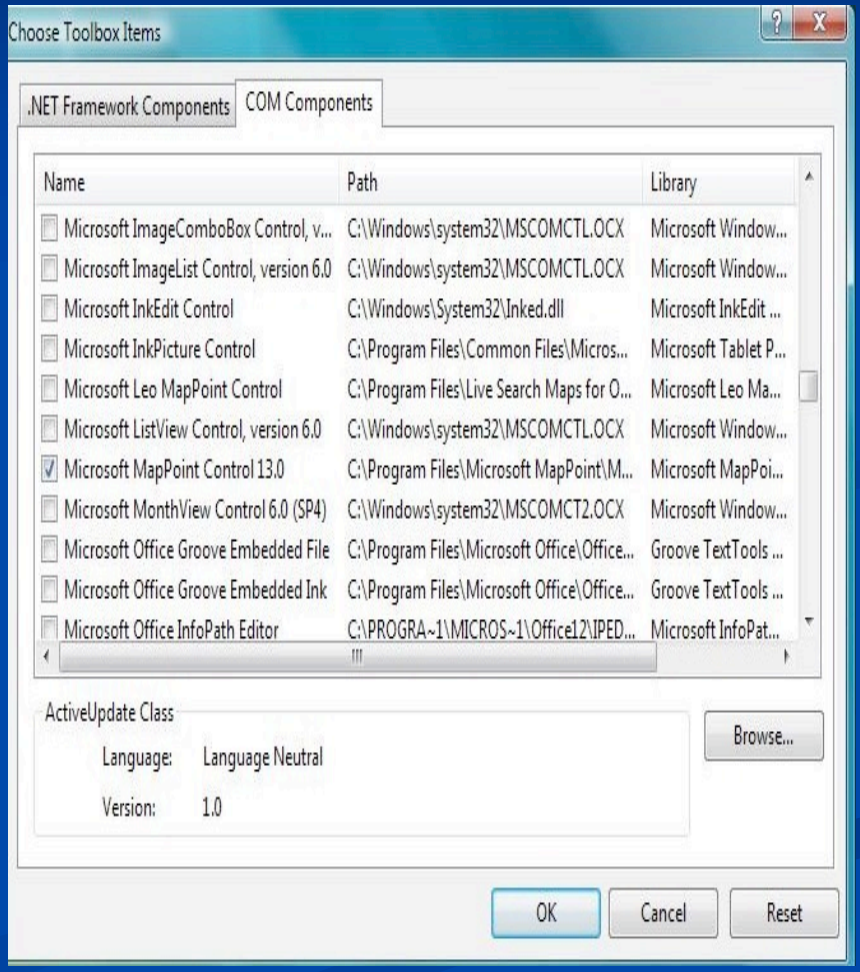
Adding a Group to the Toolbox





Adding the MapPoint Control to the tool box





Displaying a Live Map

- At the top of the HTML page add the following DOCTYPE declaration.
  - <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
- In the header section of an HTML page, add a META element with the **charset** attribute set to **"utf-8"**, as follows.
  - <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
- Also in the header section, add a reference to the map control, as follows.
  - <script type="text/javascript" src="http://dev.virtualearth.net/mapcontrol/mapcontrol.ashx?v=6"> </script>
- In the body of the page, add a DIV element to the page to contain the map. The size of the map is defined by the height and width of the DIV element. The position of the map is set by using the "position", "top", and "left" properties. You can set these values either inline or by defining the values in a style class and then referencing that class, as follows.
  -
- Create a new instance of the VEMap Class and call the VEMap.LoadMap Method, as follows.
  - var map = new VEMap('myMap'); map.LoadMap();