Prepared Statements / Parameterized Statements

Article Number: 77 | Rating: 3/5 from 1 votes | Last Updated: Tue, Jul 29, 2014 at 11:17 PM

Prepared Statements / Parameterized Statements

In database management systems, a prepared statement or parameterized statement is a feature used to execute the same or similar database statements repeatedly with high efficiency. Typically used with SQL statements such as queries or updates, the prepared statement takes the form of a template into which certain constant values are substituted during each execution.

The typical workflow of using a prepared statement is as follows:

1. **Prepare:**The statement template is created by the application and sent to the database management system (DBMS). Certain values are left unspecified, called parameters, placeholders or bind variables (labelled "?" below):
   - INSERT INTO PRODUCT (name, price) VALUES (?, ?)
2. The DBMS parses, compiles, and performs query optimization on the statement template, and stores the result without executing it.
3. **Execute:** At a later time, the application supplies (or binds) values for the parameters, and the DBMS executes the statement (possibly returning a result). The application may execute the statement as many times as it wants with different values. In this example, it might supply 'Bread' for the first parameter and '1.00' for the second parameter.

As compared to executing SQL statements directly, prepared statements offer two main advantages:

1. The overhead of compiling and optimizing the statement is incurred only once, although the statement is executed multiple times. Not all optimization can be performed at the time the prepared statement is compiled, for two reasons: the best plan may depend on the specific values of the parameters, and the best plan may change as tables and indexes change over time.
2. Prepared statements are resilient against SQL injection, because parameter values, which are transmitted later using a different protocol, need not be correctly escaped. If the original statement template is not derived from external input, SQL injection cannot occur.

On the other hand, if a query is executed only once, server-side prepared statements can be slower because of the additional round-trip to the server. Implementation limitations may also lead to performance penalties: some versions of MySQL did not cache results of prepared queries, and some DBMSs such as PostgreSQL do not perform additional query optimization during execution.

A stored procedure, which is also precompiled and stored on the server for later execution, has similar advantages. Unlike a stored procedure, a prepared statement is not normally written in a procedural language and cannot use or modify variables or use control flow structures, relying instead on the declarative database query language. Due to their simplicity and client-side emulation, prepared statements are more portable across vendors.

Software Support

Prepared statements are widely supported by major DBMSs, including MySQL, Oracle, DB2, Microsoft SQL Server and PostgreSQL. Prepared statements are normally executed through a non-SQL binary protocol, for efficiency and protection from SQL injection, but with some DBMSs such as MySQL are also available using a SQL syntax for debugging purposes.

A number of programming languages support prepared statements in their standard libraries and will emulate them on the client side even if the underlying DBMS does not support them, including Java's JDBC, Perl's DBI, PHP's PDO and Python's DB-API. Client-side emulation can be faster for queries which are executed only once, by reducing the number of round trips to the server, but is usually slower for queries executed many times. It resists SQL injection attacks equally effectively.

Posted by: Bhavesh Patel - Tue, Jul 29, 2014 at 10:10 PM This article has been viewed 232 times.
Filed Under: Others / Misc

Article Rating (1 Votes)

★★★☆☆

You've Already Voted.

Bookmark Article (CTRL-D)

Attachments

There are no attachments for this article.

Related Articles

"Minute Of Meeting" - Guidelines
Viewed 557 times since Fri, Dec 19, 2014

Stand Up Meeting
Viewed 638 times since Tue, Feb 11, 2014

Email Writing - Guidelines
Viewed 566 times since Mon, Sep 16, 2013

GPS
Viewed 270 times since Thu, Aug 29, 2013

Difference Between Layers & Tiers
Viewed 312 times since Tue, Nov 12, 2013