<http://mail-archives.apache.org/mod_mbox/lucene-dev/200110.mbox/%3C4BC270C6AB8AD411AD0B00B0D0493DF0EE7C30@mail.grandcentral.com%3E>

From DCutting@grandcentral.com Fri Oct 5 18:54:51 2001

Return-Path: <DCutting@grandcentral.com>

Mailing-List: contact lucene-dev-help@jakarta.apache.org; run by ezmlm

Delivered-To: mailing list lucene-dev@jakarta.apache.org

Received: (qmail 44335 invoked from network); 5 Oct 2001 18:54:51 -0000

Received: from unknown (HELO mta.12.com) (65.198.8.41)

by daedalus.apache.org with SMTP; 5 Oct 2001 18:54:51 -0000

Received: (qmail 14012 invoked from network); 5 Oct 2001 18:52:15 -0000

Received: from unknown (HELO riker.grandcentral.com) (10.102.15.55)

by mta.12.com with SMTP; 5 Oct 2001 18:52:15 -0000

Received: by mail.grandcentral.com with Internet Mail Service (5.5.2653.19)

id <42Y1HDH3>; Fri, 5 Oct 2001 11:44:35 -0700

Message-ID: <4BC270C6AB8AD411AD0B00B0D0493DF0EE7C30@mail.grandcentral.com>

From: Doug Cutting <DCutting@grandcentral.com>

To: "'lucene-dev@jakarta.apache.org'" <lucene-dev@jakarta.apache.org>

Subject: FW: Lucene 1.2 and directory write permissions?

Date: Fri, 5 Oct 2001 11:44:34 -0700

MIME-Version: 1.0

X-Mailer: Internet Mail Service (5.5.2653.19)

Content-Type: text/plain;

charset="iso-8859-1"

Content-Transfer-Encoding: quoted-printable

X-Spam-Rating: daedalus.apache.org 1.6.2 0/1000/N

Here's one vote for putting locks in a separate directory. Anyone =

dislike

that?

Doug

-----Original Message-----

From: Snyder, David [mailto:dsnyder@netgenics.com]

Sent: Friday, October 05, 2001 11:23 AM

To: Doug Cutting

Subject: RE: Lucene 1.2 and directory write permissions?

The lock file synchronization is very handy for us as we do updates in =

a

separate process from searches... I was very pleased to see this in =

there!

I think splitting out the locks into a separate directory would solve =

our

problem... we have a reference set of data that we use for testing and =

our

sysadmin wants to make sure it doesn't accidentally get overwritten. =

It's

still possible (using links and other permission magic) but becomes =

more of

a maintenance headache. I definitely vote for the locks subdirectory =

idea.

It can be created when the indexes are initially loaded, we can go =

change

the permissions or maybe make it a link to a tmp directory or =

something, and

then our regular index files can be safe. Do you think this is =

something

very difficult to do? (I have yet to build Lucene myself, but I would =

love

to contribute... we are actually working on some XML based loaders that =

may

be of general interest)

Lucene has been great for us, but the way... we are indexing genetic =

data

(not the sequences themselves, but all the annotations and description =

stuff

that scientists tack on) and Lucene has been excellent... our indexes =

(we

use many with the multisearcher) are about 13 gigs now and Lucene has =

hardly

broken a sweat.

Thanks for your help,

Dave

David Snyder

Se=F1or Software Engineer

NetGenics, Inc.

1717 E. 9th St., #1700

Cleveland, OH 44114

(216) 861-4007

-----Original Message-----

From: Doug Cutting [mailto:DCutting@grandcentral.com]

Sent: Friday, October 05, 2001 12:24 PM

To: 'Snyder, David'

Subject: RE: Lucene 1.2 and directory write permissions?

> From: Snyder, David [mailto:dsnyder@netgenics.com]

>=20

> I've been porting our application to use the 1.2 release=20

> candidate 1 build

> and now have a problem opening searchers on our existing=20

> indexes. I get a

> Permission Denied exception... our permissions are set up to=20

> allow reading

> of the directory and contained files during a search, but not =

writing.

Hmm. That is a problem. The reader now creates a lock file while it =

is

opening the index to keep a writer process from deleting files while =

they're

being opened. When opening an index the reader must first read the =

list of

files to open, then open them. If between reading the list and opening =

a

file that file were to disappear, then the open would fail. This was =

the

longstanding race condition that is fixed by the lock files. A writing

process will now wait for the reader to open all of the files before

updating things.

Perhaps we should instead write lock files in a subdirectory of the =

index

named "locks". You could make that directory read/write, but make the

parent read-only. Alternately, we could have an flag that turns off =

the use

of lock files, for those who know that there is no other process that =

is

potentially simultaneously updating the index. Which approach would =

folks

prefer?

Doug