

GMap Landkarten Steuerelement unter VB2017

1 Installation

Installation Vers 1.7 alte Version (nur wenige Kartenanbieter, einige gehen nicht):

Dateien gmap.net.core.dll und gmap.net.windowsforms.dll downloaden und in Projektordner Debug einfügen
Steuerelement suchen und auf gmap.net.windowsforms.dll einstellen, Steuerelement ist verfügbar

Installation Vers 2.02.neue Version:

in Projektmanager auf Verweise mit re Maus > NuGet Pakete verwalten > suche nach GMap > gefundene Pakete:

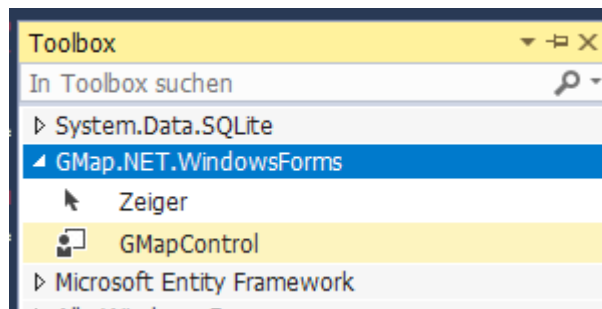
GMap.Net.Core Vers 2.02

GMap.Net.Windows Vers 2.02

GMap.Net.WinForms Vers 2.02

installieren.

Projekt neu starten, es wird in Tools eine neue Gruppe mit Steuerelement angezeigt



GMapControl kann jetzt verwendet werden!

2 Verwendung

eine Form erstellen und Steuerelement aus Toolbox einfügen

Im Eigenschaftsfenster die Eigenschaften festlegen bzw. ändern:

einen Button zum Karte laden und Checkbox für online oder offline

Eigenschaften:

MaxZoom	18	← Max Zoom max. Kartenvergrößerung
▣ MinimumSize	0; 0	
MinZoom	4	← Min Zoom min Kartenansicht (4 ist Ländergröße sichtbar)
Modifiers	Friend	
MouseWheelZoomE	True	
MouseWheelZoomT	MousePositionAn	
NegativeMode	False	
▣ Padding	0; 0; 0; 0	
PolygonsEnabled	True	
RightToLeft	No	
RoutesEnabled	True	
ScaleMode	Integer	
SelectedAreaFillColor	<input type="checkbox"/> 33; 65; 105; :	
ShowTileGridLines	False	
▣ Size	1092; 594	
TabIndex	6	
TabStop	True	
Tag		
ToolTip auf ToolTip	"Alt" u. Bereich Zi	
UseWaitCursor	False	
Visible	True	
Zoom	4	← beim Start angezeigter Zoom

Schaltfläche zum Karten laden

Kartenanbieter auswählbar

```
Private Sub Button6_Click(sender As Object, e As EventArgs) Handles Button6.Click
    GMapControl1.MapProvider = GMap.NET.MapProviders.OpenStreetMapProvider.Instance 'OpenStreetMap Höhen Geschäfte W-Wege

    'Auswahl online offline
    If RadioButton1.Checked = True Then GMap.NET.GMaps.Instance.Mode = GMap.NET.AccessMode.ServerAndCache 'online
    If RadioButton2.Checked = True Then GMap.NET.GMaps.Instance.Mode = GMap.NET.AccessMode.CacheOnly 'offline

    'Cache wird in C:\Users\Administrator\AppData\Local\GMap.NET\TileDBv5\en gespeichert Datei: Data.gmdb ist Standardeinstellung

    GMapControl1.CacheLocation = Application.StartupPath & "\\Karten\" & cachespeicherort 'Cache Pfad
    GMap.NET.GMaps.Instance.ImportFromGMDB(Application.StartupPath & "\\Karten\" & cachespeicherort) 'Pfad zum laden des Cache

    GMapControl1.MarkersEnabled = True 'Marker sichtbar
    GMapControl1.Position = New GMap.NET.PointLatLng(50.917, 10.0) 'Kartenmittelpunkt auf Deutschland

End Sub
```

Die angezeigte Karte kann mit re Maus verschoben werden und mit Scrollrad der Zoom geändert werden.
Mit „Alt“ und einen Bereich auf der Karte ziehen wird der Bereich im Cache gespeichert und ist auch offline verfügbar!

3 Koordinaten an Mauszeiger ausgeben

bei einem Doppelklick auf die Karte kann die Position ermittelt werden z.B in einem Textfeld zur Anzeige
Dazu die Prozedur `GMapControl1_MouseDoubleClick` aufrufen (Eigenschaften Gmapcontrol1 und auf den Blitz klicken, dort sind alle verfügbaren Prozeduren aufgelistet).

Zu beachten ist die Orientierung Nord und Ost sind positive Koordinaten, Süd und West sind negative Koordinaten.

```

Private Sub GMapControl1_MouseDoubleClick(sender As Object, e As MouseEventArgs) Handles GMapControl1.MouseDoubleClick
    'Koordinaten an Maus in Labelfeld
    Dim lng As Double
    Dim lat As Double

    lat = GMapControl1.FromLocalToLatLng(e.X, e.Y).Lat
    lng = GMapControl1.FromLocalToLatLng(e.X, e.Y).Lng

    TextBox3.Text = Format(lat, "00.000000")
    TextBox4.Text = Format(lng, "000.000000")

End Sub

```

eine Erweiterung wäre bei einer angehakten Checkbox einen Marker an der Position zu erstellen:
dafür der Code:

```

Private Sub GMapControl1_MouseDoubleClick(sender As Object, e As MouseEventArgs) Handles GMapControl1.MouseDoubleClick
    'Koordinaten an Maus in Labelfeld
    Dim lng As Double
    Dim lat As Double

    lat = GMapControl1.FromLocalToLatLng(e.X, e.Y).Lat
    lng = GMapControl1.FromLocalToLatLng(e.X, e.Y).Lng

    TextBox3.Text = Format(lat, "00.000000")
    TextBox4.Text = Format(lng, "000.000000")

    If CheckBox1.CheckState = 1 Then
        'Marker setzen
        Dim tooltip As String

        Dim markers As GMap.NET.WindowsForms.GMapOverlay = New GMap.NET.WindowsForms.GMapOverlay("markers")
        Dim m As New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New GMap.NET.PointLatLng(lat, lng),
GMap.NET.WindowsForms.Markers.GMarkerGoogleType.green_pushpin)
        GMapControl1.Overlays.Add(markers)
        tooltip = InputBox("Markertext") 'Tooltext eingeben
        If Trim(tooltip) <> "" Then

            m.ToolTipText = tooltip 'Tooltip Text

```

```

m.ToolTip.Fill = Brushes.Beige 'Hintergrund
m.ToolTip.Foreground = Brushes.Black 'Schriftfarbe
m.ToolTip.TextPadding = New Size(15, 15) 'GRÖÙe
m.ToolTip.Stroke = Pens.Blue 'Umrandung d. Textfeldes
End If

```

```

m.Tag = marktag + 1 'Marker Eintrag Tag
markers.Id = markid + 1 'Marker Eintrag ID

```

```

markers.Markers.Add(m) 'Marker setzen

```

```

marktag = marktag + 1 'Zähler hoch setzen
markid = markid + 1 'Zähler hochsetzen

```

```

End If
End Sub

```

die Variable marktag und markid sind public deklariert und dienen zur eindeutigen Kennzeichnung der Marker für das Löschen. Löschen wird später besprochen.

4 Marker aus vorgegebenen Koordinaten setzen

Gebraucht wird ein Button zum setzen des Markers und zwei Textfelder für die Eingabe der Koordinaten und zwei Comboboxen für die Ausrichtung N,S bzw. E,W

Code:

```

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    'Marker aus Koordinate erstellen
    Dim breite1 As Double 'N S
    Dim laenge1 As Double 'E W

    'Werte prüfen
    If Trim(TextBox1.Text) = "" Or Trim(TextBox2.Text) = "" Then MsgBox("falscher Wert", 0, "Achtung") : Exit Sub

```

```

breite1 = TextBox1.Text
laenge1 = TextBox2.Text

'N oder S bearbeiten
If ComboBox2.Text = "N" Then breite1 = breite1 * 1
If ComboBox2.Text = "S" Then breite1 = breite1 * -1
If ComboBox3.Text = "E" Then laenge1 = laenge1 * 1
If ComboBox3.Text = "W" Then laenge1 = laenge1 * -1

'Marker setzen
Dim tooltext As String

Dim markers As GMap.NET.WindowsForms.GMapOverlay = New GMap.NET.WindowsForms.GMapOverlay("markers")
Dim m As New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New GMap.NET.PointLatLng(breite1, laenge1),
GMap.NET.WindowsForms.Markers.GMarkerGoogleType.blue_pushpin)
GMapControl1.Overlays.Add(markers)
tooltext = InputBox("Markertext") 'Tooltip eingeben
If Trim(tooltext) <> "" Then

    m.ToolTipText = tooltext 'Tooltip Text
    m.ToolTip.Fill = Brushes.Beige 'Hintergrund
    m.ToolTip.Foreground = Brushes.Black 'Schriftfarbe
    m.ToolTip.TextPadding = New Size(15, 15) 'GRöße
    m.ToolTip.Stroke = Pens.Blue 'Umrandung d. Textfeldes
End If

m.Tag = marktag + 1 'Marker Eintrag Tag
markers.Id = markid + 1 'Marker Eintrag ID

markers.Markers.Add(m) 'Marker setzen

marktag = marktag + 1 'Zähler hoch setzen
markid = markid + 1 'Zähler hochsetzen

TextBox1.Text = ""
TextBox2.Text = ""


End Sub

```

Die Art des Markers wird in der Zeile:

```
Dim m As New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New GMap.NET.PointLatLng(lat, lng),  
GMap.NET.WindowsForms.Markers.GMarkerGoogleType.green_pushpin)
```

auswählbar von den Google Markern



bestimmt

es können auch individuelle Markerbilder verwendet werden.

```
Dim m As New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New GMap.NET.PointLatLng(lat, lng), New Bitmap("symb.gif")) 'Punkt
```

Unterstützt wird *.jpg, *.gif, *.png, *.ico als Bildformat.

5 alle Marker löschen

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    'alle Marker löschen
    GMapControl1.Overlays.Clear()      'alle Marker löschen
    GMapControl1.Refresh()             'aktualisieren
End Sub
```

6 einen bestimmten Marker löschen

Dazu muss eine Prozedur mit Namen `OnMarkerClick` aufgerufen werden, zu finden in Eigenschaften von GMap Steuerelement und dann auf den Blitz klicken.

```
Private Sub GMapControl1_OnMarkerClick(item As GMap.NET.WindowsForms.GMapMarker, e As MouseEventArgs) Handles GMapControl1.OnMarkerClick
    'löschen ausgewählter Marker

    'Dim marktaga As String = item.Tag      'ermittelt den Tag des ausgewählten Markers wird nicht verwendet
    Dim markid As String = item.Overlay.Id 'ermittelt ID von ausgewählten Marker
    Dim element

    For i = 0 To (GMapControl1.Overlays.Count) - 1 'Schleife für alle Markers
        element = GMapControl1.Overlays.Item(i)    'Element mit Index i

        If element.id = markid Then                'von ausgewählten Marker die ID gleich ID in Schleife
            'GMapControl1.Overlays.RemoveAt(i)    'Marker mit Index löschen Problem wenn aus dieser Sub
            loeschindex = i
            GMapControl1.Refresh()                 'aktualisieren
            Exit For                               'Schleifen Abbruch wenn gefunden
        End If
    Next

End Sub
```


Das eigentliche löschen erfolgt nicht in der Prozedur (ist mir nicht gelungen), es wird lediglich der Index des löschenden Markers ermittelt. Das Löschen erfolgt dann über eine Schaltfläche.

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    'ausgewählten Marker löschen
    If loeschindex < 0 Then
        MsgBox("kein Marker vorher ausgewählt", 0, "Achtung")
        Exit Sub
    End If
    GMapControl1.Overlays.RemoveAt(loeschindex) 'Marker mit Index löschen
    GMapControl1.Refresh() 'aktualisieren
    loeschindex = -1

End Sub
```

Also Marker anklicken(Finger als Maussymbol) und dann auf Schaltfläche klicken.

Die Variable **loeschindex** ist Public deklariert damit sie im Button und in der Auswahlprozedur verfügbar ist.

Problem: Wird gleich in der Prozedur gelöscht, wie angedacht, läßt sich die Karte anschließend nicht mehr Verschieben und Zoomen, deshalb der Umweg über eine separate Schaltfläche.

7 Löschen einer bestimmten Gruppe von Markern

Voraussetzung beim Setzen des Markers erhalten diese einen einheitlichen Eintrag in der marker.id (auch Text ist möglich)

das Löschen erfolgt wie folgt:

```
Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
    'löschen aller Pos Marker mit gleicher ID sie wird beim Erstellen festgelegt
    Dim element

    For i = GMapControl1.Overlays.Count - 1 To 0 Step -1 'Schleife für alle Markers
        element = GMapControl1.Overlays.Item(i) 'Element mit Index i
        If element.id = "Pos" Then 'von ausgewählten Marker die ID gleich ID in Schleife
```

```
        GMapControl1.Overlays.RemoveAt(i) 'Marker mit Index
        GMapControl1.Refresh()           'aktualisieren
    End If
Next
GMapControl1.Refresh()                 'aktualisieren
End Sub
```

8 Zwei Marker setzen und einen immer an letzter Position bringen

Man Marker an eine gefahrene Strecke im bestimmten Zeitintervall setzen. Die Positionen könnten von einer GPS Maus geliefert werden. Es ist dann sinnvoll die letzte Position besonders zu kennzeichnen.

Hier ein Beispiel:

In zwei Textfeldern wird die Position eingegeben (evt. gekoppelt an eine Datenherkunft von einer GPS Maus). Ausgelöst wird das setzen über einen Button, der wiederum über einen Timer automatisch mit einem Klick-Ereignis betätigt wird. Es entstehen auf der Karte Marker (als kleiner Punkt) die letzte Position wird zusätzlich mit einem Pfeil gekennzeichnet.

Code:

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
    'GPS Pos manuell setzen
    Dim tooltip As String
    Dim lat As Double
    Dim lng As Double

    'Werte für Breite und Länge prüfen
    If Trim(TextBox5.Text) = "" Or Trim(TextBox6.Text) = "" Then Exit Sub 'auf leer prüfen
    If IsNumeric(TextBox5.Text) = False Or IsNumeric(TextBox6.Text) = False Then Exit Sub 'auf Numeric prüfen

    lat = TextBox5.Text
    lng = TextBox6.Text
    If lat = alteBreite And lng = alteLaenge Then Exit Sub 'bei keiner Pos Änderung braucht kein Merker gesetzt zu werden

    Dim posmarkers As GMap.NET.WindowsForms.GMapOverlay = New GMap.NET.WindowsForms.GMapOverlay("pomarkers") 'Punkt
    Dim posmarkers2 As GMap.NET.WindowsForms.GMapOverlay = New GMap.NET.WindowsForms.GMapOverlay("pomarkersring") 'Pfeil
    Dim m As New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New GMap.NET.PointLatLng(lat, lng), New Bitmap("symb.gif")) 'Punkt
    Dim n As New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New GMap.NET.PointLatLng(lat, lng), New Bitmap("Pfeil.gif")) 'Pfeil

    GMapControl1.Overlays.Add(posmarkers)
    tooltip = "Position: " & lat & " - " & lng 'Tooltip eingeben
    m.ToolTipText = tooltip 'Tooltip Text
    m.ToolTip.Fill = Brushes.Beige 'Hintergrund
    m.ToolTip.Foreground = Brushes.Black 'Schriftfarbe
```

```
m.ToolTip.TextPadding = New Size(15, 15) 'GRöße
m.ToolTip.Stroke = Pens.Blue 'Umrandung d. Textfeldes
m.Tag = "Pos" 'Marker Eintrag Tag
posmarkers.Id = "Pos" 'Marker Eintrag ID über diese ID können alle Marker gleichzeitig gelöscht werden
posmarkers.Markers.Add(m) 'Marker setzen
```

```
'alten Pfeil löschen
'löschen Pos Marker Pfeil mit gleicher ID PosPfeil
Dim element
For i = GMapControl1.Overlays.Count - 1 To 0 Step -1 'Schleife für alle Markers
    element = GMapControl1.Overlays.Item(i) 'Element mit Index i
    If element.id = "PosPfeil" Then 'Marker die ID gleich PosPfeil
        GMapControl1.Overlays.RemoveAt(i) 'Marker mit Index
        GMapControl1.Refresh() 'aktualisieren
    End If
Next
GMapControl1.Refresh() 'aktualisieren
```

```
'Pfeil setzen
GMapControl1.Overlays.Add(posmarkers2)
posmarkers2.Id = "PosPfeil" 'Marker Eintrag ID über diese ID können alle Marker gleichzeitig gelöscht werden
posmarkers2.Markers.Add(n) 'Marker setzen als Ring
```

```
'alte Positionen speichern da bei keiner Bewegung auch kein neuer Marker gesetzt werden muss
alteBreite = lat
alteLaenge = lng
```

```
End Sub
```

Um zu vermeiden, dass bei Stillstand unnötig Marker gesetzt werden wird die alte Position gespeichert (Variable alteBreite und alteLaenge, beide Public deklariert) und nur bei Änderung der Position wird ein neuer Marker gesetzt.

9 Offline Nutzung der Karten

Die aufgerufenen Karten werden automatisch gecacht und sind auch im Offline Modus verfügbar.

Es gibt die Möglichkeit im Online Modus gezielt einen Bereich in den Cache-Ordner zu laden, indem man die „Alt“ Taste drückt und den gewünschten Bereich mit der Maus zieht. Der Download beginnt sofort. Leider ist es mir nicht gelungen den Verlauf bzw. das Ende des Downloads anzuzeigen.

Also Warten bis Festplatten LED sich beruhigt.

Der Standardordner für den Cache ist:

C:\Users\Administrator\AppData\Local\GMap.NET\TileDBv5\en gespeichert Datei: Data.gmdb

Man kann diesen Ort auch ändern und auch mehrere Dateien verwenden.

Beispiel für anderen Cache Speicherort in der Prozedur zum Laden der Karte:

```
GMapControl1.CacheLocation = Application.StartupPath & "\\Karten\" & cachespeicherort 'Cache Pfad
```

```
GMap.NET.GMaps.Instance.ImportFromGMDB(Application.StartupPath & "\\Karten\" & cachespeicherort) 'Pfad zum laden des Cache
```

Hier im Unterordner Karten im Projektordner. Wobei die Variable cachespeicherort einen weiteren Unterordner darstellt, der flexibel ausgewählt werden kann. In diesem Unterordner wird dann automatisch eine Dateistruktur (\TileDBv5\en\ Data.gmdb) angelegt.

In meinem Projekt arbeite ich mit mehreren Ordnern, die es erlaubt verschiedene Cache Gebiete zu verwalten.

Die Datei Größe der Cache-Datei *Data.gmdb* wird automatisch festgelegt.

Ein komplettes Projekt mit GPS Empfang kann unter der E-Mail: **info@schaefer-marksuhl.de** angefordert werden.