

Re: NGen introduces instability

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework/2006-12/msg00145.html>

- *From:* "Damien" <Damien.The.Unbeliever@xxxxxxxxxxxxx>
 - *Date:* 6 Dec 2006 03:52:52 -0800
-

Bill wrote:

Damien wrote:

Bill wrote:

We have a section of code that executes cleanly until that code is NGen'ed. After a native image has been created, that same section of code produces a .NET Runtime Fatal Execution Engine Failure and the application closes without warning.

Here is the error for the Event Viewer: .NET Runtime version 2.0.50727.42 – Fatal Execution Engine Error (7A05E2B3) (80131506)

We've re-created this issue on XP, Win2000, and Win2003. We've executed the native image in the debugger; however the debugger does not produce anything insightful.

Has anyone else encountered this?

Thank you for your time,
Bill O'Neil

I'm guessing it's the variable called "blarg", on line 32.

You say you've reproduced this on XP, 2000 and 2003. Can you produce a simple sample piece of code that exhibits this problem, or can you only reproduce it with your full solution.

It's a *lot* easier to diagnose problems when we have code in front of

Re: NGen introduces instability

us.

Damien

The code should not matter, right? The IL executes just fine, but as

But obviously the code does matter. Otherwise, your statement is the same as saying that every application/library that is NGen'ed crashes. And if that was the case, I'm sure more people would have noticed that by now.

soon as it is NGen'ed, the same code causes .NET to fail. It's my understanding that NGen should not alter the behavior of the program. We are not using any unsafe code blocks or unmanaged code in the problematic method.

To answer your question, I don't know what causes it to crash. I believe it occurs after a call to ShowDialog(), but I can not confirm. I've NGen'ed the debug versions of the program in an attempt to isolate the issue, but the debug version does not crash, only the NGen'ed release version.

Thanks,
Bill

I think you'll have to find the problem the hard way – commenting out large chunks of code til the problem goes away, then adding code back until you can reliably reproduce the problem. You'd then probably be able to provide some repro code so that others can throw their eyeballs at the problem.

Damien

.