

Smart Access

Solutions for Microsoft® Access® Developers

Performance

Peter Vogel



With a combination of abstract rules and real-world stories, editor Peter Vogel provides you with the rules that you need to create Access applications that run as fast as possible. And he throws in some advice on advancing your career.

THE slowest thing that you can do is read from or write to your hard disk. This is an ironic fact of life for Access developers (or any database developer): Your goal is to access and update your database, and that very activity is what slows down your application more than anything else. Life's like that. This is reality; adjust!

Since data access is what slows you down, you shouldn't worry about all those programming tricks that are supposed to speed up your application. Writing faster code is like speeding in the city: You just get to the next red light as fast as possible. What you want to do is reduce the time spent at the red light—the time spent waiting for data access. So that's what this article is all about: reducing the time spent at your red lights.

I never believe in anything that I haven't seen for myself. So for many of these rules, I've provided a real-world story from my life that demonstrates the rule in action. If you're a longtime reader of *Smart Access*, you'll recognize some (or all) of these stories. There's a reason for this: I'd like to think that having made these mistakes once, I've never made them again. At any rate, that's my story and I'm sticking to it. In about half of these stories, applying these techniques made me a hero (in the rest, not applying them made me the goat).

First: Normalize your operational data

If you don't know the first three laws of normalization, learn them. As you normalize your data, you'll find that you generate extra tables. Many developers back off from normalization in order, they think, to improve their performance. They're concerned that having to join together multiple tables will slow them down. Here's a news flash: The designers of your RDBMS have made your RDBMS very good at what it does. And one of the things that your RDBMS is very, very good at is joining tables.

Continues on page 4

June 2005

Volume 13, Number 6

- 1 Performance**
Peter Vogel
- 2 Editorial: Architecture**
Peter Vogel
- 8 Let Your Users Sort it Out**
Christopher R. Weber
- 13 Two Functions You Don't Use Enough**
Wayne Wallace
- 15 Access Answers: I've Got Plenty of Nothing, and Did Someone Say Holiday?**
Doug Steele
- 20 June 2005 Downloads**

 Applies to Access 95	 Applies to Access 97	 Applies to Access 2000	 Applies to Access 2002
 Applies to Access 2003			
Accompanying files available online at www.pinnaclepublishing.com			