引文格式：李国伟,李行义,朱敏华,等. 基于Redis和PostGIS的矢量瓦片动态渲染技术研究[J]. 地理空间信息, 2026, 24(1): 66-70. doi:10. 3969/j.issn. 1672-4623.2026.01.014

基于Redis和PostGIS的矢量瓦片动态渲染技术研究

李国伟[1]，李行义[1]，朱敏华[1]，贾玉明[1]，贺子康[2]
（1. 中国能源建设集团广东省电力设计研究院有限公司，广东 广州 510663；2. 广东天安项目管理有限公司，广东 广州 510663）

摘　要：结合PostGIS和Redis的技术优势，提出了一种矢量瓦片动态渲染技术，以解决传统瓦片技术在存储、传输和渲染效率等方面的不足，特别是针对大规模地理数据和高并发访问环境下的应用场景。利用PostGIS进行矢量数据的动态切片，通过实时生成瓦片避免预生成瓦片的存储压力；采用Redis缓存技术存储频繁请求的瓦片数据，以降低服务器负载。对矢量瓦片渲染系统在实际应用中进行实验验证。结果表明，该技术显著提升了地图服务的加载效率和渲染性能，适用于地理大数据的在线地图服务、智能城市管理等场景，具有广泛的应用前景。初次加载时间平均为60 ms，在缓存机制作用下，后续请求的响应时间降至5 ms以内，表明缓存系统能有效降低服务器压力并提高数据传输效率。

关键词：矢量瓦片；动态渲染；PostGIS；Redis；地理大数据
中图分类号：P208　　**文献标志码：**B　　**文章编号：**1672-4623（2026）01-0066-05

Research on Dynamic Rendering Technology of Vector Tiles Based on Redis and PostGIS

LI Guowei[1], LI Xingyi[1], ZHU Minhua[1], JIA Yuming[1], HE Zikang[2]
(1. China Energy Engineering Group Guangdong Electric Power Design Institute Co., Ltd., Guangzhou 510663, China;
2. Guangdong Tian'an Project Management Co., Ltd., Guangzhou 510663, China)

Abstract: We proposed a dynamic rendering technology of vector tiles based on the strengths of PostGIS and Redis to address the limitations of traditional tile technology, particularly in terms of data retrieval speed, flexibility, and scalability. By leveraging the spatial data storage capabilities of PostGIS and the high-speed data retrieval features of Redis, the proposed technology significantly enhances the performance and scalability of vector tile rendering. Experimental results indicate that this technology significantly improves the loading efficiency and rendering performance of map services, is well-suited for applications such as online map services for geographical large data, smart city management, and other related fields, showing broad potential for future use. The initial load time averages 60 ms, while subsequent requests from caching mechanism, reducing the response time less than 5 ms, demonstrating that the caching system effectively alleviates server load and improves data transmission efficiency.

Key words: vector tile, dynamic rendering, PostGIS, Redis, geographical large data

传统栅格瓦片由于需要预先生成大量图像，占用较大的存储空间，且在传输过程中效率较低，难以满足现代应用对实时性和交互性的高要求[1]。此外，栅格瓦片在高分辨率设备上容易失真，且地图样式固定、无法灵活定制[2]，难以满足用户对个性化地图服务的需求。矢量瓦片技术作为一种高效的数据传输与渲染方法，已成为现代地图服务的核心技术之一[3-4]，在数据存储、传输效率和显示效果上具有显著优势[5]。矢量瓦片通过将矢量数据按空间位置切片，以紧凑的二进制形式存储并传输，显著提升了地图的加载速度和渲染效果，不但能减少预先生成瓦片的存储需求，还能保证地图数据的实时更新，提升了地图的交互性[6-7]。PostGIS作为PostgreSQL数据库的空间扩展模块，具有强大的矢量数据存储和查询功能，支持空间索引和空间操作。通过PostGIS，矢量瓦片可根据请求动态生成，不再依赖于预生成的瓦片库[6]。通过结合PostGIS的实时数据切片功能，地图系统可以更灵活地响应用户需求，提供更高效的地图渲染服务[8-9]。Redis作为一种内存数据库，被广泛应用于高并发、高需求的应用场景中。Redis能将频繁请求的瓦片数据缓存在内存中，避免了数据库的重复查询，降低了数据库负载，从而显著提升系统响应速度[10]。此外，Redis还支持集群模式，能在大规模并发访问时保持高可用性和扩展性。因此，将Redis与PostGIS相结合可有效提高矢量瓦片渲染的效率和用户体验，尤其是在用户访问量大、频率高的应用场景下[11-12]。鉴于

项目来源：中国能建广东院科技资助项目（EV11861W）。
第一作者简介：李国伟（1986—），硕士，工程师，研究方向为电力工程空间地理信息及数字化，E-mail：liguowei@gedi.com.cn。

此，本文提出了一种基于PostGIS和Redis的矢量瓦片动态渲染技术，旨在解决传统瓦片生成方式中存在的存储压力大、传输效率低、交互性能差和显示失真等问题，以满足现代GIS应用中对大规模地理数据高效、动态渲染和个性化显示的需求。

1　PostGIS助力的矢量瓦片实现

1.1　矢量瓦片数据技术

矢量瓦片技术涵盖数据标准[13-14]、组织模型、几何与属性编码以及动态切片与渲染等方面。数据标准多采用Mapbox矢量瓦片（MVT）标准[15]，以保障数据紧凑、传输高效和跨平台兼容；组织模型通过分级金字塔结构划分地理数据层级，并结合数据压缩与缓存机制，提升存储和传输效率；几何编码将地理要素空间信息转化为二进制，以屏幕坐标的形式存储，采用相对坐标系统，属性编码以键值对存储非空间数据，并采用Protobuf压缩[16]，客户端接收后解析结合样式表渲染；动态切片与渲染技术作为现代地图服务的核心[17]，结合PostGIS数据存储、动态免切片技术、Redis缓存设计和高效渲染机制，在大规模数据处理和多用户并发访问时，可实现地图数据实时更新与灵活展示，满足用户对高精度和高响应速度的需求。

1.2　PostGIS的矢量数据存储

PostGIS为矢量数据的存储和管理提供了强大的工具和功能[18]。在矢量瓦片技术体系中，空间数据类型可存储各类几何信息[19]。不仅能通过标准SQL语句查询这些数据，而且支持空间索引，可显著提高地理数据的查询速度和效率[20]。同时，PostGIS提供了ST_Intersects、ST_Within、ST_Transform、ST_AsMVTGeom和ST_AsMVT等空间函数，用于复杂的空间分析和数据处理。在生成矢量瓦片的过程中，PostGIS中的矢量数据能根据需求进行动态查询和切片处理，确保每次请求都能获取最新的地理信息，以实现矢量瓦片的动态生成和实时更新。

1.3　矢量瓦片动态免切片技术流程

矢量瓦片动态免切片技术的具体操作流程为：当客户端发出地图请求时，服务器首先根据请求的瓦片范围，通过PostGIS检索出对应的矢量数据[21]；然后利用动态切片算法实时生成与请求范围对应的瓦片数据，瓦片数据通常采用MVT格式；最后将生成的瓦片数据直接发送到客户端，无需进行额外存储。动态免切片技术能根据实时数据生成瓦片，确保地图内容的时效性，同时减少存储需求，提升系统的灵活性和响应速度。

2　Redis驱动的矢量瓦片优化策略

2.1　矢量瓦片缓存优化

为进一步提升瓦片的生成和传输效率，矢量瓦片系统通常引入Redis技术缓存瓦片数据和优化渲染机制。瓦片数据以键值对的形式存储在Redis中，键通常由瓦片的唯一标识符（如层级、行号、列号等XYZ组合）构成，值则是对应的MVT格式数据。当客户端发出瓦片请求时，系统首先查询Redis缓存，若缓存中已有对应数据，则直接返回缓存数据[22]；若没有，则实时生成瓦片并将其存储到Redis中，以备后续请求使用。通过这种缓存机制，系统可大幅减少瓦片的生成时间和数据库压力。

2.2　矢量瓦片渲染机制优化

矢量瓦片渲染机制是将编码后的瓦片数据转换为可视化地图的关键过程。客户端接收到MVT格式的瓦片数据后，首先解析瓦片中的几何和属性信息；再基于预定义的样式表，利用渲染引擎（WebGL等）将这些信息转化为最终的地图显示效果。渲染过程中允许用户在不同缩放级别下浏览地图，同时进行缩放、平移和属性查询等动态交互。矢量瓦片的灵活性使得地图样式能根据不同的用户需求进行调整，提供定制化的地图体验。此外，得益于WebGL等高性能图形技术，结合PostGIS的矢量数据存储、动态免切片技术流程、Redis缓存设计和高效渲染机制，在浏览器或移动设备上实现了矢量瓦片的快速流畅渲染，确保用户在各种设备和网络环境下都能获得优质的地图服务。

3　Redis与PostGIS技术集成分析

3.1　集成优势分析

PostGIS与Redis的结合为矢量瓦片动态渲染带来了多方面优势。在数据存储与管理方面，PostGIS的空间数据处理能力确保了矢量数据的高效组织与快速检索，动态生成矢量瓦片，减少了存储负担，保证了时效性；Redis通过将频繁访问的瓦片数据存储在内存中，缩短数据获取时间，降低查询压力，提升响应速度，尤其是在高并发场景下，能有效维持系统的稳定性和高效性。在地图渲染过程中，PostGIS提供的准确数据为渲染提供了基础，而Redis的快速数据供应使

得渲染引擎能及时获取所需瓦片数据，结合WebGL等高性能渲染技术，实现了流畅的地图显示和动态交互效果。

3.2 数据一致性问题

PostGIS中的数据可能会更新，而Redis缓存中的数据若不能同步，将导致用户获取的数据不一致，如当地理要素的属性或位置在PostGIS中修改后，缓存中的旧瓦片数据仍可能被提供给客户端。因此，采用数据更新通知机制，若PostGIS中的数据发生变化，则触发一个事件通知相关的缓存更新模块，根据更新的数据范围，确定受影响的瓦片，删除Redis中对应的缓存数据，确保下次请求这些瓦片时，从PostGIS重新生成最新的瓦片数据，从而保证数据的一致性。

3.3 缓存淘汰策略优化

随着瓦片请求的不断增加，Redis缓存空间可能会被填满。如何选择合适的缓存淘汰策略，确保缓存中存储的是最有价值的瓦片数据，是一个关键问题。因此，需要综合考虑瓦片的访问频率、最近访问时间和瓦片数据大小等因素设计缓存淘汰策略。例如，可以采用改进的最近最少使用算法，为每个瓦片设置一个权重，权重由访问频率和最近访问时间共同决定，数据量大的瓦片权重适当降低，优先淘汰权重低的瓦片。

3.4 空间索引与缓存的协同

PostGIS的空间索引在加速数据查询方面具有重要作用。如何确保空间索引与Redis缓存机制的有效协同，避免在缓存未命中时重复进行不必要的索引查询，是提高系统性能的关键。本文通过建立缓存与空间索引的关联机制来解决该问题。在查询空间索引前，检查缓存中是否存在与查询条件相关的部分结果或元数据，若存在，则可利用这些数据对空间索引查询进行优化，缩小查询范围；并在将查询结果存入缓存时记录相关的索引信息，以便后续查询时能快速判断是否可以利用缓存数据进行优化，提高空间索引与缓存的协同效率，减少查询时间。

4 实验与结果分析

4.1 实验数据与环境搭建

本文选取广东省范围内的陆上风电机、光伏电板和输电线路3种地理数据图层作为实验数据，采用PostGIS的shp文件导入工具，将原始shp数据成功导入PostGIS空间数据库内，并精准配置相应的坐标系参数，完成了空间数据与属性数据的结构化存储，数据总量为652 MB，包含459 054条记录（表1）。实验数据覆盖了广东省范围内的地理信息，包括点、线、面3种几何类型数据，确保了实验结果的普适性和可靠性。

表1 实验采用的测试数据

测试数据图层	数据类型	数据量/条	数据大小/MB
陆上风电机	点	118 832	180
输电线路	线	156 358	328
光伏电板	面	183 864	144

实验服务器端硬件搭载13代Intel(R) Core(TM) i9-13900处理器，主频高达2.00 GHz，内存容量达到64 GB，操作系统采用Windows10企业版。数据库采用PostgreSQL v12.9版本和PostGIS v3.1版本，为地理空间数据处理提供了强大支持。此外，通过Node.js v18.17.1和Vue.js v3.3.8等前端技术以及ArcGIS API v4.23.7地图引擎为实验提供了全面的技术支持。

4.2 客户端渲染性能分析

基于B/S架构，采用Spring Boot和Vue3+ArcGIS API设计开发系统，通过对3类不同数据进行矢量数据切片、缓存和加载，分析了两次加载过程中响应时间的差异。结果表明，初次加载时，由于需要从PostGIS数据库中检索原始矢量数据并进行切片，加载时间较长。以陆上风电机为例，在9/412/226层级下，瓦片的初次加载时间需要60 ms；而当数据进入Redis缓存后，第二次加载响应时间显著缩短，仅需约5 ms。两次矢量瓦片加载时间的对比见表2。随着缩放层级的增加，瓦片的数据量逐渐减少，加载时间随之缩短。在高层级（Z10及以上）下，缓存加载时间基本稳定在5 ms以内，表明缓存机制在高频请求的环境下能够维持稳定的响应速度。此外，随着缩放层级的下降，矢量瓦片中的数据量增加，导致初次加载的响应时间逐渐延长。即使在数据量较大的情况下，缓存机制仍能保持较短的响应时间，充分验证了该系统在复杂地理数据环境下的稳定性和高效性。

在缓存机制的支持下，实验进一步测试了不同图层（图1）、不同缩放层级（表3）下的缓存响应性能。以第6~16层的缩放级别为例，缓存响应时间与瓦片数据量呈现出良好的对应关系，即使在较大的数据量（陆上风电机L6层）下，缓存加载的平均响应时间仍保持在25.8 ms；在所有图层的联合请求下，即使是L16层的瓦片请求，响应时间也能维持在17.5 ms，表明缓存系统对大规模矢量数据的处理性能具有良好的适应性和扩展性。

表2　矢量瓦片初次和缓存切片加载时间

瓦片层级行列号Z/X/Y	陆上风电机图层			输电线路图层			光伏电板图层		
	初次加载响应时间/ms	缓存加载响应时间/ms	瓦片大小/KB	初次加载响应时间/ms	缓存加载响应时间/ms	瓦片大小/KB	初次加载响应时间/ms	缓存加载响应时间/ms	瓦片大小/KB
6/52/28	131	9	31.07	732	56	239.92	705	49	34.96
7/103/56	114	8	22.06	184	17	60.78	688	18	25.50
8/206/113	93	5	6.26	84	5	7.22	515	6	9.24
9/412/226	60	5	0.26	74	5	6.89	473	5	7.31
10/824/452	54	5	0.19	39	5	2.39	245	6	0.96
11/1648/904	60	5	0.19	39	5	0.54	114	5	0.46
12/3296/1808	53	5	0.19	38	5	0.19	95	5	0.19
13/6592/3616	54	5	0.19	38	5	0.19	93	5	0.19
14/13184/7232	61	5	0.19	39	5	0.19	93	5	0.19
15/26368/14464	54	5	0.19	38	5	0.19	93	5	0.19
16/52736/28928	54	5	0.19	38	5	0.19	110	5	0.19
17/105472/57856	58	5	0.19	39	5	0.19	93	5	0.19
18/210944/115712	69	5	0.19	39	5	0.19	93	5	0.19

表3　缓存矢量瓦片响应速度及数据量统计

地图缩放层级	陆上风电机图层		输电线路图层		光伏电板图层		所有图层	
	瓦片平均大小/KB	平均缓存响应时间/ms	瓦片平均大小/KB	平均缓存响应时间/ms	瓦片平均大小/KB	平均缓存响应时间/ms	瓦片平均大小/KB	平均缓存响应时间/ms
L6	55.9	25.8	436.0	21	77.6	28.0	569.0	40.1
L7	57.4	27.0	520.0	24	79.8	29.5	665.0	42.4
L8	41.1	18.5	533.0	26	55.0	24.0	651.0	46.6
L9	29.9	17.4	365.0	20	35.9	22.0	392.0	38.8
L10	22.9	16.1	241.0	22	33.3	21.0	255.0	28.5
L11	14.2	13.7	58.8	18	22.0	20.0	66.0	25.4
L12	13.5	12.6	25.3	16	15.0	17.7	28.3	27.7
L13	8.9	14.6	9.8	17	8.0	16.9	10.0	26.5
L14	5.1	12.5	3.3	15	5.8	12.3	5.5	22.1
L15	2.5	11.0	1.2	12	1.8	11.0	1.2	21.9
L16	2.2	10.5	0.9	11	1.1	10.0	0.9	17.5

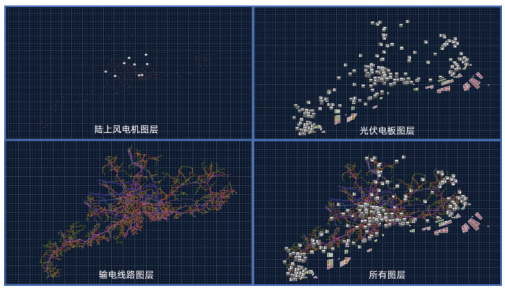


图1　不同图层加载渲染可视化

4.3　客户端查询性能分析

矢量瓦片通过将几何图形数据与属性信息相结合，生成包含详细几何特征（类别、名称等）的属性（图2），从而提升矢量数据的查询效率。通过矢量数据的属性信息来储存表格数据，可显著拓宽数据查询的信息覆盖面，提高查询结果的全面性和精确度。地图图层按特定顺序叠加，矢量数据图层可覆盖在栅格影像的底图上，实现了多种数据类型的综合表达，不仅丰富了矢量数据表现，还增强了场景的呈现效果。通过建立更全面的环境认知，并结合设备设施的属性信息，能源管理人员查询时能更全面地确认各设备和地理要素的空间位置和状态。

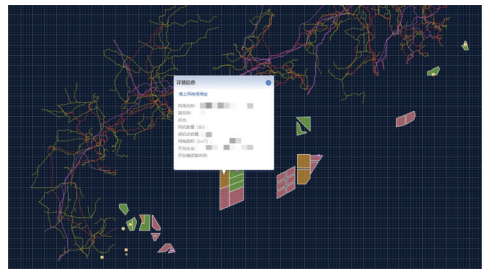


图2　交互查询示意图（局部）

客户端的查询性能对比见表4，可以看出，缓存机制显著提升了查询效率，效果明显；虽然各图层的缓存查询时间存在一定波动，但相较于初次查询时间，波动范围较小，缓存查询时间相对稳定。

表4 查询性能对比

陆上风电机图层			输电线路图层			光伏电板图层		
点查经纬度	初次查询时间/ms	缓存查询时间/ms	点查经纬度	初次查询时间/ms	缓存查询时间/ms	点查经纬度	初次查询时间/ms	缓存查询时间/ms
111.48, 20.99	131	48	110.21, 21.22	362	44	110.06, 21.37	596	79
112.25, 21.48	583	79	110.85, 21.65	256	33	110.16, 21.19	277	57
112.23, 20.98	262	53	110.01, 20.67	331	37	110.31, 21.08	272	41

5 结 语

本文提出了一种基于PostGIS和Redis的矢量瓦片动态渲染方法，并通过实验验证了该方法在地理大数据环境中的有效性。该方法充分利用PostGIS的实时空间数据管理能力和Redis的高效缓存机制，构建了一个可动态生成、存储和渲染矢量瓦片的系统。结果表明，该系统在复杂多用户并发环境中能持续保持较高的响应速度，显著提升了地图服务的加载效率和查询性能；尤其是在多图层叠加和大数据量请求场景下，缓存机制有效减轻了服务器负载，无论是初次加载还是缓存加载，系统的响应时间均能满足现代GIS应用需求，为实时地图渲染和动态交互提供了坚实的技术支持，整体性能得到提高。

本文展示了PostGIS和Redis的矢量瓦片技术在GIS中的巨大应用潜力，应用场景还可扩展至大数据可视化、智能城市管理等领域，为GIS技术的发展提供了更多可能性。未来可以进一步优化矢量瓦片的数据压缩和传输技术，以应对更大规模的地理数据处理需求。随着WebGL等高性能渲染技术的快速发展，未来可探索在更高分辨率设备上提升渲染效果的同时，增强用户交互的灵活性，进一步优化整体使用感受。

参考文献

[1] 方登茂,张晓平,刘梁. 基于PostGIS的矢量瓦片按需组装技术研究[J]. 城市勘测,2021(5):65-69

[2] 刘杰,魏娟. 矢量地图渲染技术研究[J]. 地理空间信息,2021,19(11):128-130

[3] 杨士乐,邹文明,陶彪,等. 矢量瓦片技术体系的地理信息公共服务方法研究[J]. 测绘科学,2020,45(10):168-172

[4] 聂沛,陈广胜,景维鹏. 矢量瓦片并行构建与分布式存储模型研究[J]. 地球信息科学学报,2020,22(7):1 487-1 496

[5] 毕晓玲,任致华,申振,等. 矢量瓦片技术在生态保护红线“一张图”中的应用研究[J]. 环境生态学,2021,3(1):23-28

[6] 唐雅玲,马晨燕,宋佳,等. 矢量Web网络地图应用研究[J]. 测绘与空间地理信息,2018,41(2):77-80

[7] 范梦琪,宋伟东,郑人维,等. 基于矢量瓦片技术的Web电子海图优化方法[J]. 海洋科学,2021,45(2):68-75

[8] 胡晓曦,蔡思腾,刘刚,等. 基于MVT的矢量瓦片免切片技术[J]. 河北省科学院学报,2024,41(2):68-73

[9] 周锐,王彬. 基于开源软件的校园地理信息系统开发[J]. 武汉大学学报(理学版),2012,58(增刊1):376-380

[10] 刘胜杰,倪慧珠,郭见兵,等. 一种基于Redis的矢量数据快速加载方法[J]. 杭州师范大学学报(自然科学版),2022,21(5):530-535

[11] 魏林风. 基于Redis的实时WebGIS系统设计与实现[J]. 通讯世界,2017,24(22):18-20

[12] 戚将辉,张丰,杜震洪,等. 基于内存数据库的矢量数据存储与空间索引研究[J]. 浙江大学学报(理学版),2015,42(3):365-370

[13] 朱秀丽,周治武,李静,等. 网络矢量地图瓦片技术研究[J]. 测绘通报,2016(11):106-109

[14] 王超,李佩虎,王德清,等. 基于矢量瓦片的交互式警用电子地图在线制图与服务系统研究[J]. 中国安全防范技术与应用,2018,(6):63-68

[15] 徐瑞. 基于WebGL的三维GIS在城市规划中的应用研究[D]. 武汉:武汉大学,2018

[16] 兰悦. 分布式小对象存储系统元数据管理的设计与实现[D]. 成都:电子科技大学,2021

[17] 魏祖宽,胡娟,金在弘. WebGIS混合缓存的应用与研究[J]. 计算机系统应用,2009,18(9):144-148

[18] 曾侃. 基于开源数据库PostgreSQL的地理空间数据管理方法研究[D]. 杭州:浙江大学,2007

[19] 周玉科,周成虎,陈荣国,等. 一种网络空间数据发布与在线处理平台的设计与实现[J]. 地球信息科学学报,2011,13(4):486-491

[20] 陈文韬,特日根. 基于PostGIS的遥感数据服务系统检索研究[J]. 测绘与空间地理信息,2021,44(7):78-83

[21] 王文杰,王乃生,金鼎,等. 网络地图在桌面端应用中的浏览与输出[J]. 测绘技术装备,2018,20(4):50-53

[22] 任彦丞,范超. 瓦片地图技术在可视化基站管理系统中的应用研究[J]. 数字技术与应用,2014,32(12):85-86