目录

[瓦片地图技术研究 2](#_Toc486877834)

[网络地图投影 2](#_Toc486877835)

[球面墨卡托投影 2](#_Toc486877836)

[地理投影 4](#_Toc486877837)

[网络地图规范 4](#_Toc486877838)

[WMS规范 5](#_Toc486877839)

[WMTS规范： 6](#_Toc486877840)

[TMS规范 8](#_Toc486877841)

# 瓦片地图技术研究

瓦片地图是当今主流的网络地图服务形式，几乎所有主流的地图厂商如谷歌地图，百度地图，高德地图等都以瓦片地图的形式来提供地图服务。本文对以瓦片地图为代表的网络地图技术进行深入研究，并评价不同的网络地图技术的优缺点。

## 网络地图投影

网络地图的展示特点（空间分析结果，导航路线，位置等的展示）决定了它并不需要精确的分带投影（高斯投影，横轴墨卡托投影）。为了让不同的网络地图之间有更好的互操作性（相同区域的图层可以叠加，不同区域的图层可以拼接），网络地图需要使用一个全球统一的投影。

根据以上的这些要求，在网络地图中一般使用球面墨卡托投影和地理投影。下文分别对这两种投影进行深入介绍。

### 球面墨卡托投影

墨卡托投影是一个等角正轴切圆柱投影。

#### 球体模型

标准的墨卡托投影使用一个旋转椭球体来建模地球。基于网路地图的需求，以及大规模网络服务中对于计算速度的要求，通常会使用一个球体模型来建模地球，文献显示使用球体模型带来的误差小于0.3%，这在以展示为主要目的的网络地图中是完全可以接受的。

#### 球体半径确定方法：

一般球体的半径a可以是以下的值：

1. 椭球体的赤道半径
2. 椭球体的半轴长度的几何平均数或者算术平均数
3. 一个体积等于椭球体的球体的半径

这些半径的差值在35km以内，对于一个小比例尺地图，这样的差异可以忽略。所以我们使用均值**6371km**和**40030km**分别作为椭球体的**半径**和**周长**。

#### 球面墨卡托投影公式的推导

投影是一个从地理坐标(经度λ,纬度φ)到笛卡尔坐标(横轴x，纵轴y)的函数。下文给出球面墨卡托投影函数的推导。

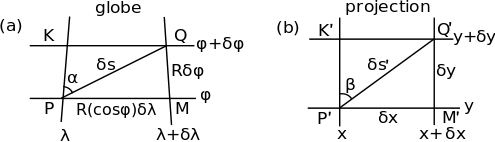
##### 投影的概念模型

投影函数的推导基于以下这3条**规定**：

1. 投影在赤道上没有变形。
2. 经度相同的点被投影到同一条母线。（1，2确定了x关于λ的函数）
3. 在任何一点没有角度变形。

##### 分析元素

分析元素是一个在小球体上的小区域，通过分析这个小区域投影前后的性质可以推导出，投影函数。



分析元素

设球体上的分析元素的四个端点是：

它投影后平面上的分析元素的四个端点分别是：

()

我们可以得到纬度方向的长度的比例:

我们也可以得到经度方向的长度的比例：

根据，投影在赤道上没有变形，以及经度相同的点被投影到同一条母线上，我们可以得到。

这里假设了投影坐标的原点在本初子午线（0度经线）上。

当这个分析元素足够小的时候可得:

根据，投影前后没有角度变形，也就是说，经纬度的变形相同。可以得到：

即：

求解，可以得到投影函数：

### 地理投影

投影公式：

这个投影简单地把经度作为投影后的x坐标，维度作为投影后的y坐标。由于它不是等角投影，所以和前述之墨卡托投影（等角投影）相比，不具有方向指示功能。但是它的图像像元坐标和地理坐标之间的映射关系十分简单，所以适合作为全球范围的专题图投影。

## 网络地图规范

在网络地图的发展中出现了3个重要的网络地图规范**WMS**（Web Map Service网络地图服务），**WMTS**（Web Map Tile Service网络地图瓦片服务）,**TMS**（Tile Map Service瓦片地图服务），这些规范事实上决定了现在所有的网络地图的实现方式，本文对这些规范分别进行详细的研究。

### WMS规范

WMS（Web Map Service）（网络地图服务）是OGC（Open Geospatial Consortium）（开放地理空间协会）制定的第一个个网络地图服务接口规范。这个规范的目的是希望不同的网络地图服务提供的数据可以在客户端共同操作。WMS目前存在两个规范，它们是WMS1.1.1和WMS1.3.0。WMS的接口以URL中的query参数的形式来定义。不同的query参数组合对应不同的请求。

#### 接口类型:

WMS规定了3种不同的REQUEST（请求）代表服务器需要提供的3种不同接口，它们分别是：

1. GetCapabilities: 获得服务的能力描述，即服务的元数据，包括服务包含的图层，图层的范围，图层的样式等

2. GetMap: 获得地图图像

3. GetFeatureInfo: 获得指定坐标位置的要素信息

每一个不同的REQUEST都有与之配套的query参数，下面详细的解释每个REQUEST需要的其它query参数。

##### GetMap:

1. 'layers':必选，表示希望获得的图层名，形式是layers=layer\_name1,layer\_name2或者layers=layer\_name 或者layers=\_\_alll\_\_

2. 'styles': 必选表示希望为图层应用的样式名，形式是styles=style\_name1,style\_name2

3. 'srs':必选，表示返回数据的空间参照系，形式是srs=EPSG:3857

4. 'bbox': 必选，表示返回的数据的空间范围，形式是bbox=12312,13123,31312,31231

5. 'width': 必选，表示地图图片的宽度,形式是width=256

6. 'height': 必选，表示地图图片的高度，形式是height=256

7. 'format': 必选，返回的地图图片的格式，允许的参数为('image/png','image/png8', 'image/jpeg')中的一个。

8. 'transparent': 可选，表示返回的地图图片背景是否透明，默认值是False，允许的参数为('TRUE', 'FALSE','true','True','false','False')中的一个。

9. 'bgcolor': 可选，表示返回的地图图片的背景颜色，默认值是none

10. 'exceptions': 可选，表示当服务出现异常时的方案，默认是'application/vnd.ogc.se\_xml', 允许的参数是('application/vnd.ogc.se\_xml', 'application/vnd.ogc.se\_inimage', 'application/vnd.ogc.se\_blank','text/html')中的一个

注意：

1. transparent,format是互相制约的，如果format是不支持透明的图片格式如image/jpeg,那么返回的图像也不支持透明背景。transparent比bgcolor的优先级要高。如果没有指定bgcolor也不是transparent，那么背景颜色的值由实现自行决定。

2. 如果exception的值是application/vnd.ogc.se\_xml指示当服务器发生异常时，将异常消息用一个xml文件返回，如果是application/vnd.ogc.se\_inimage将异常消息绘制在图片上并将图片返回，如果是application/vnd.ogc.se\_blank则返回一个空白图片，如果是text/html则将异常消息用一个html返回。

##### GetFeatureInfo:

1. 'layers': 同GetMap

2. 'styles': 同GetMap

3. 'srs': 同GetMap

4. 'bbox': 同GetMap

5. 'width': 同GetMap

6. 'height': 同GetMap

7. 'format': 同GetMap

8. 'transparent': 同GetMap

9. 'bgcolor': 同GetMap

10. 'exceptions': 同GetMap

11. 'query\_layers': 必选，希望查询数据的图层，形式是，query\_layers=layer\_name1[,lyaer\_name2]\* 或query\_layers=\_\_all\_\_

12. 'info\_format': 必选， 返回的查询数据的格式， 允许的参数是('text/plain', 'text/xml')中的一个。

13. 'feature\_count': 查询的要素的数目， 可选， 默认值是1

14. 'x': 查询的x坐标

15. 'y': 查询的y坐标

16. 'i': 象元x位置（1.3.0）

17. 'j': 象元y位置（1.3.0）

注意：

1. 查询某个位置的要素的信息，可以使用地理坐标或者像元坐标，一般是有后者。

WMS规范，描述了很强的服务器能力。在图像数据方面，对于服务器提供的图层，客户端可以要求返回任意范围，任意图层组合，任意样式组合的图片。在属性数据方法，客户端可以要求返回任意位置任意图层的属性数据。这样的能力，决定了所有的数据只能是在服务端动态计算生成，这在大规模的网络服务环境中是不现实与昂贵的。为了适应大规模网络地图服务的需求，出现了限制的WMS，在图像数据方法，限制了服务器返回的数据范围，使得服务器有可能缓存计算结果，提高服务能力。但是WMS终究只适合小规模的网络地图服务。

### WMTS规范：

WMTS（Web Map Tile Service 网络地图瓦片服务）是OGC（Open Geospatial Consortium）推出的一又个网络地图接口标准。在WMTS出现之前最流行的网络地图接口标准是OGC推出的WMS（Web Map Service），由于WMS在请求时动态渲染地图的特性，它不适用于请求密集的场景。为了克服这个问题，OGC在借鉴其它网络地图厂商方案的基础上，提出了WMTS标准。简单来说，WMTS是一个受限的WMS，它通过限制请求的地图的范围，比例尺，地图的图片格式，图片大小等属性，使得客户端可能请求的地图数量有限。然后WMTS可以预生成这些地图（标准把这些地图称为瓦片），也可以在第一次请求时动态产生这些地图瓦片然后缓存在服务器，客户端或者其它中间服务器中。为了使得缓存策略生效，WMTS还规定了客户的请求编码：KVP（key value pair），RESTful ，SOAP的模板，使得访问相同地图瓦片的请求相同，以利用http协议的缓存策略。

#### WMTS接口规范：

##### KVP:

KVP编码使用键值对列表来表示请求，每个键值对被定义为”参数名=参数值”的形似如“service=WMTS”,键值对列表可以用http的get或者post方法来发送，在get方法中，键值对列表作为url的query参数来发送，在post方法中，键值对列表以消息体的形似发送，每一个键值对在消息体中占一行。

|  |  |  |
| --- | --- | --- |
| GetCapabilities | https://www.testmap.com/maps.cgi?  service=WMTS&  version=1.0.0&  request=GetCapabilities | ServiceMetaDataDocument |
| GetTile | https://www.testmap.com/maps.cgi?  service=WMTS&  request=GetTile&  version=1.0.0&  layer=layer1&  style=default&  format=image/png&  TileMartrixSet=layer1matrixset&  TileMatrix=10m&  TileRow=1&  TileCol=1 | png图片 |
| GetFeatureInfo | https://www.testmap.com/maps.cgi?  service=WMTS&  request=GetTile&  version=1.0.0&  layer=layer1&  style=default&  format=image/png&  TileMartrixSet=layer1matrixset&  TileMatrix=10m&  TileRow=1&  TileCol=1 |  |

##### RESTful

##### SOAP

### TMS规范

TMS（Tile Map Service 瓦片地图服务）是OSGeo(Open Source Geospatial Foundation ，开源地理空间基金会)提出的一个非正式标准。这个标准的提出是基于地图服务厂商已经事实实现的瓦片地图服务，但是它和一般地图厂商使用的瓦片地图服务在原点坐标的选择上不同（TMS选择左下角作为原点，一般网络地图厂商选着左上角作为原点）。TMS规范因为它高性能的特点是现在主流的网络地图实现方案。下文讨论TMS规范的细节。

#### 栅格地图转瓦片地图

TMS网络地图服务最基础的步骤就是瓦片地图生产。完成了瓦片地图生产后，只需把瓦片地图当作一般的网络资源使用成熟的内容分发技术来提供即完成网络地图服务的基础搭建。

下文描述一种瓦片生成技术（球面墨卡托投影）。它首先生成分辨率最高的基础瓦片，然后对基础瓦片应用重采样技术来逐级生成剩余的瓦片。

##### 基础瓦片（比例尺最大的瓦片）生成

###### 分辨率

在0级比例尺下，球面墨卡托投影坐标系下的全球范围被绘制在**256px\*256px**的瓦片上，在不考虑变形的情况下，每个pixel代表的实际距离称为分辨率。它的值计算如下。

1. 球体半径=R=**6378137米**
2. 球体赤道周长 = C=2ΠR=**40075017米**
3. 瓦片宽度=**256**
4. 初始分辨率=initial=**C/256**

根据，n级比例尺的瓦片是4个相邻的n+1级比例尺的瓦片组合得到的性质，

1级比例尺的分辨率 = 0级比例尺的分辨率 / 2 \*\* 1

2级比例尺的分辨率 = 1级比例尺的分辨率 / 2 \*\* 1 = 0级比例尺的分辨率 / 2 \*\* 2

...

res = n级比列尺的分辨率 = 初始分辨率 / 2 \*\* 比例尺等级

###### 获得当前比例尺下瓦片矩阵的行列号范围：

为了保证瓦片的互操作性，即不同服务器产生的瓦片可以在客户端互相叠加，不同区域的图层可以互相拼接，所以需要使用一个全球统一的瓦片编码方式，而不是在每一个图层上应用独立的瓦片编码方式。

前面讨论了，因为瓦片是一个方形的**256px\*256px**的图像而且瓦片金字塔的结构是一个4叉树结构，所以必选选择一个正方形的范围来代表全球。一般会使用前面讨论的球面墨卡托投影，并且把南北的范围限制为赤道周长。在已知比例尺级别和栅格图像左上角坐标和右下角坐标时，我们可以计算出栅格地图，在当前比例尺级别下，左上角瓦片的行列号，和右下角瓦片的行列号，从而也得到了这个瓦片矩阵的行列号范围。

在已知当前比例尺的分辨率的情况下， 我们可以计算当前坐标点，到地图左下端点（TMS标准）（即 （-20037508.342789244，-20037508.342789244））在x轴方向的距离x和在y轴方向的距离y。

x轴象元坐标 = px = x - (-20037508.342789244) / res

y轴象元坐标 = py = y - (-20037508.342789244) / res

因为瓦片的大小是256\*256

x轴瓦片坐标 = math.floor((px / 256))

y轴瓦片坐标 = math.floor((py / 256))

###### 根据瓦片行列号范围生成瓦片图片集

确定了瓦片的行列号范围以后，我们实际得到了这个瓦片矩阵中所有瓦片的行列号。为了生成瓦片图片集，我们只需要生成其中的每一个瓦片图像。具体的操作思路是，根据瓦片的行列号和瓦片的比例尺得到瓦片在投影坐标系中的范围，然后再利用栅格图像的地理变换信息，将瓦片的投影范围转换为栅格图像中的像元范围，然后读出像元数据，重采样到瓦片大小。

1. 根据瓦片的行列号，得到在这一缩放级别下，瓦片左下角的球面墨卡托坐标和右上角的球面墨卡托坐标。

转换方法：

根据瓦片行列号tx,ty计算瓦片的左下角pixel坐标：

pminx = tx \* 256

pminy = ty \* 256

和右上角pixel坐标

pmaxx = (tx + 1) \* 256

pmaxy = (ty + 1) \* 256

然后在根据分辨率，和地图左下角的投影坐标，计算出瓦片角点的投影坐标。

minx = (pminx \* res) - left\_bottom\_x

miny = (pminy \* res) - left\_bottom\_y

maxx = (pmaxx \* res) - left\_bottom\_x

maxy = (pmaxy \* res) - left\_bottom\_y

根据瓦片角点的坐标，从栅格图像中读取对应区域生成新的图像输出，即为瓦片地图。

2.

根据瓦片的角点投影坐标，再利用栅格图像上存储的地理变换信息（图像左上角的投影坐标，每一个栅格象元对应的投影坐标距离），即可以算出需要截取的栅格图像范围。

ulx = minx

uly = maxy

lrx = maxx

lry = miny

ulx对应的x轴投影坐标 = GeoTransform[0]

uly对应的y轴投影坐标 = GeoTransform[3]

单位象元对应的x轴投影距离 = GeoTransform[1]

单位象元对应的y轴投影距离 = GeoTransform[5](负值？)

要读取的栅格范围的左上角图像坐标x = (ulx - GeoTransform[0]) / GeoTransform[1]

要读取的栅格范围的左上角图像坐标y = (uly - GeoTransform[3]) / GeoTransform[5]

要读取的栅格范围的宽度 = rxsize = (lrx - ulx) / GeoTransform[1]

要读取的栅格范围的高度 = rysize = (lry - uly) / GeoTransform[5]

3. 从栅格图像中读取出指定的栅格范围，GDAL允许指定，读出的图像的大小，如果大小和瓦片大小相同则直接输出，否则则需要采用相应的重采样算法进行重采样，使得读出图像的大小和瓦片大小相同。

##### 从大比例尺的瓦片中生成小比例尺的瓦片

容易证明：n级比例尺的（x,y）瓦片是由n+1级比例尺的(2x, 2x+1) X(笛卡尔积) (2y, 2y+1)，这4个瓦片(tx, ty)组合得到的。

这4个瓦片的组合规则如下：

if tx % 2 != 0:

tileposx = tilesize(n+1级瓦片在n级瓦片中的x像元坐标， tilesize是瓦片大小)

else:

tileposx = 0

if ty % 2 != 0:

tileposy = 0

else:

tileposy = tilesize(n+1级瓦片在n级瓦片中的y像元坐标)

确定了n+1级的瓦片和组合规则后，读取并写入得到一幅tilesize \* 2 \* tilesize \* 2的瓦片图像，然后再根据给定的重采样方法，重采样到tilesize \* tilesize的瓦片。

根据从大比例尺到小比例尺的顺序， 由最大比例尺的瓦片图像生成次大一级比例尺的瓦片图像，然后再以此为基础，生成更小比例尺的图像，以此类推，直到生成所有瓦片图像。

上文描述了生成球面墨卡托投影下的瓦片地图的概念方法。

##### 瓦片编码

如何组织瓦片地图也影像着网络地图服务的性能，下面介绍一些瓦片地图编码方法。

###### /z/x/y编码

服务器可以使用文件目录结构来组织瓦片，每一个比例尺的瓦片图像组织在一个目录中，以比例尺的级别作为目录名如1。然后在这个目录中，将列号相同的瓦片图像组织在一个目录中，以列号作为目录名，如1。然后在该目录中，每一个瓦片以行号命名如1。这是一个最直观的组织瓦片的方式。在这个例子中/1/1/1这个位置代表比例尺为1的4个瓦片中位于左下角的那个瓦片。

###### 四叉树编码

编码方法

1. n级比例尺中的瓦片，用n位2进制数分别表示瓦片的行号y和列号x，对x，y进行交错编码，然后将编码的数转为4进制。

以3级比例尺中x=3，y=5的瓦片为例：

x的3位二进制编码 = 011

y的3位二进制编码 = 101

对x,y进行交错编码得到 = 100111

将交错编码转为4进制数 = 213

这里213就是3级比例尺中(3,5)的四叉树编码。

编码性质

1. 编码的长度和瓦片的比例尺等级一致

2. 一个瓦片的四叉树key的前置位(除最后一位)是它的父瓦片的四叉树key。如'231'的父瓦片是'23'; '23'的父瓦片是'2'

3. 在二维笛卡尔坐标系中相邻的瓦片，它们的四叉树key在一维上也相邻。

上面的这些性质都有利于瓦片的存储和读取优化。性质2可以在缩放地图时更快的定位和取回相应的父子瓦片，性质3可以在平移地图时更快的定位和取回相邻位置的兄弟瓦片。

#### 矢量地图转瓦片地图

栅格地图的空间分辨率（每个像元对应的实际距离）是确定，如果为栅格地图应用过大的比例尺，那么将会因为重采样产生模糊的瓦片。矢量地图的空间分辨率（非比例尺）是无限的，所以理论上我们可以应用无限大的比例尺。但是矢量地图转瓦片地图和栅格地图转瓦片地图并无本质区别，下文简述矢量地图转瓦片地图的思路。

1. 将矢量地图投影到球面墨卡托坐标系

2. 确定矢量地图的外包矩形，和将应用的最大比例尺级别

3. 根据外包矩形的范围和最大比例尺的级别计算矢量地图覆盖的瓦片行列号范围

4. 对于范围内的瓦片，根据比例尺和瓦片的行列号反算覆盖的投影范围，然后将矢量地图缩放到给定的范围，并渲染在瓦片图片上

5. 使用和栅格地图相同的方法， 利用基础瓦片生成其它小比例尺的瓦片。

注意：一个瓦片限定的投影坐标范围的纵横比应该为1，否则会出现拉伸变形，导致瓦片无法正确拼接。