

Title:

SQL Server Stored Procedures

Word Count:

562

Summary:

A database index is similar to an index in a book - it is comprised of a lookup value, and a number identifier that corresponds to the row number in a table. In SQL Server, there are two kinds of indexes - clustered and non-clustered.

Keywords:

database design nj, It Consulting nj, custom programming

Article Body:

"SQL Server Stored Procedures

Stored Procedures are SQL statements saved in a database as a n object. They can be called interactively through the Query Analyzer, and by other stored procedures. They can be defined with parameters to make them more flexible, and can return result sets and status codes.

Their advantages include:

The ability to restrict access to the underlying tables;

Increased security in a networked environment;

The vehicle to modularize programming on the database end of an application by allowing re-usable code with callable subroutines;

Faster execution because they are pre-compiled;

Preservation of integrity in tables through enforced data procedures; and

Reduction of programming error and network traffic in passing in ad hoc queries.

Disadvantages (which pale by comparison) include:

Declining portability of code because these are SQL Server specific commands;

Less powerful and flexible programming language as compared to frone-end front-end interface languages;

Sub-par development environment, although with Visual Studio .NET, this facility has been enhanced.

Stored procedures can be created in the Query Analyzer either by using the raw language, templates, or cutting and pasting from the View Creation GUI. . And they can be created in the Enterprise Manager by right-clicking on the StoredProcedure section and choosing New. . The raw syntax is:

```
CREATE PROC [ EDURE ] procedure_name [ ; number ]  
[ { @ parameter data_type }  
[ VARYING ] [ = default ] [ OUTPUT ]  
] [ , ... n ]  
  
[ WITH  
{ RECOMPILE | ENCRYPTION | RECOMPILE , ENCRYPTION } ]  
  
[ FOR REPLICATION ]  
  
AS sql_statement [ ... n ]
```

Stored Procedures (Sprocs) are executed either by invoking it by name, and or supplying the required parameters (if any). If the procedure is not the first command in a batch, you must precede the procedure name with 'EXEC'. The reason for this requirement is that in batch mode, SQL Server looks for statements by finding keywords.

A stored procedure at creation time may reference a table which does not exist, but not another stored procedure refer a table, which does not exist. However, it cannot refer a stored procedure that does not exist at all. At August 16, 2005 will be issued. You can get a list of sprocs which reference a specific table or objects referenced by a specific sproc by issuing the command

```
sp_depends (table name|procedure name)
```

And you can also use the Enterprise Manager by right-clicking the object and choosing Display Dependencies in the All Tasks Menu.

Sprocs can be viewed from the Enterprise Manager by right-click, Properties, by querying the system table, syscomments, or by using the system stored Procedure, sp_helptext followed by the name of the stored procedure you wish to view. The best way to edit the stored procedure is through the Enterprise Manager, but they can be edited programmatically using the ALTER PROCEDURE command:

```
ALTER PROC [ EDURE ] procedure_name [ ; number ]  
[ { @ parameter data_type }  
[ VARYING ] [ = default ] [ OUTPUT ]  
] [ , ... n ]  
[ WITH  
{ RECOMPILE | ENCRYPTION  
| RECOMPILE , ENCRYPTION
```

```
}  
]  
[ FOR REPLICATION ]  
AS  
sql_statement [ ... n ]
```

An efficient way to create or edit a stored procedure is to use the Query Analyzer, which can create a script for dropping an existing sproc, and creating a new one. The new one can be created using the View editor to click and drag to create a SELECT statement. Then that SQL statement can be cut and pasted into the Query Analyzer, where it can then be tested and debugged.

Stored procedures can be debugged in the Query Analyzer by opening the object browser and right clicking on a stored procedure and choosing Debug.

"