

Re: What is your database application development environment?

Source:

<http://coding.derkeiler.com/Archive/Delphi/borland.public.delphi.non-technical/2006-03/msg03490.html>

- *From:* "Paul Nichols (TeamB)" <paul@xxxxxxxx>
 - *Date:* Sun, 19 Mar 2006 23:33:35 -0500
-

Kevin Frevert wrote:

We recently had a 'heated' discussion on our current database application development environment. It is currently..
Production – Test – Sandbox

Most call it Dev, Test–QA, Production, but whatever you call it, this is the norm.

We develop in the sandbox, test it on a recent restored database, and if all goes well, the code goes to production. This environment has worked extremely well since MSSQL 6.5.

Most of the time, Dev databases are much smaller. Many Test or QA databases contain a recent snapshot of Production.

Last week a developer accidentally copied a stored proc from the sandbox to production (skipping the test db) without checking to make sure the production sp was the same as what was in the sandbox (before he made the requested changes). Without getting into the politics, the <bleep> hit the fan and the 'powers-that-be' decided to 'fix' the problem by weekly production restores/overwrites of both the test and sandbox databases.

Well that too, would be the norm.

We already had weekly restores of the test database, but we all felt the sandbox was exactly that, an area that let us 'play' and try out new things. Our sandbox never was, and never intended to be, a mirror of the

Re: What is your database application development environment?

production environment.

Usually that is the case. Something like SPs however, should be tested before they are promoted to Test or QA

The 'powers-that-be' said we would have no more of this 'cowboy development' around here and 'everyone else' does it his way (whoever 'everyone' is and whatever 'it' is). Most of the other developers with laptops also have a (smaller) copy of the database to do development in which we were told we also had to do weekly restores. I'll throw in the 'power-that-be' has zero database administration experience.

That sounds like overkill, but be thankful at least they understand that there should be a promotion process and some manner of Change Control. Not bad, but pushing Prod to Laptops definitely is a little extreme.

What was wrong with the way we were doing things? Is this how other shops do database app development? What is your database application development environment?

Ideally, no developer should have access to Prod. That is a no-no, even though far too many companies have no formal Change Control processes in place.

I shudder to think what a real Enterprise would do, if developers were free to promote to production and kill half of the systems by an inadvertent change. Not a good idea for Developers to promote their own code at all. I would absolutely agree with the "powers that be" on this one.

Any links/articles with advice is also appreciated.

No links, but having been in several large fortune 1000 and 500 companies, as R&D Team Lead and Architect, I can tell you that 90% would not let a developer near production. Development should end at Test. Promotion should be done by a Change control and security team, only after a thorough QA and UAT process has been performed. These promotion/QA teams may require a code review and code walkthrough at the QA level as well.

Although developers working in small companies may think this odd, when you have hundreds of programs that the company MUST depend upon from CMS to BPM to Financials to HR, the last thing you want is for rogue code to bring down a system.

Re: What is your database application development environment?

Re: What is your database application development environment?

I would not look at this at losing control or power, but rather as protecting your posterior. I would feel much better if during the testing process, some bug was found rather than discovering that some bug in the production environment affected some HR process and 1000 people did not get paid, or were paid too much/not enough.

You can bet your buns would get burned and overtime is something you may not want to complain about. It is far better to let someone else take responsibility. Bugs are going to happen and occur and no test plan is perfect, but the practice of eliminating as many bugs as possible, before they can affect entire systems, is worth two pounds of cure.

You may want to express your concern on trying to install huge databases on development environments, but to make sure that no code is promoted before its time, is a good idea.

Sounds like you have a pretty good company to work for there, overall.