GMap.NET源代码结构

王喜进 于57s

2016.9

目录

[1 引言 3](#_Toc462219483)

[2 概述 3](#_Toc462219484)

[3 GMap.NET类结构设计 3](#_Toc462219485)

[4 GMap.NET代码结构解析 5](#_Toc462219486)

[4.1 GMap.NET.Core 5](#_Toc462219487)

[4.1.1 GMap.NET 5](#_Toc462219488)

[4.1.2 GMap.NET.CacheProviders 30](#_Toc462219489)

[4.1.3 GMap.NET.Internals 35](#_Toc462219490)

[4.1.4 GMap.NET.MapProviders 56](#_Toc462219491)

[4.1.5 GMap.NET.Projections 98](#_Toc462219492)

[4.2 GMap.NET.WindowsForms 105](#_Toc462219493)

[4.2.1 GMap.NET.ObjectModel 105](#_Toc462219494)

[4.2.2 GMap.NET.WindowsForms 106](#_Toc462219495)

[5 GMap.NET应用实例 124](#_Toc462219496)

[6 结束语 124](#_Toc462219497)

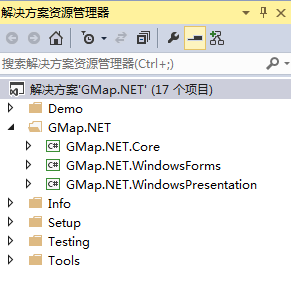
1. 引言

GMap.NET是专门为C#开发者提供的桌面地图控件，鉴于其功能强大，且完全开源，在实际项目中也有应用，所以决定对其源代码进行剖析，了解其所以然，这样不但可以对自己在GIS地里信息系统方面有所提升，并且希望能帮助一些其他同样正在研究GMap.NET的朋友，大家共同进步。

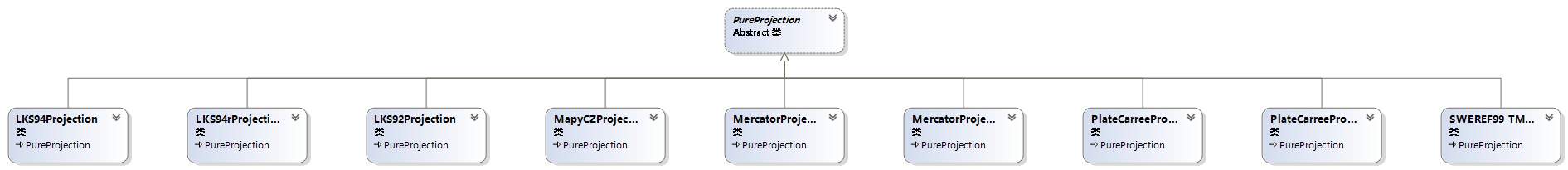
在这里，还是要对GMap.NET的开源作者表示感谢！

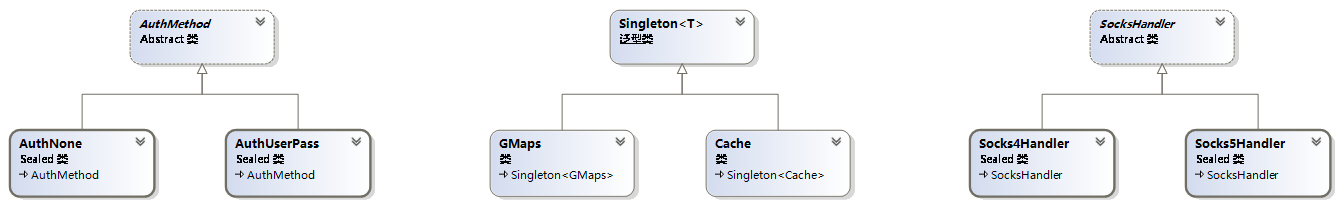
1. 概述

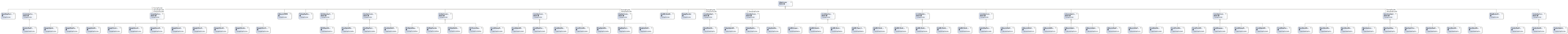
GMap.NET的解决方案总体结构如下图所示：



1. GMap.NET类结构设计
   1. GMap.NET.Core

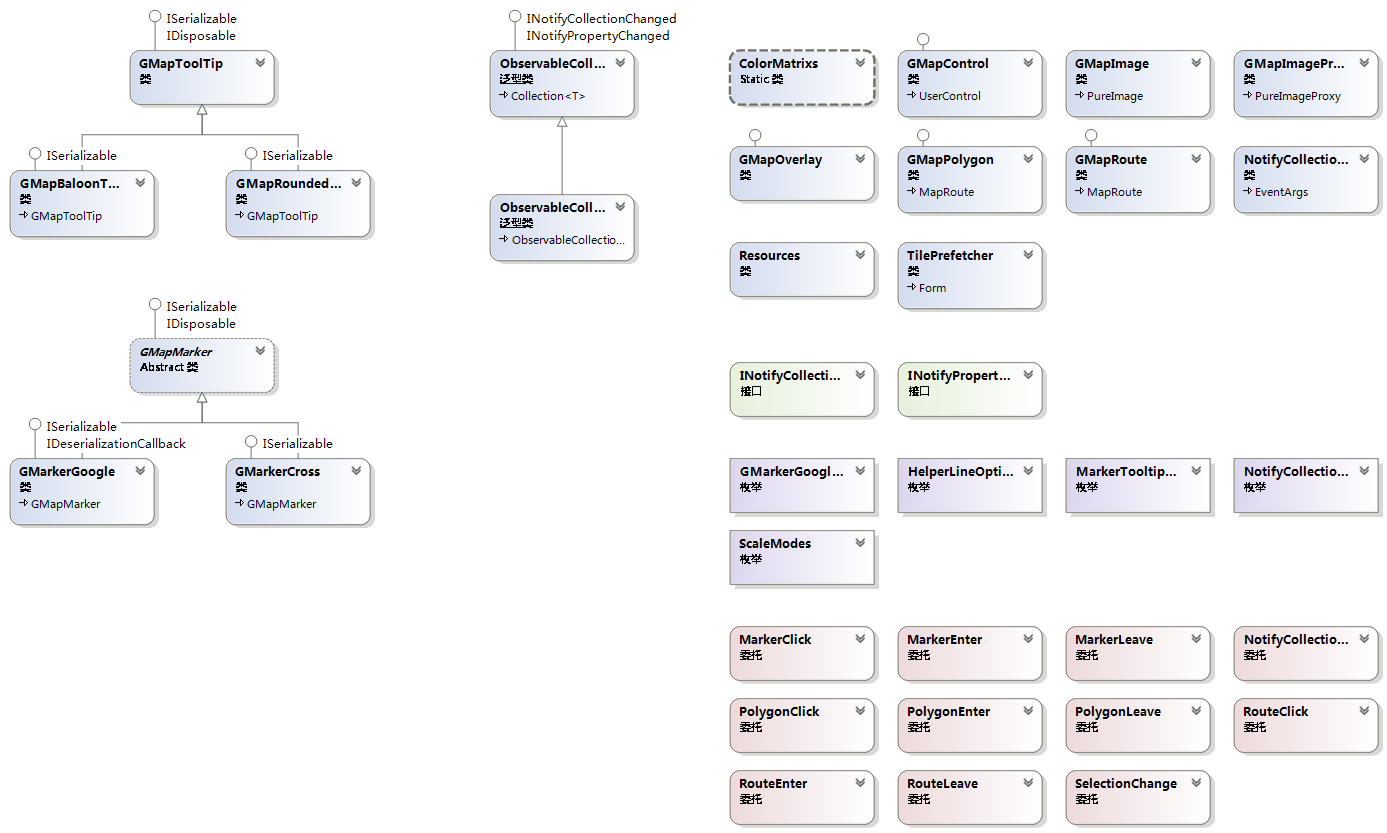








* 1. GMap.NET.WindowsForms



1. GMap.NET代码结构解析
   1. GMap.NET.Core
      1. GMap.NET
         1. AccessMode.cs
2. 概述

此cs文件为瓦片地图访问方式枚举类。

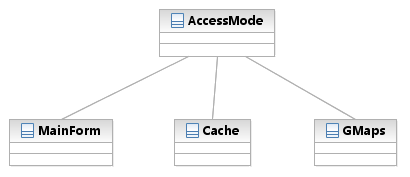
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// tile access mode  /// </summary>  public enum AccessMode  {  /// <summary>  /// access only server  /// </summary>  ServerOnly,  /// <summary>  /// access first server and caches localy  /// </summary>  ServerAndCache,  /// <summary>  /// access only cache  /// </summary>  CacheOnly,  } |

1. 类的方法定义

无。

* + - 1. Delegates.cs

1. 概述

此cs文件为定义全局委托事件。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| namespace GMap.NET  {  using GMap.NET.MapProviders;  public delegate void PositionChanged(PointLatLng point);  public delegate void TileLoadComplete(long ElapsedMilliseconds);  public delegate void TileLoadStart();  public delegate void TileCacheComplete();  public delegate void TileCacheStart();  public delegate void TileCacheProgress(int tilesLeft);  public delegate void MapDrag();  public delegate void MapZoomChanged();  public delegate void MapTypeChanged(GMapProvider type);  public delegate void EmptyTileError(int zoom, GPoint pos);  } |

1. 类的方法定义

无。

* + - 1. DirectionsProvider.cs

1. 概述

此cs文件为方向接口的定义。

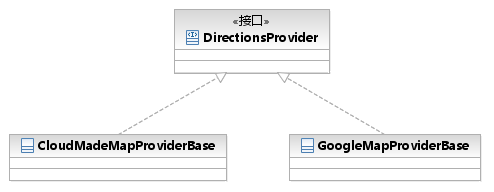
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| /// <summary>  /// directions interface  /// </summary>  interface DirectionsProvider  {  DirectionsStatusCode GetDirections(out GDirections direction, PointLatLng start, PointLatLng end, bool avoidHighways, bool avoidTolls, bool walkingMode, bool sensor, bool metric);  DirectionsStatusCode GetDirections(out GDirections direction, string start, string end, bool avoidHighways, bool avoidTolls, bool walkingMode, bool sensor, bool metric);  /// <summary>  /// service may provide more than one route alternative in the response  /// </summary>  /// <param name="status"></param>  /// <param name="start"></param>  /// <param name="end"></param>  /// <param name="avoidHighways"></param>  /// <param name="avoidTolls"></param>  /// <param name="walkingMode"></param>  /// <param name="sensor"></param>  /// <param name="metric"></param>  /// <returns></returns>  IEnumerable<GDirections> GetDirections(out DirectionsStatusCode status, string start, string end, bool avoidHighways, bool avoidTolls, bool walkingMode, bool sensor, bool metric);  /// <summary>  /// service may provide more than one route alternative in the response  /// </summary>  /// <param name="status"></param>  /// <param name="start"></param>  /// <param name="end"></param>  /// <param name="avoidHighways"></param>  /// <param name="avoidTolls"></param>  /// <param name="walkingMode"></param>  /// <param name="sensor"></param>  /// <param name="metric"></param>  /// <returns></returns>  IEnumerable<GDirections> GetDirections(out DirectionsStatusCode status, PointLatLng start, PointLatLng end, bool avoidHighways, bool avoidTolls, bool walkingMode, bool sensor, bool metric);  DirectionsStatusCode GetDirections(out GDirections direction, PointLatLng start, IEnumerable<PointLatLng> wayPoints, PointLatLng end, bool avoidHighways, bool avoidTolls, bool walkingMode, bool sensor, bool metric);  DirectionsStatusCode GetDirections(out GDirections direction, string start, IEnumerable<string> wayPoints, string end, bool avoidHighways, bool avoidTolls, bool walkingMode, bool sensor, bool metric);  } |

* + - 1. Extensions.cs

1. 概述

此cs文件为全局静态类，提供一些全局方法。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| public static T GetValue<T>(SerializationInfo info, string key) where T : class  public static T GetValue<T>(SerializationInfo info, string key, T defaultValue) where T : class  public static T GetStruct<T>(SerializationInfo info, string key, T defaultValue) where T : struct  public static Nullable<T> GetStruct<T>(SerializationInfo info, string key, Nullable<T> defaultValue) where T : struct |

* + - 1. GDirections.cs

1. 概述

此cs文件定义了GDirections和GDrectionsStep类，GDirections类为路径对象类，GDirectionsStep为路径步进对象类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| /// <summary>  /// contains a short textual description for the route, suitable for naming and disambiguating the route from alternatives.  /// </summary>  public string Summary;  /// <summary>  /// contains a human-readable representation of the duration.  /// </summary>  public string Duration;  /// <summary>  /// contains a value of the duration.  /// </summary>  public uint DurationValue;  /// <summary>  /// contains a human-readable representation of the distance, displayed in units as used at the origin  /// (or as overridden within the units parameter in the request), in the language specified in the request.  /// (For example, miles and feet will be used for any origin within the United States.)  /// </summary>  public string Distance;  /// <summary>  /// contains a value of the distance.  /// </summary>  public uint DistanceValue;  /// <summary>  /// contains the latitude/longitude coordinates of the origin of this leg. Because the Directions API  /// calculates directions between locations by using the nearest transportation option (usually a road)  /// at the start and end points, start\_location may be different than the provided origin of this leg if,  /// for example, a road is not near the origin.  /// </summary>  public PointLatLng StartLocation;  /// <summary>  /// contains the latitude/longitude coordinates of the given destination of this leg. Because the Directions  /// API calculates directions between locations by using the nearest transportation option (usually a road)  /// at the start and end points, end\_location may be different than the provided destination of this leg if,  /// for example, a road is not near the destination.  /// </summary>  public PointLatLng EndLocation;  /// <summary>  /// contains the human-readable address (typically a street address) reflecting the start\_location of this leg.  /// </summary>  public string StartAddress;  /// <summary>  /// contains the human-readable address (typically a street address) reflecting the end\_location of this leg.  /// </summary>  public string EndAddress;  /// <summary>  /// contains the copyrights text to be displayed for this route. You must handle and display this information yourself.  /// </summary>  public string Copyrights;  /// <summary>  /// contains an array of steps denoting information about each separate step of the leg of the journey.  /// </summary>  public List<GDirectionStep> Steps;  /// <summary>  /// contains all points of the route  /// </summary>  public List<PointLatLng> Route; |

1. 类的方法定义

|  |
| --- |
| public override string ToString()  {  return Summary + " | " + Distance + " | " + Duration;  } |

* + - 1. GeocodingProvider.cs

1. 概述

此cs文件为地理编码提供者接口。

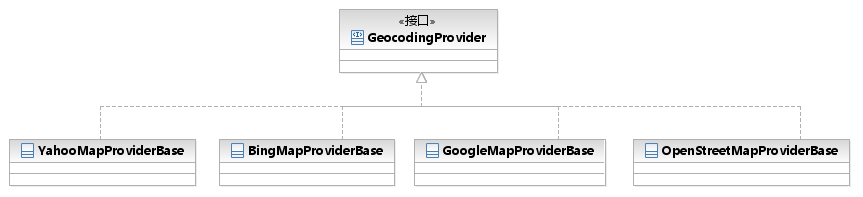
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| /// <summary>  /// geocoding interface  /// </summary>  public interface GeocodingProvider  {  GeoCoderStatusCode GetPoints(string keywords, out List<PointLatLng> pointList);  PointLatLng? GetPoint(string keywords, out GeoCoderStatusCode status);  GeoCoderStatusCode GetPoints(Placemark placemark, out List<PointLatLng> pointList);  PointLatLng? GetPoint(Placemark placemark, out GeoCoderStatusCode status);  // ...  GeoCoderStatusCode GetPlacemarks(PointLatLng location, out List<Placemark> placemarkList);  Placemark ? GetPlacemark(PointLatLng location, out GeoCoderStatusCode status);  } |

* + - 1. GMaps.cs

1. 概述

此cs文件为地图框架的功能提供类，主要是对Cache、地图切片、地图数据导入导出等。

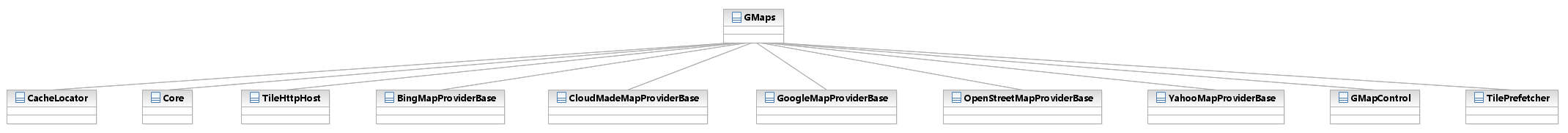
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// tile access mode  /// </summary>  public AccessMode Mode = AccessMode.ServerAndCache;  /// <summary>  /// is map ussing cache for routing  /// </summary>  public bool UseRouteCache = true;  /// <summary>  /// is map using cache for geocoder  /// </summary>  public bool UseGeocoderCache = true;  /// <summary>  /// is map using cache for directions  /// </summary>  public bool UseDirectionsCache = true;  /// <summary>  /// is map using cache for placemarks  /// </summary>  public bool UsePlacemarkCache = true;  /// <summary>  /// is map ussing cache for other url  /// </summary>  public bool UseUrlCache = true;  /// <summary>  /// is map using memory cache for tiles  /// </summary>  public bool UseMemoryCache = true;  /// <summary>  /// primary cache provider, by default: ultra fast SQLite!  /// </summary>  public PureImageCache PrimaryCache  {  get  {  return Cache.Instance.ImageCache;  }  set  {  Cache.Instance.ImageCache = value;  }  }  /// <summary>  /// secondary cache provider, by default: none,  /// use it if you have server in your local network  /// </summary>  public PureImageCache SecondaryCache  {  get  {  return Cache.Instance.ImageCacheSecond;  }  set  {  Cache.Instance.ImageCacheSecond = value;  }  }  /// <summary>  /// MemoryCache provider  /// </summary>  public readonly MemoryCache MemoryCache = new MemoryCache();  /// <summary>  /// load tiles in random sequence  /// </summary>  public bool ShuffleTilesOnLoad = false;  /// <summary>  /// tile queue to cache  /// </summary>  readonly Queue<CacheQueueItem> tileCacheQueue = new Queue<CacheQueueItem>();  bool? isRunningOnMono;  /// <summary>  /// return true if running on mono  /// </summary>  /// <returns></returns>  public bool IsRunningOnMono  {  get  {  if(!isRunningOnMono.HasValue)  {  try  {  isRunningOnMono = (Type.GetType("Mono.Runtime") != null);  return isRunningOnMono.Value;  }  catch  {  }  }  else  {  return isRunningOnMono.Value;  }  return false;  }  } |

1. 类的方法定义

|  |
| --- |
| public void SQLitePing()  public bool ExportToGMDB(string file)  public bool ImportFromGMDB(string file)  public bool OptimizeMapDb(string file)  void EnqueueCacheTask(CacheQueueItem task)  public void CancelTileCaching()  void CacheEngineLoop()  // etc… |

* + - 1. GPoint.cs

1. 概述

此cs文件为GMap点的结构体对象。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. GpsLog.cs

1. 概述

此cs文件为gps日志结构体对象。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. gpx.cs

1. 概述

此cs文件为gpx实体类对象，定义了gpxType、metadataType、personType、emailType、trksegType、wptType、linkType、fixType、extensionsType、trkType、rteType、boundsType、copyrightType等实体类或枚举。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. GRect.cs

1. 概述

此cs文件为地图矩形结构体对象。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. GSize.cs

1. 概述

此cs文件为地图大小结构体对象。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. Interface.cs

1. 概述

此cs文件为地图控件接口对象。

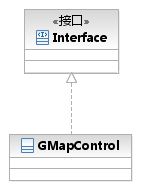
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| PointLatLng Position  {  get;  set;  }  GPoint PositionPixel  {  get;  }  string CacheLocation  {  get;  set;  }  bool IsDragging  {  get;  }  RectLatLng ViewArea  {  get;  }  GMapProvider MapProvider  {  get;  set;  }  bool CanDragMap  {  get;  set;  }  RenderMode RenderMode  {  get;  }  // events  event PositionChanged OnPositionChanged;  event TileLoadComplete OnTileLoadComplete;  event TileLoadStart OnTileLoadStart;  event MapDrag OnMapDrag;  event MapZoomChanged OnMapZoomChanged;  event MapTypeChanged OnMapTypeChanged;  void ReloadMap();  PointLatLng FromLocalToLatLng(int x, int y);  GPoint FromLatLngToLocal(PointLatLng point);  #if !PocketPC  #if SQLite  bool ShowExportDialog();  bool ShowImportDialog();  #endif  #endif |

* + - 1. LanguageType.cs

1. 概述

此cs文件语言枚举类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MapRoute.cs

1. 概述

此cs文件地图路径类对象。

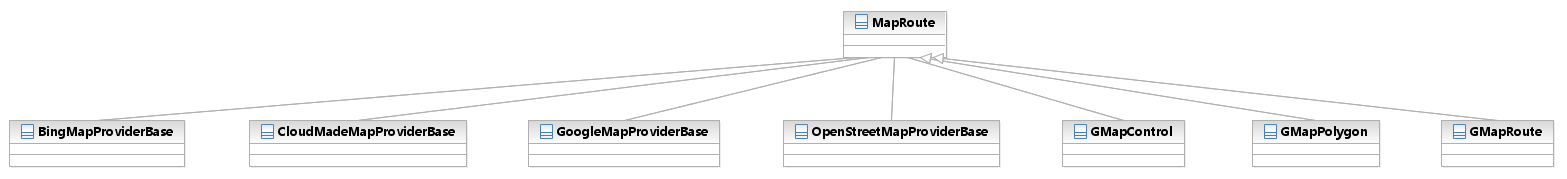
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

|  |
| --- |
| /// <summary>  /// points of route  /// </summary>  public readonly List<PointLatLng> Points = new List<PointLatLng>();  /// <summary>  /// route info  /// </summary>  public string Name;  /// <summary>  /// custom object  /// </summary>  public object Tag;  /// <summary>  /// route start point  /// </summary>  public PointLatLng? From  {  get  {  if(Points.Count > 0)  {  return Points[0];  }  return null;  }  }  /// <summary>  /// route end point  /// </summary>  public PointLatLng? To  {  get  {  if(Points.Count > 1)  {  return Points[Points.Count - 1];  }  return null;  }  }  /// <summary>  /// route distance (in km)  /// </summary>  public double Distance  {  get  {  double distance = 0.0;  if(From.HasValue && To.HasValue)  {  for(int i = 1; i < Points.Count; i++)  {  distance += GMapProviders.EmptyProvider.Projection.GetDistance(Points[i - 1], Points[i]);  }  }  return distance;  }  } |

1. 类的方法定义

|  |
| --- |
| public MapRoute(IEnumerable<PointLatLng> points, string name)  protected MapRoute(SerializationInfo info, StreamingContext context)  public virtual void GetObjectData(SerializationInfo info, StreamingContext context)  public virtual void OnDeserialization(object sender) |

* + - 1. MapType.cs

1. 概述

此cs文件为地图类型枚举类对象。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MouseWheelZoomType.cs

1. 概述

此cs文件为地图鼠标滚轮缩放类型枚举类对象。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. Placemark.cs

1. 概述

此cs文件为地标信息结构体。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| public struct Placemark  {  string address;  /// <summary>  /// the address  /// </summary>  public string Address  {  get  {  return address;  }  internal set  {  address = value;  }  }  /// <summary>  /// the accuracy of address  /// </summary>  public int Accuracy;  // parsed values from address  public string ThoroughfareName;  public string LocalityName;  public string PostalCodeNumber;  public string CountryName;  public string AdministrativeAreaName;  public string DistrictName;  public string SubAdministrativeAreaName;  public string Neighborhood;  public string StreetNumber;  public string CountryNameCode;  public string HouseNo;  internal Placemark(string address)  {  this.address = address;  Accuracy = 0;  HouseNo = string.Empty;  ThoroughfareName = string.Empty;  DistrictName = string.Empty;  LocalityName = string.Empty;  PostalCodeNumber = string.Empty;  CountryName = string.Empty;  CountryNameCode = string.Empty;  AdministrativeAreaName = string.Empty;  SubAdministrativeAreaName = string.Empty;  Neighborhood = string.Empty;  StreetNumber = string.Empty;  }  } |

1. 类的方法定义

无。

* + - 1. PointLatLng.cs

1. 概述

此cs文件为坐标点结构体。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. PureImageCache.cs

1. 概述

此cs文件为图片缓存接口。

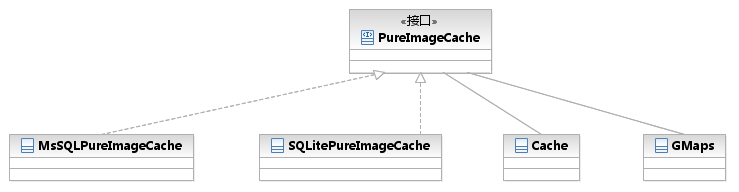
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| /// <summary>  /// pure abstraction for image cache  /// </summary>  public interface PureImageCache  {  /// <summary>  /// puts image to db  /// </summary>  /// <param name="tile"></param>  /// <param name="type"></param>  /// <param name="pos"></param>  /// <param name="zoom"></param>  /// <returns></returns>  bool PutImageToCache(byte[] tile, int type, GPoint pos, int zoom);  /// <summary>  /// gets image from db  /// </summary>  /// <param name="type"></param>  /// <param name="pos"></param>  /// <param name="zoom"></param>  /// <returns></returns>  PureImage GetImageFromCache(int type, GPoint pos, int zoom);  /// <summary>  /// delete old tiles beyond a supplied date  /// </summary>  /// <param name="date">Tiles older than this will be deleted.</param>  /// <param name="type">provider dbid or null to use all providers</param>  /// <returns>The number of deleted tiles.</returns>  int DeleteOlderThan(DateTime date, int ? type);  } |

* + - 1. PureProjection.cs

1. 概述

此cs文件为地图投影抽象类，包含多个抽象方法或者虚函数，以及数学计算方法。

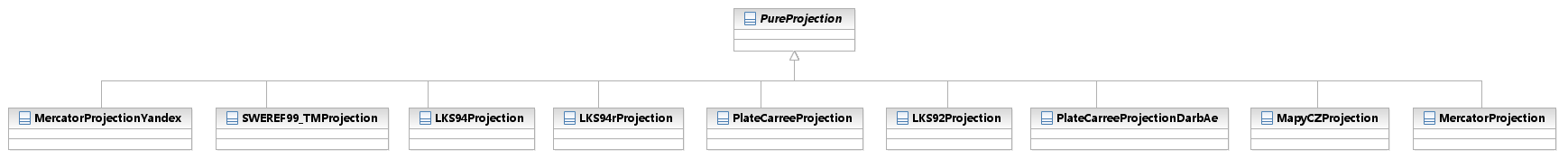
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| public abstract GSize TileSize  public abstract double Axis  public abstract double Flattening  public virtual RectLatLng Bounds |

1. 类的方法定义

|  |
| --- |
| public abstract GPoint FromLatLngToPixel(double lat, double lng, int zoom);  public abstract PointLatLng FromPixelToLatLng(long x, long y, int zoom);  public virtual GPoint FromPixelToTileXY(GPoint p)  public virtual GPoint FromTileXYToPixel(GPoint p)  public abstract GSize GetTileMatrixMinXY(int zoom);  public abstract GSize GetTileMatrixMaxXY(int zoom);  public virtual GSize GetTileMatrixSizeXY(int zoom)  public virtual GSize GetTileMatrixSizePixel(int zoom)  public virtual double GetGroundResolution(int zoom, double latitude) |

* + - 1. RectLatLng.cs

1. 概述

此cs文件为矩形坐标结构体。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. RenderMode.cs

1. 概述

此cs文件为地图绘制类型枚举类。

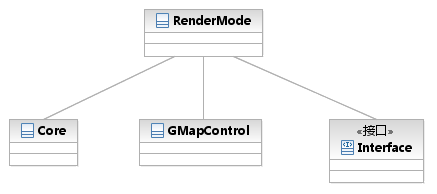
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. RoutingProvider.cs

1. 概述

此cs文件为路径选择接口。

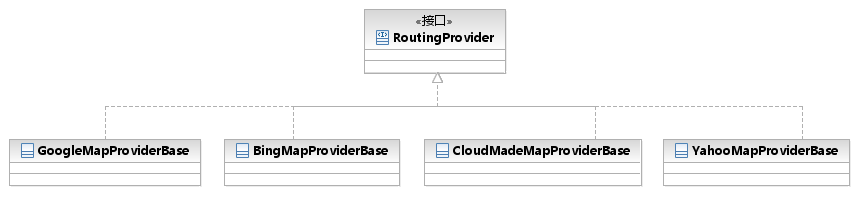
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| /// <summary>  /// routing interface  /// </summary>  public interface RoutingProvider  {  /// <summary>  /// get route between two points  /// </summary>  MapRoute GetRoute(PointLatLng start, PointLatLng end, bool avoidHighways, bool walkingMode, int Zoom);  /// <summary>  /// get route between two points  /// </summary>  MapRoute GetRoute(string start, string end, bool avoidHighways, bool walkingMode, int Zoom);  } |

* + - 1. Singleton.cs

1. 概述

此cs文件为类的单列泛型类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. SizeLatLng.cs

1. 概述

此cs文件为坐标大小结构体。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. StatusCodes.cs

1. 概述

此cs文件定义了地理编码状态代码枚举类GeoCoderStatusCode和方向状态代码枚举类DirectionsStatusCode。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. ZipStorer.cs

1. 概述

此cs文件为文件的压缩、解压类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + 1. GMap.NET.CacheProviders
       1. MemoryCache.cs

1. 概述

此cs文件为内存缓存类。

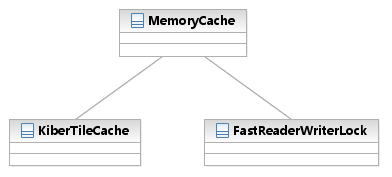
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// the amount of tiles in MB to keep in memmory, default: 22MB, if each ~100Kb it's ~222 tiles  /// </summary>  public int Capacity  {  get  {  kiberCacheLock.AcquireReaderLock();  try  {  return TilesInMemory.MemoryCacheCapacity;  }  finally  {  kiberCacheLock.ReleaseReaderLock();  }  }  set  {  kiberCacheLock.AcquireWriterLock();  try  {  TilesInMemory.MemoryCacheCapacity = value;  }  finally  {  kiberCacheLock.ReleaseWriterLock();  }  }  }  /// <summary>  /// current memmory cache size in MB  /// </summary>  public double Size  {  get  {  kiberCacheLock.AcquireReaderLock();  try  {  return TilesInMemory.MemoryCacheSize;  }  finally  {  kiberCacheLock.ReleaseReaderLock();  }  }  } |

1. 类的方法定义

|  |
| --- |
| internal byte[] GetTileFromMemoryCache(RawTile tile)  internal void AddTileToMemoryCache(RawTile tile, byte[] data)  internal void RemoveOverload()  public void Dispose() |

* + - 1. MsSQLCePureImageCache.cs

1. 概述

此cs文件为ms sql server图片缓存类。

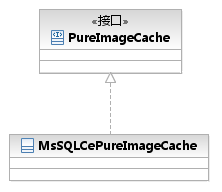
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MsSQLPureImageCache.cs

1. 概述

此cs文件为ms sql server图片缓存类。

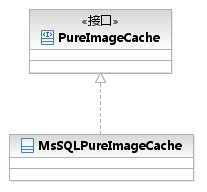
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MySQLPureImageCache.cs

1. 概述

此cs文件为MySQL图片缓存类。

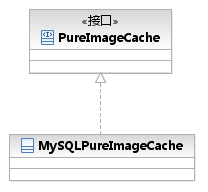
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. PostgreSQLPureImageCache.cs

1. 概述

此cs文件为postgresql server图片缓存类

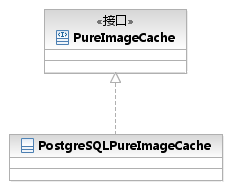
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. SQLitePureImageCache.cs

1. 概述

此cs文件为SQLite图片缓存类，是系统最快的瓦片图访问、存储方式。

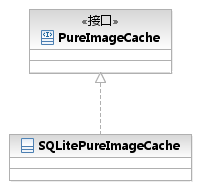
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + 1. GMap.NET.Internals
       1. SocksProxySocket
          1. AuthMethod.cs

1. 概述

此cs文件为

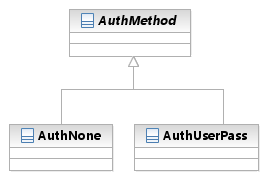
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// Gets or sets the socket connection with the proxy server.  /// </summary>  /// <value>The socket connection with the proxy server.</value>  protected Socket Server {  get {  return m\_Server;  }  set {  if (value == null)  throw new ArgumentNullException();  m\_Server = value;  }  }  /// <summary>  /// Gets or sets a byt array that can be used to store data.  /// </summary>  /// <value>A byte array to store data.</value>  protected byte[] Buffer {  get {  return m\_Buffer;  }  set {  m\_Buffer = value;  }  }  /// <summary>  /// Gets or sets the number of bytes that have been received from the remote proxy server.  /// </summary>  /// <value>An integer that holds the number of bytes that have been received from the remote proxy server.</value>  protected int Received {  get {  return m\_Received;  }  set {  m\_Received = value;  }  }  // private variables  /// <summary>Holds the value of the Buffer property.</summary>  private byte[] m\_Buffer;  /// <summary>Holds the value of the Server property.</summary>  private Socket m\_Server;  /// <summary>Holds the address of the method to call when the proxy has authenticated the client.</summary>  protected HandShakeComplete CallBack;  /// <summary>Holds the value of the Received property.</summary>  private int m\_Received; |

1. 类的方法定义

|  |
| --- |
| public abstract void Authenticate();  public abstract void BeginAuthenticate(HandShakeComplete callback); |

* + - * 1. AuthNone.cs

1. 概述

此cs文件为“没有认证”的实现类。

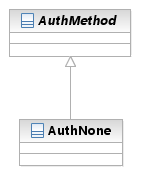
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. AuthUserPass.cs

1. 概述

此cs文件为“用户/密码认证”的实现类。

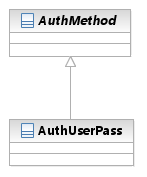
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. IAsyncProxyResult.cs

1. 概述

此cs文件为异步代理操作结果类。

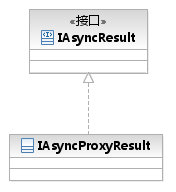
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. ProxyException.cs

1. 概述

此cs文件为代理异常类。

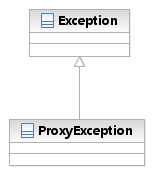
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. ProxySocket.cs

1. 概述

此cs文件为定义了代理类型枚举ProxyTypes和代理套接字类。

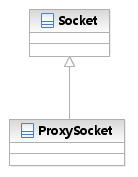
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// Gets or sets the EndPoint of the proxy server.  /// </summary>  /// <value>An IPEndPoint object that holds the IP address and the port of the proxy server.</value>  public IPEndPoint ProxyEndPoint  {  get  {  return m\_ProxyEndPoint;  }  set  {  m\_ProxyEndPoint = value;  }  }    /// <summary>  /// Gets or sets the type of proxy server to use.  /// </summary>  /// <value>One of the ProxyTypes values.</value>  public ProxyTypes ProxyType  {  get  {  return m\_ProxyType;  }  set  {  m\_ProxyType = value;  }  }    /// <summary>  /// Gets or sets a user-defined object.  /// </summary>  /// <value>The user-defined object.</value>  private object State  {  get  {  return m\_State;  }  set  {  m\_State = value;  }  }    /// <summary>  /// Gets or sets the username to use when authenticating with the proxy.  /// </summary>  /// <value>A string that holds the username that's used when authenticating with the proxy.</value>  /// <exception cref="ArgumentNullException">The specified value is null.</exception>  public string ProxyUser  {  get  {  return m\_ProxyUser;  }  set  {  if (value == null)  throw new ArgumentNullException();  m\_ProxyUser = value;  }  }    /// <summary>  /// Gets or sets the password to use when authenticating with the proxy.  /// </summary>  /// <value>A string that holds the password that's used when authenticating with the proxy.</value>  /// <exception cref="ArgumentNullException">The specified value is null.</exception>  public string ProxyPass  {  get  {  return m\_ProxyPass;  }  set  {  if (value == null)  throw new ArgumentNullException();  m\_ProxyPass = value;  }  }    /// <summary>  /// Gets or sets the asynchronous result object.  /// </summary>  /// <value>An instance of the IAsyncProxyResult class.</value>  private IAsyncProxyResult AsyncResult  {  get  {  return m\_AsyncResult;  }  set  {  m\_AsyncResult = value;  }  }    /// <summary>  /// Gets or sets the exception to throw when the EndConnect method is called.  /// </summary>  /// <value>An instance of the Exception class (or subclasses of Exception).</value>  private Exception ToThrow  {  get  {  return m\_ToThrow;  }  set  {  m\_ToThrow = value;  }  }    /// <summary>  /// Gets or sets the remote port the user wants to connect to.  /// </summary>  /// <value>An integer that specifies the port the user wants to connect to.</value>  private int RemotePort  {  get  {  return m\_RemotePort;  }  set  {  m\_RemotePort = value;  }  }    // private variables  /// <summary>Holds the value of the State property.</summary>  private object m\_State;  /// <summary>Holds the value of the ProxyEndPoint property.</summary>  private IPEndPoint m\_ProxyEndPoint = null;  /// <summary>Holds the value of the ProxyType property.</summary>  private ProxyTypes m\_ProxyType = ProxyTypes.None;  /// <summary>Holds the value of the ProxyUser property.</summary>  private string m\_ProxyUser = null;  /// <summary>Holds the value of the ProxyPass property.</summary>  private string m\_ProxyPass = null;  /// <summary>Holds a pointer to the method that should be called when the Socket is connected to the remote device.</summary>  private AsyncCallback CallBack = null;  /// <summary>Holds the value of the AsyncResult property.</summary>  private IAsyncProxyResult m\_AsyncResult;  /// <summary>Holds the value of the ToThrow property.</summary>  private Exception m\_ToThrow = null;  /// <summary>Holds the value of the RemotePort property.</summary>  private int m\_RemotePort; |

1. 类的方法定义

|  |
| --- |
| public ProxySocket(AddressFamily addressFamily, SocketType socketType, ProtocolType protocolType) : this(addressFamily, socketType, protocolType, "") { }  public ProxySocket(AddressFamily addressFamily, SocketType socketType, ProtocolType protocolType, string proxyUsername) : this(addressFamily, socketType, protocolType, proxyUsername, "") { }  public ProxySocket(AddressFamily addressFamily, SocketType socketType, ProtocolType protocolType, string proxyUsername, string proxyPassword)  public new void Connect(EndPoint remoteEP)  public void Connect(string host, int port)  public new IAsyncResult BeginConnect(EndPoint remoteEP, AsyncCallback callback, object state)  public IAsyncResult BeginConnect(string host, int port, AsyncCallback callback, object state)  public new void EndConnect(IAsyncResult asyncResult)  internal IAsyncProxyResult BeginDns(string host, HandShakeComplete callback) |

* + - * 1. Socks4Handler.cs

1. 概述

此cs文件为Implements the SOCKS4[A] protocol。

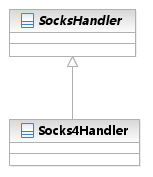
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. Socks5Handler.cs

1. 概述

此cs文件为Implements the SOCKS5 protocol。

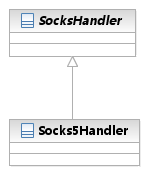
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. SocksHandler.cs

1. 概述

此cs文件为套接字事件抽象类。

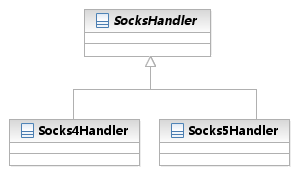
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// Gets or sets the socket connection with the proxy server.  /// </summary>  /// <value>A Socket object that represents the connection with the proxy server.</value>  /// <exception cref="ArgumentNullException">The specified value is null.</exception>  protected Socket Server {  get {  return m\_Server;  }  set {  if (value == null)  throw new ArgumentNullException();  m\_Server = value;  }  }  /// <summary>  /// Gets or sets the username to use when authenticating with the proxy server.  /// </summary>  /// <value>A string that holds the username to use when authenticating with the proxy server.</value>  /// <exception cref="ArgumentNullException">The specified value is null.</exception>  protected string Username {  get {  return m\_Username;  }  set {  if (value == null)  throw new ArgumentNullException();  m\_Username = value;  }  }  /// <summary>  /// Gets or sets the return value of the BeginConnect call.  /// </summary>  /// <value>An IAsyncProxyResult object that is the return value of the BeginConnect call.</value>  protected IAsyncProxyResult AsyncResult {  get {  return m\_AsyncResult;  }  set {  m\_AsyncResult = value;  }  }  /// <summary>  /// Gets or sets a byte buffer.  /// </summary>  /// <value>An array of bytes.</value>  protected byte[] Buffer {  get {  return m\_Buffer;  }  set {  m\_Buffer = value;  }  }  /// <summary>  /// Gets or sets the number of bytes that have been received from the remote proxy server.  /// </summary>  /// <value>An integer that holds the number of bytes that have been received from the remote proxy server.</value>  protected int Received {  get {  return m\_Received;  }  set {  m\_Received = value;  }  } |

1. 类的方法定义

|  |
| --- |
| public abstract void Negotiate(string host, int port);  public abstract void Negotiate(IPEndPoint remoteEP);  public abstract IAsyncProxyResult BeginNegotiate(IPEndPoint remoteEP, HandShakeComplete callback, IPEndPoint proxyEndPoint);  public abstract IAsyncProxyResult BeginNegotiate(string host, int port, HandShakeComplete callback, IPEndPoint proxyEndPoint); |

* + - * 1. SocksHttpWebRequest.cs

1. 概述

此cs文件定义了套接字网络请求类SocksHttpWebRequest和套接字网络响应类，实现对网络请求和响应的封装。

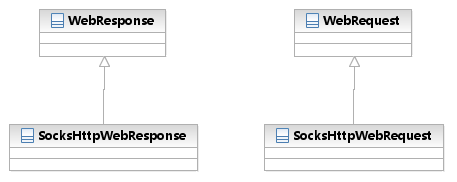
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. Cache.cs

1. 概述

此cs文件定义了缓存位置类CacheLocator、缓存枚举CacheType和缓存类Cache，系统默认缓存为SQLitePureImageCache。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| /// <summary>  /// abstract image cache  /// </summary>  public PureImageCache ImageCache;  /// <summary>  /// second level abstract image cache  /// </summary>  public PureImageCache ImageCacheSecond; |

1. 类的方法定义

无。

* + - 1. CacheQueueItem.cs

1. 概述

此cs文件定义了缓存队列结构体CacheQueueItem和缓存使用枚举类CacheUsage。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. Core.cs

1. 概述

此cs文件为地图控制的核心类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| public PointLatLng position;  public GPoint positionPixel;  public GPoint renderOffset;  public GPoint centerTileXYLocation;  public GPoint centerTileXYLocationLast;  public GPoint dragPoint;  public GPoint compensationOffset;  public GPoint mouseDown;  public GPoint mouseCurrent;  public GPoint mouseLastZoom;  public MouseWheelZoomType MouseWheelZoomType = MouseWheelZoomType.MousePositionAndCenter;  public bool MouseWheelZoomEnabled = true;  public PointLatLng? LastLocationInBounds = null;  public bool VirtualSizeEnabled = false;  public GSize sizeOfMapArea;  public GSize minOfTiles;  public GSize maxOfTiles;  public GRect tileRect;  public GRect tileRectBearing;  //public GRect currentRegion;  public float bearing = 0;  public bool IsRotated = false;  public bool fillEmptyTiles = true;  public TileMatrix Matrix = new TileMatrix();  public List<DrawTile> tileDrawingList = new List<DrawTile>();  public FastReaderWriterLock tileDrawingListLock = new FastReaderWriterLock();  #if !NET40  public readonly Stack<LoadTask> tileLoadQueue = new Stack<LoadTask>();  #endif |

1. 类的方法定义

无。

* + - 1. DrawTile.cs

1. 概述

此cs文件为绘制瓦片图的结构体。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| public GPoint PosXY;  public GPoint PosPixel;  public double DistanceSqr; |

1. 类的方法定义

|  |
| --- |
| public int CompareTo(DrawTile other)  public override string ToString()  public bool Equals(DrawTile other) |

* + - 1. FastReaderWriterLock.cs

1. 概述

此cs文件为快速度写锁的密封类，并包含Win32API类NativeMethods。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. FastResourceLock.cs

1. 概述

此cs文件为快速资源锁内部密封类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. KiberTileCache.cs

1. 概述

此cs文件为瓦片缓存字典类。

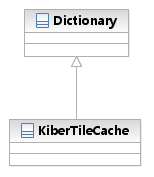
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| /// <summary>  /// the amount of tiles in MB to keep in memmory, default: 22MB, if each ~100Kb it's ~222 tiles  /// </summary>  public int MemoryCacheCapacity = 22;  /// <summary>  /// current memmory cache size in MB  /// </summary>  public double MemoryCacheSize  {  get  {  return memoryCacheSize / 1048576.0;  }  } |

1. 类的方法定义

|  |
| --- |
| public new void Add(RawTile key, byte[] value)  private new void Remove(RawTile key)  public new void Clear() |

* + - 1. LoadTask.cs

1. 概述

此cs文件定义了瓦片加载任务结构体LoadTask和瓦片加载比较类LoadTaskComparer。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. PureImage.cs

1. 概述

此cs文件定义了图片抽象代理类PureImageProxy和图片抽象类PureImage。

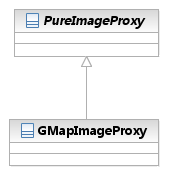
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| abstract public PureImage FromStream(Stream stream);  abstract public bool Save(Stream stream, PureImage image); |

* + - 1. RawTile.cs

1. 概述

此cs文件定义了瓦片行结构体RawTile和结构体比较类RawTileComparer。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

|  |
| --- |
| public int Type;  public GPoint Pos;  public int Zoom; |

1. 类的方法定义

|  |
| --- |
| public override string ToString()  public bool Equals(RawTile x, RawTile y)  public int GetHashCode(RawTile obj) |

* + - 1. Stuff.cs

1. 概述

此cs文件是一些公共的方法的定义和实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. Tile.cs

1. 概述

此cs文件为瓦片结构体。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. TileHttpHost.cs

1. 概述

此cs文件为瓦片网络主机实现类。

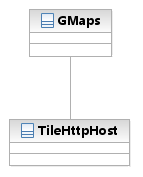
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. TileMatrix.cs

1. 概述

此cs文件为瓦片矩阵类。

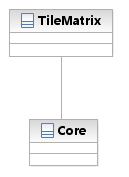
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| List<Dictionary<GPoint, Tile>> Levels = new List<Dictionary<GPoint, Tile>>(33);  FastReaderWriterLock Lock = new FastReaderWriterLock(); |

1. 类的方法定义

|  |
| --- |
| public void ClearAllLevels()  public void ClearLevel(int zoom)  public void ClearLevelAndPointsNotIn(int zoom, List<DrawTile> list)  public void ClearLevelsBelove(int zoom)  public void ClearLevelsAbove(int zoom)  public void EnterReadLock()  public void LeaveReadLock()  public Tile GetTileWithNoLock(int zoom, GPoint p)  public Tile GetTileWithReadLock(int zoom, GPoint p)  public void SetTile(Tile t)  public void Dispose() |

* + 1. GMap.NET.MapProviders
       1. ArcGIS
          1. ArcGIS\_DarbAE\_Q2\_2011\_NAVTQ\_Eng\_V5\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

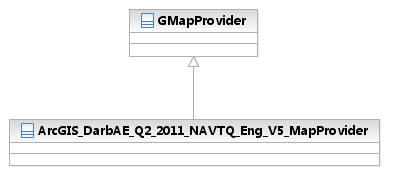
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("E03CFEDF-9277-49B3-9912-D805347F934B");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_DarbAE\_Q2\_2011\_NAVTQ\_Eng\_V5\_MapProvider";  public override string Name  {  get  {  return name;  }  }  public override PureProjection Projection  {  get  {  return PlateCarreeProjectionDarbAe.Instance;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { this };  }  return overlays;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_Imagery\_World\_2D\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

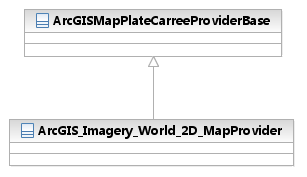
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("FF7ADDAD-F155-41DB-BC42-CC6FD97C8B9D");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_Imagery\_World\_2D\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_ShadedRelief\_World\_2D\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

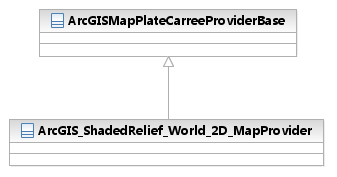
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("A8995FA4-D9D8-415B-87D0-51A7E53A90D4");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_ShadedRelief\_World\_2D\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_StreetMap\_World\_2D\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

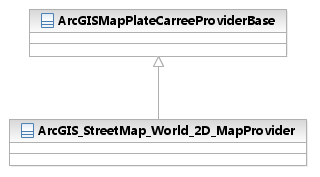
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("00BF56D4-4B48-4939-9B11-575BBBE4A718");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_StreetMap\_World\_2D\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_Topo\_US\_2D\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

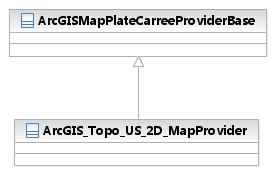
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("7652CC72-5C92-40F5-B572-B8FEAA728F6D");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_Topo\_US\_2D\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_World\_Physical\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

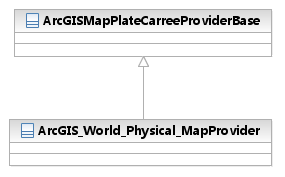
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("0C0E73E3-5EA6-4F08-901C-AE85BCB1BFC8");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_World\_Physical\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_World\_Shaded\_Relief\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

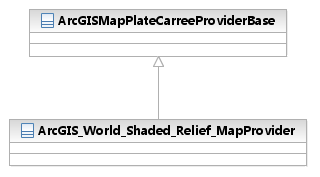
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("2E821FEF-8EA1-458A-BC82-4F699F4DEE79");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_World\_Shaded\_Relief\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_World\_Street\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("E1FACDF6-E535-4D69-A49F-12B623A467A9");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_World\_Street\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_World\_Terrain\_Base\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

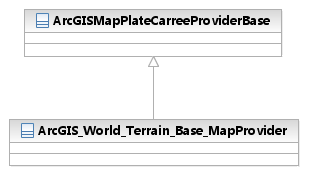
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("927F175B-5200-4D95-A99B-1C87C93099DA");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_World\_Terrain\_Base\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. ArcGIS\_World\_Topo\_MapProvider.cs

1. 概述

此cs文件为ArcGIS地图提供者实现类。

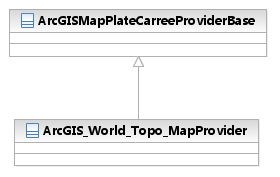
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("E0354A49-7447-4C9A-814F-A68565ED834B");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "ArcGIS\_World\_Topo\_Map";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - 1. Bing
         1. BingHybridMapProvider.cs

1. 概述

此cs文件为必应地图提供者实现类。

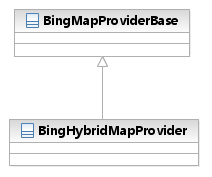
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("94E2FCB4-CAAC-45EA-A1F9-8147C4B14970");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "BingHybridMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  public override void OnInitialized()  {  base.OnInitialized();  if(!DisableDynamicTileUrlFormat)  {  //UrlFormat[AerialWithLabels]: http://ecn.{subdomain}.tiles.virtualearth.net/tiles/h{quadkey}.jpeg?g=3179&mkt={culture}  UrlDynamicFormat = GetTileUrl("AerialWithLabels");  if(!string.IsNullOrEmpty(UrlDynamicFormat))  {  UrlDynamicFormat = UrlDynamicFormat.Replace("{subdomain}", "t{0}").Replace("{quadkey}", "{1}").Replace("{culture}", "{2}");  }  }  }  #endregion |

* + - * 1. BingMapProvider.cs

1. 概述

此cs文件为必应地图提供者实现类。

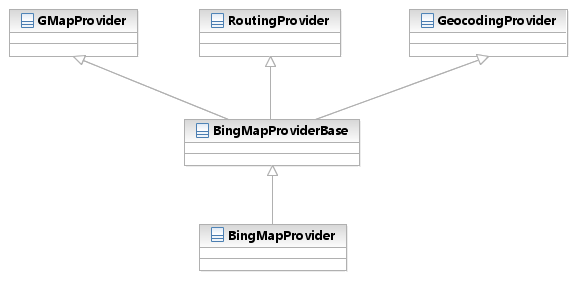
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("D0CEB371-F10A-4E12-A2C1-DF617D6674A8");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "BingMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  public override void OnInitialized()  {  base.OnInitialized();  if(!DisableDynamicTileUrlFormat)  {  //UrlFormat[Road]: http://ecn.{subdomain}.tiles.virtualearth.net/tiles/r{quadkey}.jpeg?g=3179&mkt={culture}&shading=hill  UrlDynamicFormat = GetTileUrl("Road");  if(!string.IsNullOrEmpty(UrlDynamicFormat))  {  UrlDynamicFormat = UrlDynamicFormat.Replace("{subdomain}", "t{0}").Replace("{quadkey}", "{1}").Replace("{culture}", "{2}");  }  }  }  #endregion |

* + - * 1. BingSatelliteMapProvider.cs

1. 概述

此cs文件为必应地图提供者实现类。

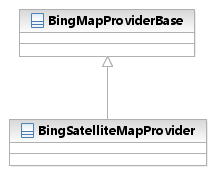
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("3AC742DD-966B-4CFB-B67D-33E7F82F2D37");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "BingSatelliteMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  public override void OnInitialized()  {  base.OnInitialized();  if(!DisableDynamicTileUrlFormat)  {  //UrlFormat[Aerial]: http://ecn.{subdomain}.tiles.virtualearth.net/tiles/a{quadkey}.jpeg?g=3179  UrlDynamicFormat = GetTileUrl("Aerial");  if(!string.IsNullOrEmpty(UrlDynamicFormat))  {  UrlDynamicFormat = UrlDynamicFormat.Replace("{subdomain}", "t{0}").Replace("{quadkey}", "{1}");  }  }  }  #endregion |

* + - 1. Czech
         1. CzechGeographicMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

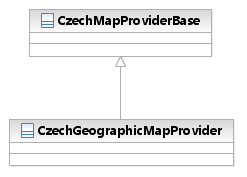
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("50EC9FCC-E4D7-4F53-8700-2D1DB73A1D48");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechGeographicMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. CzechHistoryMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

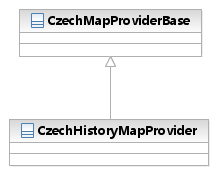
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("CD44C19D-5EED-4623-B367-FB39FDC55B8F");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechHistoryMap";  public override string Name  {  get  {  return name;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { this, CzechHybridMapProvider.Instance };  }  return overlays;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. CzechHybridMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

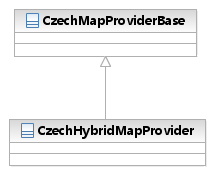
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("7540CE5B-F634-41E9-B23E-A6E0A97526FD");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechHybridMap";  public override string Name  {  get  {  return name;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { CzechSatelliteMapProvider.Instance, this };  }  return overlays;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. CzechMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

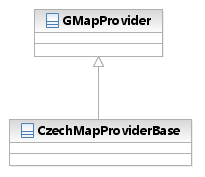
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("13AB92EF-8C3B-4FAC-B2CD-2594C05F8BFC");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. CzechSatelliteMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

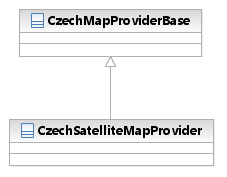
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("30F433DB-BBF5-463D-9AB5-76383483B605");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechSatelliteMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. CzechTuristMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

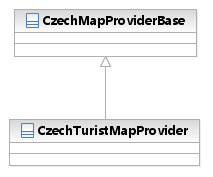
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("102A54BE-3894-439B-9C1F-CA6FF2EA1FE9");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechTuristMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. CzechTuristWinterMapProvider.cs

1. 概述

此cs文件为Czech地图提供者实现类。

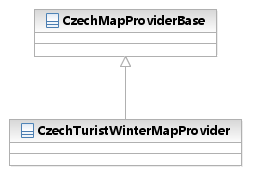
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("F7B7FC9E-BDC2-4A9D-A1D3-A6BEC8FE0EB2");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CzechTuristWinterMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - 1. CzechOld
         1. CzechHistoryMapProvider.cs
         2. CzechHybridMapProvider.cs
         3. CzechMapProvider.cs
         4. CzechSatelliteMapProvider.cs
         5. CzechTuristMapProvider.cs
      2. Etc
         1. CloudMadeMapProvider.cs

1. 概述

此cs文件为CloudMade地图提供者实现类。

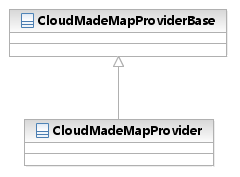
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("00403A36-725F-4BC4-934F-BFC1C164D003");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "CloudMade, Demo";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. LatviaMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

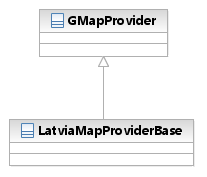
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("2A21CBB1-D37C-458D-905E-05F19536EF1F");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "LatviaMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. MapBenderWMSProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

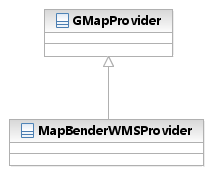
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("45742F8D-B552-4CAF-89AE-F20951BBDB2B");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "MapBender, WMS demo";  public override string Name  {  get  {  return name;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { this };  }  return overlays;  }  }  public override PureProjection Projection  {  get  {  return MercatorProjection.Instance;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. SpainMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

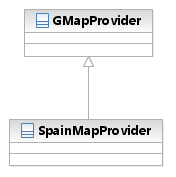
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("7B70ABB0-1265-4D34-9442-F0788F4F689F");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "SpainMap";  public override string Name  {  get  {  return name;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { this };  }  return overlays;  }  }  public override PureProjection Projection  {  get  {  return MercatorProjection.Instance;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. SwedenMapProvider.cs

1. 概述

此cs文件为Sweden地图提供者实现类。

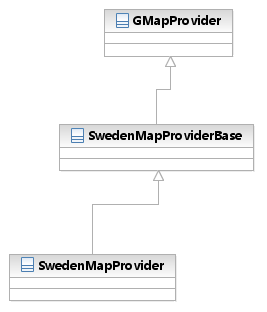
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("40890A96-6E82-4FA7-90A3-73D66B974F63");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "SwedenMap";  public override string Name  {  get  {  return name;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. TurkeyMapProvider.cs

1. 概述

此cs文件为Turkey地图提供者实现类。

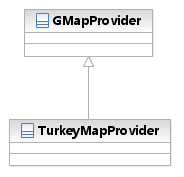
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("EDE895BD-756D-4BE4-8D03-D54DD8856F1D");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "TurkeyMap";  public override string Name  {  get  {  return name;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { this };  }  return overlays;  }  }  public override PureProjection Projection  {  get  {  return MercatorProjection.Instance;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, LanguageStr);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - * 1. WikiMapiaMapProvider.cs

1. 概述

此cs文件为WikiMapia地图提供者实现类。

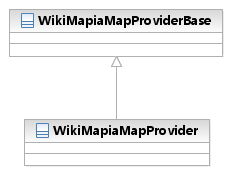
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

|  |
| --- |
| #region GMapProvider Members  readonly Guid id = new Guid("7974022B-1AA6-41F1-8D01-F49940E4B48C");  public override Guid Id  {  get  {  return id;  }  }  readonly string name = "WikiMapiaMap";  public override string Name  {  get  {  return name;  }  }  GMapProvider[] overlays;  public override GMapProvider[] Overlays  {  get  {  if(overlays == null)  {  overlays = new GMapProvider[] { this };  }  return overlays;  }  }  public override PureImage GetTileImage(GPoint pos, int zoom)  {  string url = MakeTileImageUrl(pos, zoom, string.Empty);  return GetTileImageUsingHttp(url);  }  #endregion |

* + - 1. Google
         1. China

4.1.4.6.1.1 GoogleChinaHybridMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义

4.1.4.6.1.2 GoogleChinaMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义

4.1.4.6.1.3 GoogleChinaSatelliteMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义

4.1.4.6.1.4 GoogleChinaTerrainMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + - 1. Korea

4.1.4.6.2.1 GoogleKoreaHybridMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义

4.1.4.6.2.2 GoogleKoreaMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义

4.1.4.6.2.3 GoogleKoreaSatelliteMapProvider.cs

1. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + - 1. GoogleHybridMapProvider.cs
2. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + - 1. GoogleMapProvider.cs
2. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + - 1. GoogleSatelliteMapProvider.cs
2. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + - 1. GoogleTerrainMapProvider.cs
2. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + 1. Lithuania
          1. Lithuania3dMapProvider.cs
2. 概述

此cs文件为Latvia地图提供者实现类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义
   * + - 1. LithuaniaHybridMapProvider.cs
         2. LithuaniaHybridOldMapProvider.cs
         3. LithuaniaMapProvider.cs
         4. LithuaniaOrtoFotoMapProvider.cs
         5. LithuaniaOrtoFotoOldMapProvider.cs
         6. LithuaniaReliefMapProvider.cs
         7. LithuaniaTOP50.cs
       1. NearMap
          1. NearHybridMapProvider.cs
          2. NearMapProvider.cs
          3. NearSatelliteMapProvider.cs
       2. OpenStreetMap
          1. OpenCycleLandscapeMapProvider.cs
          2. OpenCycleMapProvider.cs
          3. OpenCycleTransportMapProvider.cs
          4. OpenSeaMapHybridProvider.cs
          5. OpenStreet4UMapProvider.cs
          6. OpenStreetMapProvider.cs
          7. OpenStreetMapQuestHybridProvider.cs
          8. OpenStreetMapQuestProvider.cs
          9. OpenStreetMapQuestSatteliteProvider.cs
          10. OpenStreetMapSurferProvider.cs
          11. OpenStreetMapSurferTerrainProvider.cs
          12. OpenStreetOsmProvider.cs
       3. Ovi
          1. OviHybridMapProvider.cs
          2. OviMapProvider.cs
          3. OviSatelliteMapProvider.cs
          4. OviTerrainMapProvider.cs
       4. Yahoo
          1. YahooHybridMapProvider.cs
          2. YahooMapProvider.cs
          3. YahooSatelliteMapProvider.cs
       5. Yandex
          1. YandexHybridMapProvider.cs
          2. YandexMapProvider.cs
          3. YandexSatelliteMapProvider.cs
       6. GMapProvider.cs
     1. GMap.NET.Projections
        1. MapsLTProjection.cs
2. 概述

此cs文件为立陶宛地图提供者所需地图投影实现类LKS94Projection。

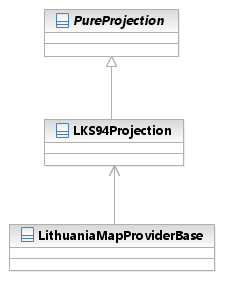
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MapsLTReliefProjection.cs

1. 概述

此cs文件为立陶宛地图提供者所需地图投影实现类LKS94rProjection。

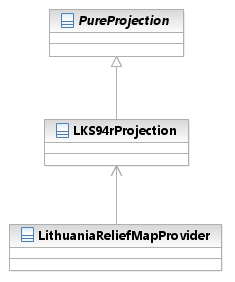
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MapsLVProjection.cs

1. 概述

此cs文件为拉脱维亚地图提供者所需地图投影实现类LKS92Projection。

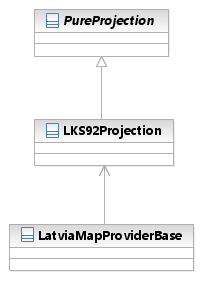
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MapyCZProjection.cs

1. 概述

此cs文件为Czech地图提供者地图投影实现类MapyCZProjection。

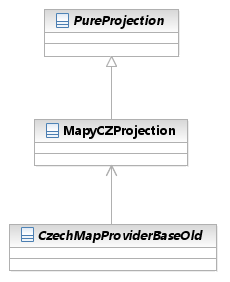
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MercatorProjection.cs

1. 概述

此cs文件为墨卡托投影实现类。

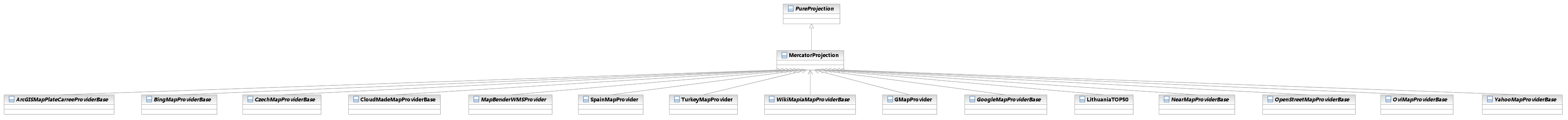
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. MercatorProjectionYandex.cs

1. 概述

此cs文件为地图投影实现类YandexMapProviderBase。

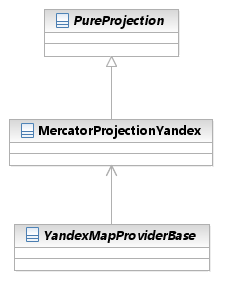
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. PlateCarreeProjection.cs

1. 概述

此cs文件为地图投影类。

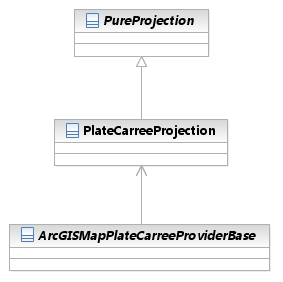
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. PlateCarreeProjectionDarbAe.cs

1. 概述

此cs文件为地图投影类。

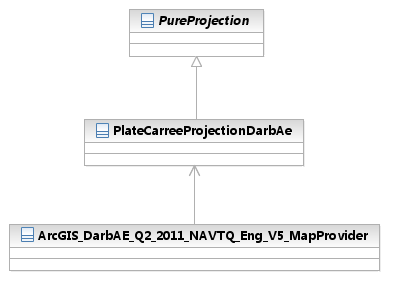
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. PlateCarreeProjectionPergo.cs

1. 概述

此cs文件暂时无效。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. SWEREF99\_TMProjection.cs

1. 概述

此cs文件为地图投影类。

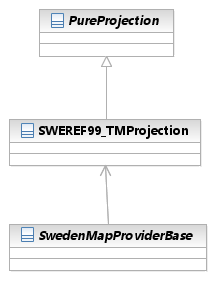
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* 1. GMap.NET.WindowsForms
     1. GMap.NET.ObjectModel
        1. ObservableCollection.cs

1. 概述

此cs文件定义了接口INotifyCollectionChanged、INotifyPropertyChanged，枚举类NotifyCollectionChangedAction和类NotifyCollectionChangedEventArgs、ObservableCollection、SimpleMonitor。

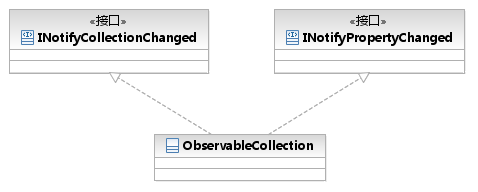
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. ObservableCollectionThreadSafe.cs

1. 概述

此cs文件为集合线程安全类实现。

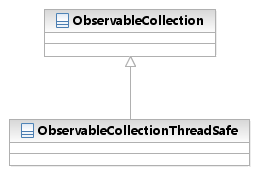
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + 1. GMap.NET.WindowsForms
       1. Markers
          1. GMarkerCross.cs

1. 概述

此cs文件为十字图标Marker。

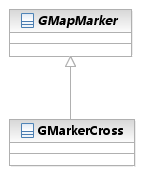
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. GMarkerGoogle.cs

1. 概述

此cs文件为GoogleMarker图标。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. ToolTips
         1. GMapBaloonToolTip.cs

1. 概述

此cs文件为气泡Tip类。

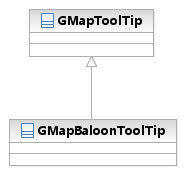
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - * 1. GMapRoundedToolTip.cs

1. 概述

此cs文件为Tip实现类。

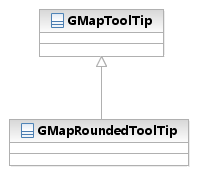
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. ColorMatrixs.cs

1. 概述

此cs文件为颜色矩阵静态类。

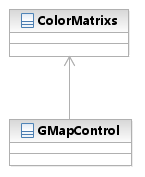
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. GMapControl.cs

1. 概述

此cs文件为地图控件类用户控件。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计
2. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. GMapImage.cs

1. 概述

此cs文件为定义了类GMapImage和GMapImageProxy。

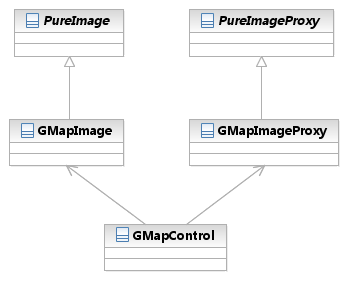
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. GMapMarker.cs

1. 概述

此cs文件为地图点图元类。

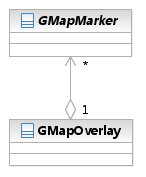
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| GMapOverlay overlay;  public GMapOverlay Overlay  {  get  {  return overlay;  }  internal set  {  overlay = value;  }  }  private PointLatLng position;  public PointLatLng Position  {  get  {  return position;  }  set  {  if(position != value)  {  position = value;  if(IsVisible)  {  if(Overlay != null && Overlay.Control != null)  {  Overlay.Control.UpdateMarkerLocalPosition(this);  }  }  }  }  }  public object Tag;  Point offset;  public Point Offset  {  get  {  return offset;  }  set  {  if(offset != value)  {  offset = value;  if(IsVisible)  {  if(Overlay != null && Overlay.Control != null)  {  Overlay.Control.UpdateMarkerLocalPosition(this);  }  }  }  }  }  Rectangle area;  /// <summary>  /// marker position in local coordinates, internal only, do not set it manualy  /// </summary>  public Point LocalPosition  {  get  {  return area.Location;  }  set  {  if(area.Location != value)  {  area.Location = value;  {  if(Overlay != null && Overlay.Control != null)  {  if(!Overlay.Control.HoldInvalidation)  {  Overlay.Control.Invalidate();  }  }  }  }  }  }  /// <summary>  /// ToolTip position in local coordinates  /// </summary>  public Point ToolTipPosition  {  get  {  Point ret = area.Location;  ret.Offset(-Offset.X, -Offset.Y);  return ret;  }  }  public Size Size  {  get  {  return area.Size;  }  set  {  area.Size = value;  }  }  public Rectangle LocalArea  {  get  {  return area;  }  }  public GMapToolTip ToolTip;  public MarkerTooltipMode ToolTipMode = MarkerTooltipMode.OnMouseOver;  string toolTipText;  public string ToolTipText  {  get  {  return toolTipText;  }  set  {  if(ToolTip == null && !string.IsNullOrEmpty(value))  {  #if !PocketPC  ToolTip = new GMapRoundedToolTip(this);  #else  ToolTip = new GMapToolTip(this);  #endif  }  toolTipText = value;  }  }  private bool visible = true;  /// <summary>  /// is marker visible  /// </summary>  public bool IsVisible  {  get  {  return visible;  }  set  {  if(value != visible)  {  visible = value;  if(Overlay != null && Overlay.Control != null)  {  if(visible)  {  Overlay.Control.UpdateMarkerLocalPosition(this);  }  else  {  if (Overlay.Control.IsMouseOverMarker)  {  Overlay.Control.IsMouseOverMarker = false;  #if !PocketPC  Overlay.Control.RestoreCursorOnLeave();  #endif  }  }  {  if(!Overlay.Control.HoldInvalidation)  {  Overlay.Control.Invalidate();  }  }  }  }  }  }  /// <summary>  /// if true, marker will be rendered even if it's outside current view  /// </summary>  public bool DisableRegionCheck = false;  /// <summary>  /// can maker receive input  /// </summary>  public bool IsHitTestVisible = true;  private bool isMouseOver = false;  /// <summary>  /// is mouse over marker  /// </summary>  public bool IsMouseOver  {  get  {  return isMouseOver;  }  internal set  {  isMouseOver = value;  }  } |

1. 类的方法定义

无。

* + - 1. GMapOverlay.cs

1. 概述

此cs文件为地图图层类。

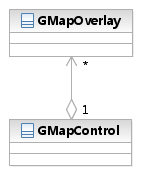
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| bool isVisibile = true;  /// <summary>  /// is overlay visible  /// </summary>  public bool IsVisibile  {  get  {  return isVisibile;  }  set  {  if(value != isVisibile)  {  isVisibile = value;  if(Control != null)  {  if(isVisibile)  {  Control.HoldInvalidation = true;  {  ForceUpdate();  }  Control.Refresh();  }  else  {  if (Control.IsMouseOverMarker)  {  Control.IsMouseOverMarker = false;  }  if (Control.IsMouseOverPolygon)  {  Control.IsMouseOverPolygon = false;  }  if (Control.IsMouseOverRoute)  {  Control.IsMouseOverRoute = false;  }  #if !PocketPC  Control.RestoreCursorOnLeave();  #endif  if(!Control.HoldInvalidation)  {  Control.Invalidate();  }  }  }  }  }  }  /// <summary>  /// overlay Id  /// </summary>  public string Id;  /// <summary>  /// list of markers, should be thread safe  /// </summary>  public readonly ObservableCollectionThreadSafe<GMapMarker> Markers = new ObservableCollectionThreadSafe<GMapMarker>();  /// <summary>  /// list of routes, should be thread safe  /// </summary>  public readonly ObservableCollectionThreadSafe<GMapRoute> Routes = new ObservableCollectionThreadSafe<GMapRoute>();  /// <summary>  /// list of polygons, should be thread safe  /// </summary>  public readonly ObservableCollectionThreadSafe<GMapPolygon> Polygons = new ObservableCollectionThreadSafe<GMapPolygon>();  GMapControl control;  public GMapControl Control  {  get  {  return control;  }  internal set  {  control = value;  }  } |

1. 类的方法定义

无。

* + - 1. GMapPolygon.cs

1. 概述

此cs文件为多边形图元类。

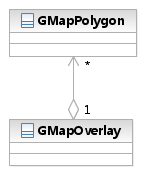
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| private bool visible = true;  /// <summary>  /// is polygon visible  /// </summary>  public bool IsVisible  {  get  {  return visible;  }  set  {  if(value != visible)  {  visible = value;  if(Overlay != null && Overlay.Control != null)  {  if(visible)  {  Overlay.Control.UpdatePolygonLocalPosition(this);  }  else  {  if (Overlay.Control.IsMouseOverPolygon)  {  Overlay.Control.IsMouseOverPolygon = false;  #if !PocketPC  Overlay.Control.RestoreCursorOnLeave();  #endif  }  }  {  if(!Overlay.Control.HoldInvalidation)  {  Overlay.Control.Invalidate();  }  }  }  }  }  }  /// <summary>  /// can receive input  /// </summary>  public bool IsHitTestVisible = false;  private bool isMouseOver = false;  /// <summary>  /// is mouse over  /// </summary>  public bool IsMouseOver  {  get  {  return isMouseOver;  }  internal set  {  isMouseOver = value;  }  }  GMapOverlay overlay;  public GMapOverlay Overlay  {  get  {  return overlay;  }  internal set  {  overlay = value;  }  } |

1. 类的方法定义

无。

* + - 1. GMapRoute.cs

1. 概述

此cs文件为线图元类。

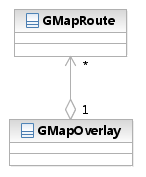
1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计



1. 类的属性定义

|  |
| --- |
| GMapOverlay overlay;  public GMapOverlay Overlay  {  get  {  return overlay;  }  internal set  {  overlay = value;  }  }  private bool visible = true;  /// <summary>  /// is marker visible  /// </summary>  public bool IsVisible  {  get  {  return visible;  }  set  {  if (value != visible)  {  visible = value;  if (Overlay != null && Overlay.Control != null)  {  if (visible)  {  Overlay.Control.UpdateRouteLocalPosition(this);  }  else  {  if (Overlay.Control.IsMouseOverRoute)  {  Overlay.Control.IsMouseOverRoute = false;  #if !PocketPC  Overlay.Control.RestoreCursorOnLeave();  #endif  }  }  {  if (!Overlay.Control.HoldInvalidation)  {  Overlay.Control.Invalidate();  }  }  }  }  }  }  /// <summary>  /// can receive input  /// </summary>  public bool IsHitTestVisible = false;  private bool isMouseOver = false;  /// <summary>  /// is mouse over  /// </summary>  public bool IsMouseOver  {  get  {  return isMouseOver;  }  internal set  {  isMouseOver = value;  }  } |

1. 类的方法定义

无。

* + - 1. GMapToolTip.cs

1. 概述

此cs文件为标牌类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

* + - 1. TilePrefetcher.cs

1. 概述

此cs文件为瓦片图下载类。

1. 功能要求

无。

1. 性能要求

无。

1. 类的结构设计

无。

1. 类的属性定义

无。

1. 类的方法定义

无。

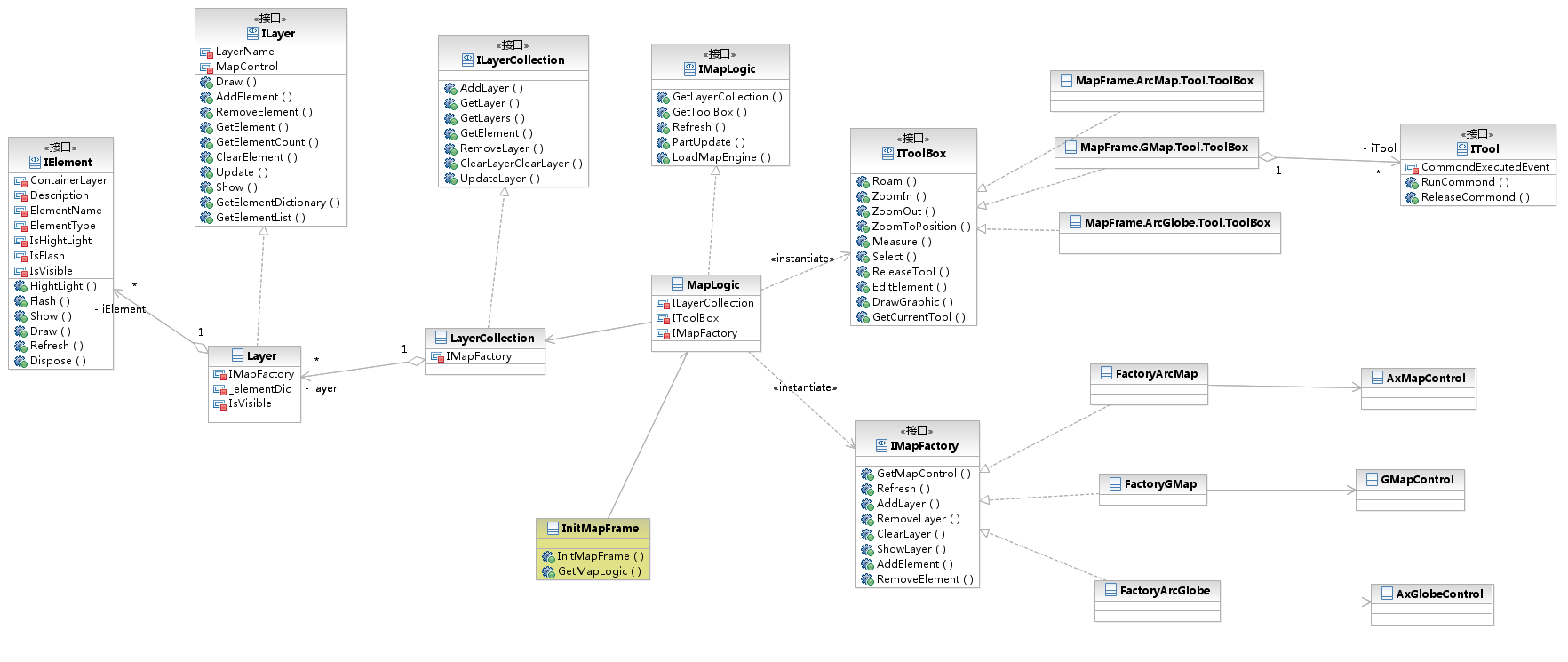
1. GMap.NET应用实例
   1. 通用地图框架
      1. 概述

通用地图框架是将多种地图引擎封装成统一的接口，再提供给GIS二次开发人员进行调用，让业务展示与地图引擎完全剥离，开发者无需再关注地图引擎的相关接口，以及地图引擎的工作方式等，从而将更多的时间和精力投入到GIS业务功能的开发，且地图框架是一次开发多次使用的方式，从而提高了开发效率。

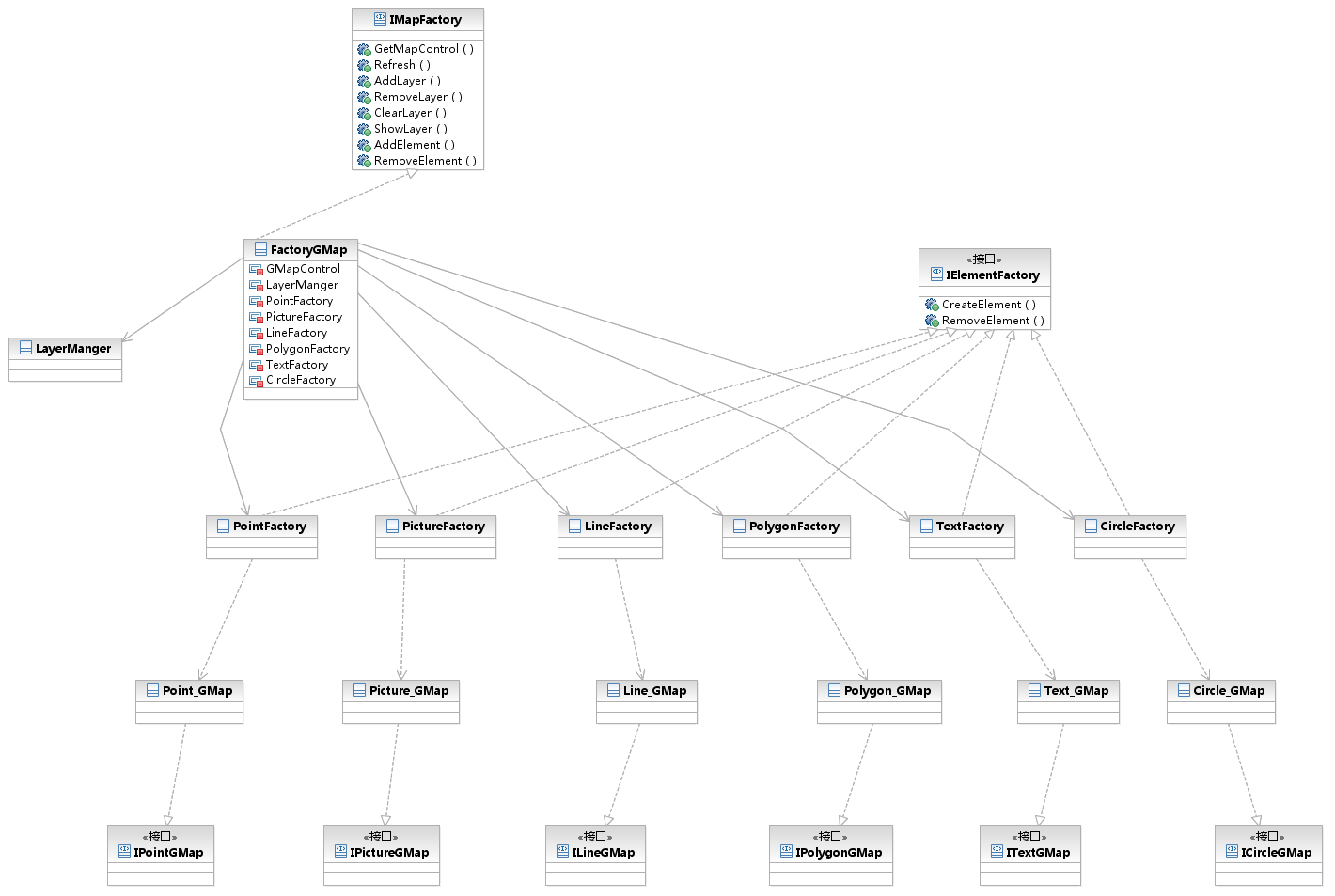
目前通用地图框架已将GMap.Net、ArcMap、ArcGlobe地图引擎封装。

* + 1. 类设计

1. 主逻辑类图



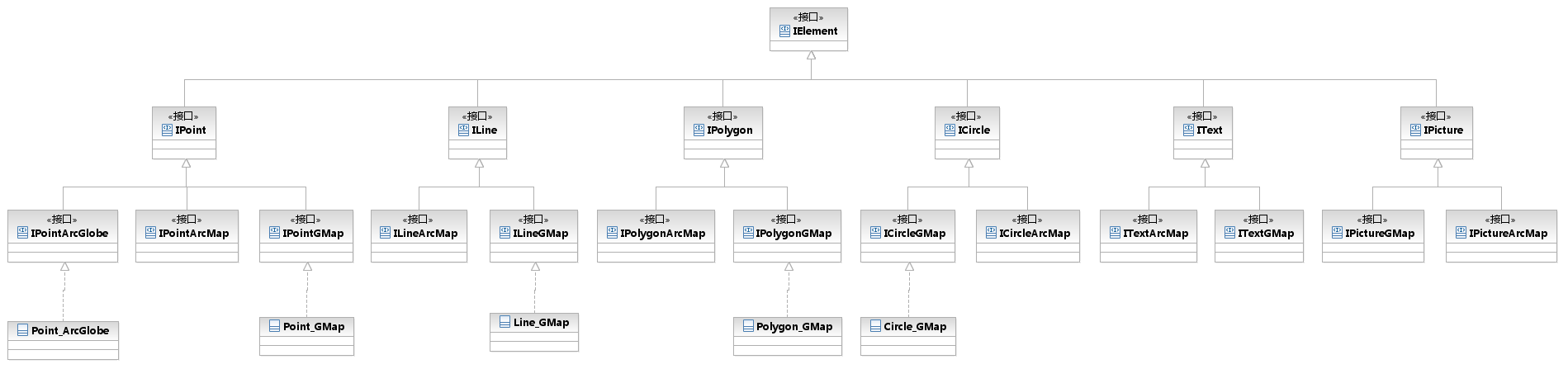
1. 图元创建删除类图



1. 工具箱类图



1. 图元结构类图



1. 结束语

地理信息系统GIS在国防、教育、矿业、海洋等许多领域应用非常广泛，且发展也比较成熟，市面上的地图引擎也比较多，但万变不离其中，只要掌握到其核心知识，everything is ok !

你很忙，但知识并不慌，加油！