
Subject: Re: IDL_IDLBridge: No gain?

Posted by [dg86](#) on Mon, 12 Aug 2013 13:59:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Ronn,

Thanks very much for fielding my question. I added a callback routine that prints the bridge's ID the elapsed time in the bridge's execution. Both the ID and the starting time are passed to each bridge as USERDATA upon creation. I removed the wait statements altogether. Instead, I stare at the command line, which reappears immediately, until the bridges report back.

The bridges report back all at once, and in apparently random order, suggesting that they are indeed running simultaneously and not sequentially. They report nearly the same running time. Unfortunately, that time seems to scale linearly with the number of running bridges.

Two bridges each report needing 11 seconds to run my expensive routine (Imfeature).
Four bridges each report 18.5 seconds.
Seven bridges report 39 seconds.
Eight bridges report 46 seconds.

Why does the execution time for one bridge depend on the number of bridges running simultaneously?

All the best,

David

On Monday, August 12, 2013 7:08:23 AM UTC-4, [rlk...@gmail.com](#) wrote:

> Hi David,

>

>

>

> What's killing you is the wait in the while loop. What you need to do is specify a callback for your routines to call after they are done.

>

>

>

> In the callbacks you can set some flag to indicate when they are called and then in your main program just check the status of the flags every second or so.

>

>

>

> -Ronn Kling
