

Other languages: [繁體中文](#)

Database - Automatic Batch Mode

The automatic batch mode is only valid for the client library of postgresql 14+ version, and will be ignored in other cases. Before talking about automatic batch processing, let's understand the pipeline mode first.

pipeline mode

Since postgresql 14, its client library provides a pipelining mode interface. In pipelining mode, new sql requests can be sent directly to the server without waiting for the result of the previous request to return (this is consistent with the concept of HTTP pipelining), For details, please refer to [Pipeline mode](#). This mode is very helpful for performance, allowing fewer database connections to support larger concurrent requests. drogon began to support this mode after version 1.7.6, drogon will automatically check whether libpq supports pipeline mode, if so, all requests sent through drogon's DbClient are in pipeline mode.

automatic batch mode

By default, for non-transactional clients, drogon creates synchronization points for each sql request, that is, each individual sql statement is an implicit transaction, which ensures that sql requests are independent of each other, which makes pipeline mode and non-pipeline mode Modes appear to the user to be completely equivalent.

However, creating a synchronization point for each sql statement also has a certain performance overhead, so drogon provides an automatic batch mode in which instead of creating a synchronization point after each sql statement, a synchronization point is created after several sql statements. the rules for creating a synchronization point in the same connection are as follows:

- A synchronization point must be created after the last sql in an EventLoop loop;
- Create a synchronization point after a sql that writes to the database;
- Create a synchronization point after a big sql;
- Create a synchronization point when the number of consecutive SQL statements after the last synchronization point reaches the upper limit;

Note that SQL statements between two synchronization points in the same link belong to the same implicit transaction. Drogon does not provide an explicit interface for opening and closing synchronization points, so these SQLs may be logically unrelated to each other, but due to they are in the same transaction, they will affect each other, therefore, this mode is not completely safe mode, it has the following problems:

- A failed sql will cause its previous sql statement to be rolled back after the last synchronization point, but the user will not receive any notification, because no explicit transaction is used;
- A failed sql will cause all subsequent sql statements before the next synchronization point to return failure;
- The judgment of writing database is based on simple keyword matching (insert, update, etc.), which does not cover all cases, such as the case of calling stored procedures through select statements, so although drogon strives to reduce the negative effects of automatic batch mode, it is not completely safe;

Therefore, automatic batch mode is helpful to improve performance, but it is not safe. It is up to the user to decide under what circumstances to use it. For example, to execute pure read-only sql statements via the DbClient in automatic batch mode.

Note Even read-only sql can sometimes cause transaction failure (such as select timeout), so its subsequent sql will also be affected by it and fail (this may not be acceptable to users, because they may not be related to each other in application logic), so, strictly speaking, the application scenarios of automatic batch mode should be limited to read-only and non-critical data queries. It is recommended that users create a separate automatic batch DbClient for such sql statements. Of course, it is easy to realize that transaction objects generated by DbClient in automatic batch mode are safe to use.

Enable automatic batch mode

When using the newPgClient interface to create a client, set the third parameter to true to enable automatic batch mode; When using a configuration file to create a client, set the auto_batch option to true to enable automatic batch mode for the client;

Next: [Request References](#)
