

联合索引又叫复合索引，对于复合索引：mysql从左到右的使用索引中的字段，一个查询可以只使用索引中的一部分，但只能是最左侧部分。例如索引是key index(a,b,c)。可以支持a | a,b|a,b,c3种组合进行查找，但不支持b, c进行查找。当最左侧字段是常量引用，索引就十分有效。

创建复合索引时，应该仔细考虑列的顺序。对索引中的所有列执行搜索或者仅对前几列执行搜索时，复合索引非常有用；仅对后面的任意列执行搜索时，复合索引则没有用处。

```
CREATE TABLE `union_test` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `a` varchar(8) COLLATE utf8_unicode_ci NOT NULL,  
  `b` varchar(8) COLLATE utf8_unicode_ci NOT NULL,  
  `c` varchar(8) COLLATE utf8_unicode_ci NOT NULL,  
  `createTime` datetime NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `union_index` (`a`,`b`,`c`) USING BTREE  
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;  
  
-- records  
INSERT INTO `union_test` VALUES ('1','a','b','c','2017-08-16 10:41:48');  
INSERT INTO `union_test` VALUES ('2','aa','bb','cc','2017-08-16 10:41:56');
```

测试结果

```
EXPLAIN SELECT a,b,c FROM union_test WHERE a = "a" AND b = "b" AND c = "c" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE a = "a" AND b LIKE "b%" AND c LIKE "c%" -- 走索引，这里三个变量如果第一个是like就不走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE a = "a" AND c = "c" AND b = "b" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE b = "b" AND a = "a" AND c = "c" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE b = "b" AND c = "c" AND a = "a" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE c = "c" AND a = "a" AND b = "b" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE c = "c" AND b = "b" AND a = "a" -- 走索引  
  
EXPLAIN SELECT a,b,c FROM union_test WHERE a = "a" AND b = "b" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE b = "b" AND a = "a" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE a = "a" AND c = "c" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE c = "c" AND a = "a" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE b = "b" AND c = "c" -- 不走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE c = "c" AND b = "b" -- 不走索引  
  
EXPLAIN SELECT a,b,c FROM union_test WHERE a = "a" -- 走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE a LIKE "a%" -- 不走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE b = "b" -- 不走索引  
EXPLAIN SELECT a,b,c FROM union_test WHERE c = "c" -- 不走索引
```

个人看法

这里可以看出如果是联合索引的条件都存在的话，顺序就不是问题了，似乎我之前理解的最左索引原则有错误，现在想想，如果条件里面含有联合索引的第一个索引字段，那管where条件的顺序和个数，则这次查询都会走索引。上述如果使用了like，比如这样SELECT a,b,c FROM union_test WHERE a LIKE "a%" AND b = "b%" AND c = "c%"致索引失效，单个索引的后半模糊不会导致索引失效，这里却导致了索引失效，我没想明白为什么，还是说如果是联合索引，最左的索引字段就一定要精确匹配呢？如果大神还望告知，谢谢。

----后记

mysql中一般使用的都是B+树来作为索引的数据存储结构，所以一定要有一个索引的引导列，应该也就是B+树的root节点，只有根节点存在才可以使用到联合索引。