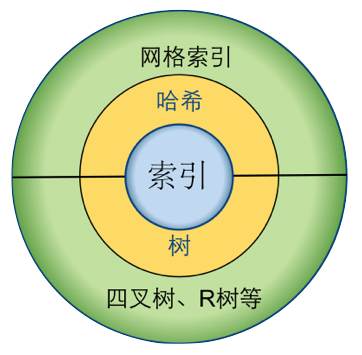
第一篇讲到了传统的索引如B树不能很好的支持空间数据，比如点（POI等）、线（道路、河流等）、面（行政边界、住宅区等）。本篇将对空间索引进行简单分类，然后介绍网格索引。（深入浅出空间索引1：<http://blog.csdn.net/zhanlijun/article/details/13171539>）

一、空间索引有哪几种？

　　传统索引使用哈希和树这两类最基本的数据结构。空间索引虽然更为复杂，但仍然发展于这两种数据结构。因此可以将空间索引划分为两大类：1）基于哈希思想，如网格索引等；2）基于树思想，有四叉树、R树等。



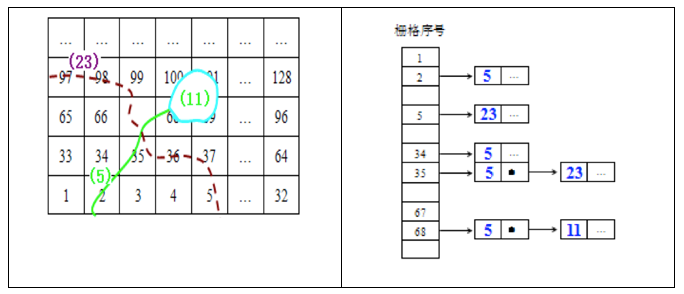
二、网格索引

　　哈希是通过一个哈希函数将关键字映射到内存或外存的数据结构，如何扩展到空间数据呢？

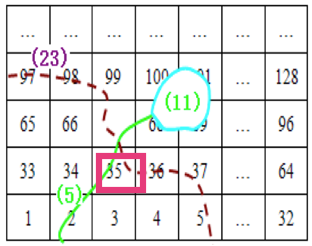
2.1. 网格索引原理

　　扩展方法：对地理空间进行网格划分，划分成大小相同的网格，每个网格对应着一块存储空间，索引项登记上落入该网格的空间对象。

　　举个例 子，我们将地理空间进行网格划分，并进行编号。该空间范围内有三个空间对象，分别是id=5的街道，23的河流和11的商圈。这时候我们可以按照哈希的数 据结构存储，每个网格对应着一个存储桶，而桶里放着空间对象，比如对2号网格，里面存储着id=5的空间对象，对35号网格，桶里放着id=5和 id=23的空间对象。



　　假如我们要查询某一空间范围内有哪些空间对象，比如下面的红框就表示空间范围，我们可以很快根据红框的空间范围算出它与35号和36号网格相交，然后分别到35号和36号网格中查找空间对象，最终找出id=5和id=23的空间对象。



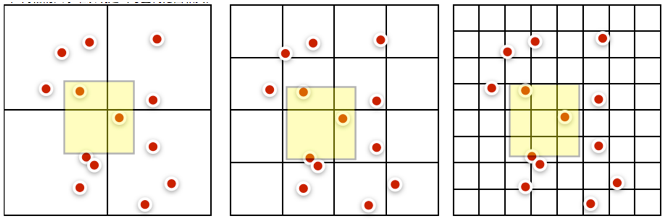
2.2. 网格索引缺点

1）索引数据冗余

　　网格与对象之间多对多关系在空间对象数量多、大小不均时造成索引数据冗余。比如11号商圈这个空间对象在68，69，100，101这4个网格都有存储，浪费了大量空间。

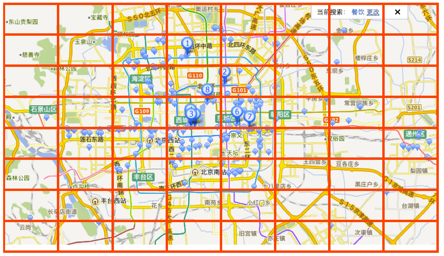
2）网格的大小难以确定

　　网格的划分大小难以确定。网格划分得越密，需要的存储空间越多，网格划分的越粗，查找效率可能会降低。对于图a，这个查询需要查询4个网格，由于4个网格覆盖了整个空间，因此这个查找其实是将空间范围内所有的点数据都遍历一遍，失去了索引的意义。



3）很多网格没有数据

　　空间数据具有明显的聚集性，比如POI只在几个热点商贸区聚集，在郊区等地方很稀疏，这将导致很多网格内没有任何空间数据。



来源： <<http://blog.csdn.net/zhanlijun/article/details/14103633>>