Atitit sql计划任务与查询优化器--统计信息模块

每一个统计信息的内容都包含以上三部分的内容。

我们依次来分析下，通过这三部分内容SQL Server如何了解该列数据的内容分布的。

**a、统计信息的总体属性项**

该部分包含以下几列：

* Name:统计信息的名称。
* Updated:统计信息的最近一次更新时间，这个时间信息很重要，根据它我们能知道该统计信息什么时候更新的，是不是最新的，是不是存在统计信息更新不及时造成统计的当前数据分布不准确等问题。
* Rows:描述当前表中的总行数。
* Rows Sampled:统计信息的抽样数据。当数据量比较多的时候，统计信息的获取是采用的抽样的方式统计的，如果数据量比较就会通过扫描全部获取比较精确的统计值。比如，上面的例子中抽样数据就为91行。
* Steps:步长值。也就是SQL Server统计信息的根据数据行的分组的个数。这个步长值也是有SQL Server自己确定的，因为步长越小，描述的数据越详细，但是消耗也越多，所以SQL Server会自己平衡这个值。
* Density:密度值，也就是列值前缀的大小。
* Average Key length:所有列的平均长度。
* String Index:表示统计值是否为字符串的统计信息。这里字符串的评估目的是为了支持LIKE关键字的搜索。
* Filter Expression：过滤表达式，这个是SQL Server2008以后版本的新特性，支持添加过滤表达式，更加细粒度进行统计分析。
* Unfiltered Rows：没有经过表达式过滤的行，也是新特性。

经过上面部分的数据，统计信息已经分析出该列数据的最近更新时间、数据量、数据长度、数据类型等信息值。

**b、统计信息的覆盖索引项**

All density：反映索引列的稠密度值。这是一个非常重要的值，SQL Server会根据这个评分项来决定该索引的有效程度。

该分值的计算公式为：density=1/表中非重复的行数。所以该稠密度值取值范围为：0-1。

该值越小说明该列的索引项选择性更强，也就说该索引更有效。理想的情况是全部为非重复值，也就是说都是唯一值，这样它的数最小。

举个例子：比如上面的例子该列存在91行，假如顾客不存在重名的情况下，那么该密度值就为1/91=0.010989，该列为性别列，那么它只存在两个值：男、女，那么该列的密度值就为0.5，所以相比而言SQL Server在索引选择的时候很显然就会选择ContactName（顾客名字）列。

简单点讲：就是当前索引的选择性高，它的稠密度值就小，那么它就重复值少，这样筛选的时候更容易找到结果值。相反，重复值多选择性就差，比如性别，一次过滤只能过滤掉一半的记录。

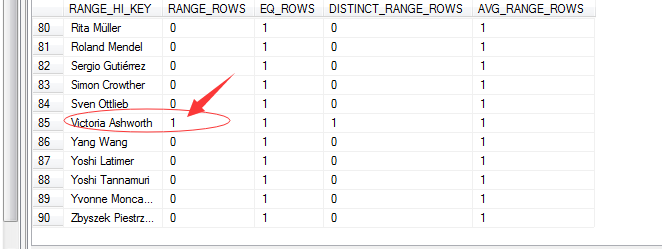
Average Length:索引的平均长度。

Columns:索引列的名称。这里因为我们是非聚集索引，所以会存在两行，一行为ContactName索引列，一行为ContactName索引列和聚集索引的列值CustomerID组合列。希望能明白这里，索引基础知识。

通过以上部分信息，SQL Server会知道该部分的数据获取方式那个更快，更有效。

**c、统计信息的直方图信息**

我们接着分析第三部分，该列直方图信息，通过这块SQL Server能直观“掌控”该列的数据分布内容，我们来看

* RANGE\_HI\_KEY：直方图中每一组数据的最大值。这个好理解，如果数据量大的话，经过分组，这个值就是当前组的最大值。上面例子的统计信息总共分了90组，总共才91行，也就是说，SQL Server为了准确的描述该列的值，大部分每个组只取了一个值，只有一个组取了俩值。
* RANGE\_ROWS：直方图的没组数据的区间行数（不包括最大值）。这里我们说了总共就91行，它分了90组，所以有一组会存在两个值，我们找到它：
* EQ\_ROWS：这里表示和上面最大值相等的行数目。因为我们不包含一样的，所以这里值都为 1
* DISTINCT\_RANGE\_ROWS：直方图每组数据区间的非重复值的数目。上限值除外。
* AVG\_RANGE\_ROWS：每个直方图平均的行数。

经过最后一部分的描述，SQL Server已经完全掌控了该表中该字段的数据内容分布了。想获取那些数据根据它就可以从容获取到，并且统计信息是排序了的。

所以当我们每次写的T-SQL语句，它都能根据统计信息评估出要获取的数据量多少，并且找到最合适的执行计划来执行。

我也相信经过上面三部分的分析，关于文章开篇我们提到的那个关于‘K’和‘Y’的问题会找到答案了，这里不解释了。

当然，如果数据量特别大，统计信息的维护也会有小小的失误，而这时候就需要我们来站出来及时的弥补。

**创建统计信息**

通过上面的介绍，其实我们已经看到了统计信息的强大作用了，所以对于数据库来说它的重要性就不言而喻了，因此，SQL Server会自动的创建统计信息，适时的更新统计信息，当然我们可以关闭掉，但是我非常不建议这么做，原因很简单：No Do  No Die...

这两项功能默认是开启的，也就是说SQL Server会自己维护统计信息的准确性。

在日常维护中，我们大可不必要去更改这两项，当然也有比较极端的情况，因为我们知道更新统计信息也是一个消耗，在非常的大的并发的系统中需要关掉自动更新功能，这种情况非常的少之又少，所以基本采用默认值就可以。

在以下情况下，SQL Server会自动的创建统计信息：

1、在索引创建时，SQL Server会自动的在索引列上创建统计信息。

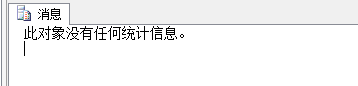
2、当SQL Server想要使用某些列上的统计信息，发现没有的时候，这时候会自动创建统计信息。

3、当然，我们也可以手动创建。

比如，自动创建的例子

select \* into CustomersStats from Customers

sp\_helpstats CustomersStats



来添加一个查询语句，然后再查看统计信息

select \* from CustomersStatswhere ContactName='Hanna Moos'

go

sp\_helpstats CustomersStats

go

在以下情况下，SQL Server会自动的更新统计信息：

 1、如果统计信息是定义在普通的表格上，那么当发生以下任一种的变化后，统计信息就会被触发更新动作。

* 表格从没有数据变成大于等于1条数据。
* 对于数据量小于500行的表格，当统计信息的第一个字段数据累计变化大于500以后。
* 对于数据量大于500行的表格，当统计信息的第一个字段数据累计变化大于500+（20%\*表格总的数据量）以后。所以对于较大的表，只有1/5以上的数据发生变化后，SQL Server才会重新计算统计信息。

2、临时表上也可以有统计信息。这也是很多情况下采用临时表优化的原因之一。其维护策略基本和普通表格一样，但是表变量不能创建统计信息。

当然，我们也可以手动的更新统计信息，更新脚本如下：

UPDATE STATISTICS Customers WITH FULLSCAN

SQL Server调优系列进阶篇（深入剖析统计信息） - 指尖流淌 - 博客园.html

作者:: 绰号:老哇的爪子claw of Eagle 偶像破坏者Iconoclast image-smasher

捕鸟王"Bird Catcher 王中之王King of Kings 虔诚者Pious 宗教信仰捍卫者 Defender of the Faith. 卡拉卡拉红斗篷 Caracalla red cloak

简称：： Emir Attilax Akbar 埃米尔 阿提拉克斯 阿克巴

全名：：Emir Attilax Akbar bin Mahmud bin attila bin Solomon Al Rapanui

埃米尔 阿提拉克斯 阿克巴 本 马哈茂德 本 阿提拉 本 所罗门  阿尔 拉帕努伊

常用名：艾提拉（艾龙），   EMAIL:1466519819@qq.com

转载请注明来源：attilax的专栏  http://blog.csdn.net/attilax

--Atiend