

## Kirk's Korner



### Quick & Simple Tips

Kirk Paul Lafler, Software Intelligence Corporation

## Tip: Exploring the World of Indexes

SAS users always seem interested in learning about techniques related to speeding up data access. One way of improving information retrieval from a SAS data set or table is to explore the world of indexes.

An index consists of one or more columns that are used to uniquely identify each row within a table. Functioning as a SAS object containing the values in one or more columns in a table, an index may be defined as numeric, character, or a combination of both. Although there is no rule that says a table has to have an index, when present, an index is most frequently used to make information retrieval more efficient.

### Determining when an Index is Necessary

To help determine when an index is necessary, it is important to understand the existing data as well as the way the base table(s) will be used. It is always important to know what queries will be used and how users will access columns of data. If an index is used to specify some order within a table, such as movie title or movie category in the MOVIES table, care should be exercised to fully assess what the full impact of that index will be.

There are times when the column(s) making up an index are obvious and other times when they are not. When determining whether an index provides any value, some very important rules should be kept in mind. An index should permit the greatest flexibility so every column in a table can be accessed and displayed. Indexes should also be assigned to discriminating column(s) only since query results will benefit greatest when this is the case.

### Rules Associated with Index Creation

When deciding when to create an index there are a few rules that SAS users should be aware of. First, an index should only be created when it is absolutely necessary. The creation of too many, or unnecessary, indexes require the expenditure of an unwarranted amount of computer resources for their maintenance. Since an index has to be updated any time a DELETE, INSERT, or UPDATE is performed on rows in a table, care should be used when deciding what indexes should be created.

Second, if the table is small sequential processing may be just as fast, or faster, than processing with an index.

Next, when the number of matches or a subset is 15% or less of the rows in the population then an index should be considered because the PROC SQL processor may take advantage of any defined indexes. This is due to the way WHERE clause processing is performed in the SAS System. As the number of rows in a subset grows in size, information retrieval may become slower than for smaller subsets consisting of fewer rows.

Finally, if the page count as displayed in the CONTENTS procedure is less than 3 pages, avoid creating or using an index.

## Types of Indexes

Two types of indexes can be defined in PROC SQL: *simple* and *composite*. When a simple index is created, it only references a single column. In contrast, a composite index references two or more columns in a table.

### Creating a Simple Index

A simple index is specifically defined for one column in a table and must be the same name as the column. Suppose you wanted to create an index consisting of movie category (CATEGORY) in the MOVIES table. Once created, the index becomes a separate object located in the specified SAS library.

#### SAS Code

```
PROC SQL;  
  CREATE INDEX CATEGORY  
    ON libref.MOVIES;  
QUIT;
```

### Creating a Composite Index

A composite index is defined for two or more columns in a table and must be specified as a different name from the column. Suppose you wanted to create an index consisting of the movie category (CATEGORY) and the movie rating (RATING) located in the MOVIES table. Users should be aware that only one composite index is allowed per set of columns. The index, as with the simple index, becomes a separate object located in the designated SAS library.

#### SAS Code

```
PROC SQL;  
  CREATE INDEX CATRATE  
    ON libref.MOVIES (CATEGORY, RATING);  
QUIT;
```

### Modifying Columns Containing Indexes

Altering the attributes of a column that contains an associated index (simple or composite) does NOT prohibit the values in the altered column from using the index. But, if a column that contains an index is dropped, then the index is also dropped. Accordingly, when a column is dropped, any data in that index is also lost.

### Removing an Index

When one or more indexes are no longer needed, the DROP INDEX statement can be used to remove them. Suppose you determined that due to poor performance that the composite index CATRATE (created earlier) is no longer needed.

#### SAS Code

```
PROC SQL;  
  DROP INDEX CATRATE  
    FROM libref.MOVIES;  
QUIT;
```

## Contact Information

If you would like more information or have any questions about this tip, please contact: Kirk Paul Lafler, Software Intelligence Corporation at [KirkLafler@cs.com](mailto:KirkLafler@cs.com). Kirk has been working with the SAS System since 1979 and is a SAS Certified Professional®. His company provides custom SAS programming, application design and development, consulting services, and hands-on SAS training to clients around the world. Kirk is the author of four books including PROC SQL: Beyond the Basics Using SAS by SAS Institute, and more than two hundred peer-reviewed articles and papers that have appeared in professional journals and SAS User Group proceedings. Kirk can be reached at:

Kirk Paul Lafler  
Software Intelligence Corporation  
World Headquarters  
P.O. Box 1390  
Spring Valley, California 91979-1390  
E-mail: [KirkLafler@cs.com](mailto:KirkLafler@cs.com)



SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.