

# Inline SQL queries or Stored Procedures: Yet-another-(healthy) Debate

Published 12/22/2008 by [Vivek Thakur](#)

Which is better: SQL queries in code or SPs? There is ofcourse no general answer but there are scenarios when one approach can be better than the other, or when a mixed approach can be employed.

Let's say we have an app which does standard CRUD operations and has no data intensive methods. In this case I would recommend using SQL queries because:

1. The code would be more maintainable, each developer can write/edit his/her own module related CRUD methods independently.
2. No source control related issues. If SPs are used, then the DB script would need to be locked and updated for each build. Also, script related errors can also be avoided (which can prevent developers from running successfully executing their "clean" code against the DB).
3. There can be network related issues with developers needing to access a central DB for their development work and updating the DDL scripts. Also, it can be a bit difficult to *work from home* in such scenarios.
4. Database independence is lost in some sense. Inline queries are more or less standard across all DBs, but SPs vary a lot syntactically.

So using SPs blindly thinking that they always offer performance benefits is not wise.

I personally advocate a mixed approach: using inline SQL for CRUD and SPs for data intensive batch operations. So for a DB intensive app, I might use SQL queries for CRUD operations and SPs for data crunching or long processing tasks. This will have the best of both worlds.

Note that the SQL queries I am talking about are not huge concatenated string version of an SP (which could be huge maintenance problem!).