

Item 183

git_comments:

git_commits:

1. **summary:** GROOVY-3691: sql.withTransaction needs to save/restore "autoCommit"
message: GROOVY-3691: sql.withTransaction needs to save/restore "autoCommit" git-svn-id:
<http://svn.codehaus.org/groovy/trunk/groovy/groovy-core@17624> a5544e8c-8a19-0410-ba12-f9af4593a198

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** sql.withTransaction needs to save/restore "autoCommit "
description: John Bito: I'd probably add the commit mode to the Sql properties, since I think it needs to be set on the connection if it's acquired from the DataSource On Tue, Aug 18, 2009 at 22:54, Fred Janon <fjanon@gmail.com> wrote: Should withTransaction preserve the state of autoCommit instead of forcing it to true before exiting?

jira_issues_comments:

1. **body:** What I was hoping is that Fred would provide a use-case. I was wondering if someone would want control over the transaction mode of the connection passed to a closure when the Sql instance is configured with a DataSource. If that were so, maybe Sql should support `sql = new Sql(DataSourceFactory.createInstance()) sql.autoCommit = false sql.cacheStatements(someClosure)` That might make it simpler to have a closure that works in withTransaction as well as cacheStatements. On the other hand, the current responsibilities are more focused on encapsulating Statement, so there's an argument against having the Sql instance maintain an autoCommit property. That being the case, it seems a real improvement for withTransaction to leave the connection in the state it started, so I think the patch is the right way to go.
label: code-design
2. If you apply that change to withTransaction(), should you also apply it to withBatch()? Seems they both have the same handling of autoCommit. John Hurst
3. patch added to withBatch and withTransaction
4. Trivial patch for saving autoCommit value - not applying until I see what other changes John was thinking of which might make it irrelevant.