Tech Info Library

DAM: Handling a Query That Returns Multiple Tables

Article Created: 4 January 1993

TOPIC --

I'm thinking of implementing a DAM (Data Access Manager) extension to manage a custom communication API. It appears that DAM supports the notion of tabular query results (for example, GetQueryResults returns a structure that describes a "table" -- the number of columns/rows returned, the datatype of each column is returned).

Can DAM handle a query that returns multiple tables? It seems that this would involve multiple GetQueryResults calls. Do applications that work with Query Documents fetch all of the tables? How would my extension's DBGetItem function indicate the end of a table?

I remember hearing talk about Sybase and EDAsql using DAM. Both of these architectures model the concept of a single query returning multiple tables.

DISCUSSION ---

DAM is constrained by the limitations of SQL (Structured Query Language). SQL views all data in "table" format. That is, it sees everything as rows of columns that contain the actual data.

Data is kept in relational databases (the kind of databases that are accessed by SQL) in "Tables". These tables are defined as having specific, named columns. Each column contains data of a specific data type. These "real" tables are created by specific Data Definition Language (DDL) statements and are then filled with data via either batch jobs, or on-line transactions.

When you do an SQL query (regardless of how you do that query, DAM or the database's own query tool), you are requesting the DBMS to extract data from one or more columns, from one or more tables. The result of that query will always be a single "logical table" containing none, one, or more rows of the requested columns.

There will never be more than one "logical table" returned with one query command (that is, Select statement). If you wish to have multiple "logical tables" of data returned, you must make multiple queries, retrieving and storing the returned data after EACH query.

An application should be written such that it sends a query, then retrieves the data from the results "logical table", and finally checks for end-of-data. (Chapter 8 of Inside Macintosh, Volume VI goes into detail about using DAM. The result code of the DBGetItem API call will tell you when you have reached the end of the results "logical table".)

In summary, data can be returned from relational databases in many ways and multiple times, but only one set (or "logical table") of data can ever be returned at one time from any query.

Keywords: <None>

===

19960215 11:05:19.00

Tech Info Library Article Number: 11196