Subject: Re: Suggestions on the use of lockfiles?
Posted by Jonathan Greenberg on Mon, 10 Feb 2003 06:16:24 GMT
View Forum Message <> Reply to Message

Robert:

Thanks! That sounds fine, and is almost what I had come up with (incidentally, I finally found out the term for this is a "race condition", since I'm not a programmer it took some searching so I could find an official term for what was happening) -- here's the only possible issue that I was concerned about:

If two computers openw,/APPEND and printf the same file, let's say their ID (computer #1 = 100 and #2 = 200), do we only have 2 possibilities in IDL:

Lockfile contains:

100
200

Or

200
100

Or is it possible to have both writing at the same time, giving some corrupted ID like:
210000 (200 with "100" inside of it)?

If the corrupted id is possible, it makes the situation a bit more confusing, since the lockfile needs to be deleted before it can move on, an this brings up some other issues (who gets to delete the file?)

--j

On 2/9/03 12:08 PM, in article Gyy1a.7580$yn1.672904@twister.austin.rr.com, "Robert Moss" <rmoss4@houston.rr.com> wrote:

> There is probably 100 different ways to approach this, but I prefer to
> start simple. Try this: when the lockfile is created, write some unique
> identifier into the lock file (e.g. the host name of the machine doing
> the locking). Just before you actually start writing the DB, confirm
> that the first line in the lockfile corresponds to the machine that is
> getting ready to alter the DB. If it does not, it can go back to the
> wait loop.
>
> Robert Moss, PhD
> rmoss4@houston.rr.com

>
> Jonathan Greenberg wrote:
>> I am having some troubles using lockfiles to do simple database read/writes
>> without getting a database corruption (I'm trying to hack my way around
>> buying Dataminer):
>>
>> I have two or more computers running this code:
>>
>> dblockname is the name of the lockfile for the database
>>
>> ***
>>
>> while [program is still running] do begin
>> ; Waits for the lockfile to not exist
>> while (file_test(dblockname) eq 1) do begin
>> wait,1
>> endwhile
>> ; Creates a lockfile
>> openw,dblocklun,dblockname,/GET_LUN
>> printf,dblocklun,''
>> free_lun,dblocklun
>>
>> [gets the database, reads and writes to it]
>> ; Deletes the lockfile, and returns to the top of the loop
>> file_delete,dblockname
>>
>> endwhile
>>
>> ***
>>
>> The problem I'm having is that from time to time (depending on the speed of
>> the database I/O), I'm getting errors where, and this is a guess, one
>> computer manages to check for a nonexistant lockfile at the same time as
>> another, and so both computers break out of the loop at the same time and
>> start fooling around with the database at the same time. At the end,
>> whichever computer deletes the lockfile first can reenter the loop, but the
>> 2nd computer gets an error since the file was already deleted (and the
>> database is likely to be corrupted at this point).
>>
>> Ideas? The astronomy lockfile procedures appear to be unix only, and I'm
>> trying to design this for any platform (Windows, Mac, UNIX). Apparently I
>> can openw the same file from two computers without error (I wish that I
>> could override this fact). Help!
>>
>> --j
>>
>