BIEN 3 - Task #887

fix disk space leak that fills the disk and crashes the import

04/11/2014 04:02 PM - Aaron Marcuse-Kubitza

Status:	Rejected	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Aaron Marcuse-Kubitza	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:			
Activity type:			

Description

the bug that triggers this Postgres bug (#902) has now been fixed, so no need to fix this

issue

- in the last full-database import, this caused disk space errors in 29 of 41 datasources:

```
ssh -t vegbiendev.nceas.ucsb.edu exec sudo su - aaronmk
export version=r13016
grep --files-with-matches -F "No space left on device" inputs/{.,}*/*/logs/$version.log.sql
# and uniqify by datasource
```

- in a test run, importing even moderate-sized datasources (eg. NY, SALVIAS) causes all the available disk space to be used up, and crashes the import:

```
ssh -t vegbiendev.nceas.ucsb.edu exec sudo su - aaronmk
export version=test_import
grep --files-with-matches -F "No space left on device" inputs/{.,}*/*/logs/$version.log.sql
# the list includes all the datasources in the test run: NY, SALVIAS
```

- because the test run crashed as well, the problem is most likely a bug in Postgres itself. this unfortunately means that we **can't run the import until we find the system bug** that is causing the problem.
    - based on the testing done for issue #902 (join columns), this is almost certainly *not* an Ubuntu bug, because it also occurs on Mac OS X
- the problem still occurs after upgrading to Ubuntu 14.04, both with the compatibility version of Postgres we had been using before the upgrade as well as with the official Ubuntu version supplied with Ubuntu 14.04

broken queries

- the disk fills up during queries that normally run quickly but now seem to run indefinitely until the disk space is used up (note the query_start times above each query, compared to when the snapshot was taken):

```
pg_stat_activity snapshot taken: "Apr  2 11:42:26 2014"

running for 5.5 h:
"query_start"
"2014-04-02 06:20:01.952174-07"
/*NVS.StemObservation*/    CREATE TEMP TABLE "stemobservation_pkeys" AS
SELECT
"in#11"."row_num"
, "stemobservation"."stemobservation_id" AS "out.stemobservation_id"
FROM "in#11"
JOIN "stemobservation" ON COALESCE("stemobservation"."height_m", CAST('NaN' AS double
precision)) = COALESCE("in#11"."NVS.StemObservation.Height::double", CAST('NaN' AS double
precision))
/* EXPLAIN:
Merge Join  (cost=1045217.21..95909185.15 rows=6323041431 width=8)
  Merge Cond: ((COALESCE(stemobservation.height_m, 'NaN'::double precision)) = (COALESCE("in#1
1"."NVS.StemObservation.Height::double", 'NaN'::double precision)))
  ->  Sort  (cost=30039.56..30492.76 rows=181279 width=12)
        Sort Key: (COALESCE(stemobservation.height_m, 'NaN'::double precision))
        ->  Seq Scan on stemobservation  (cost=0.00..14206.79 rows=181279 width=12)
  ->  Materialize  (cost=1015177.64..1050057.81 rows=6976033 width=12)
        ->  Sort  (cost=1015177.64..1032617.73 rows=6976033 width=12)
              Sort Key: (COALESCE("in#11"."NVS.StemObs
*/

running for 21 h:
"query_start"
"2014-04-01 14:56:05.573418-07"
/*CVS.taxon_observation.***/    CREATE TEMP TABLE "location_pkeys" AS
SELECT
"in#5"."row_num"
, "location"."location_id" AS "out.location_id"
FROM "in#5"
JOIN "location" ON ("location"."accesslevel" = "in#5".
"CVS.taxon_observation.**.accessRights::accesslevel" OR ("location"."accesslevel" IS NULL AND
"in#5"."CVS.taxon_observation.**.accessRights::accesslevel" IS NULL))
/* EXPLAIN:
Nested Loop  (cost=0.00..83049883712.45 rows=27683225000 width=8)
  Join Filter: ((location.accesslevel = "in#5"."CVS.taxon_observation.**.accessRights::accessl
evel") OR ((location.accesslevel IS NULL) AND ("in#5"."CVS.taxon_observation.**.accessRights::
accesslevel" IS NULL)))
  ->  Seq Scan on location  (cost=0.00..175778.45 rows=5536645 width=8)
  ->  Materialize  (cost=0.00..35434.00 rows=1000000 width=8)
        ->  Seq Scan on "in#5"  (cost=0.00..30434.00 rows=1000000 width=8)
*/

config: Ubuntu 14.04 with official Ubuntu version of Postgres

import command:
declare -ax inputs=(inputs/{ARIZ,SALVIAS}/)
export version=test_import
. bin/import_all

pg_stat_activity snapshot taken: "2014-04-23 19:20:59.165896-07"

running for 2 h:
"query_start"
"2014-04-23 17:23:43.400618-07"
/*ARIZ.omoccurrences*/    CREATE TEMP TABLE "location_pkeys" AS
SELECT
"in#6"."occurrenceID"
, "location"."location_id" AS "out.location_id"
FROM "in#6"
JOIN "location" ON COALESCE("location"."elevation_m", CAST('NaN' AS double precision)) =
COALESCE("in#6"."_alt(1=verbatimElevation, 2=_avg(1=minimumElevat.result::double", CAST('NaN'
AS double precision))
/* EXPLAIN:
Merge Join  (cost=52039.20..3508151.98 rows=230343442 width=8)
  Merge Cond: ((COALESCE(location.elevation_m, 'NaN'::double precision)) = (COALESCE("in#6"."_
alt(1=verbatimElevation, 2=_avg(1=minimumElevat.result::double", 'NaN'::double precision)))
  ->  Sort  (cost=23780.86..24261.44 rows=192230 width=12)
        Sort Key: (COALESCE(location.elevation_m, 'NaN'::double precision))
        ->  Seq Scan on location  (cost=0.00..6910.30 rows=192230 width=12)
  ->  Sort  (cost=28258.34..28857.47 rows=239654 width=12)
        Sort Key: (COALESCE("in#6"."_alt(1=verbatimElevation, 2=_avg(1=minimumElevat.result::d
ouble", 'NaN'::double precision))
        -
*/

uses >350 GB for sort temp storage (disk space increases until out of space):

# before stopping import (160 GB free):
df
```

```
Filesystem                    Size  Used Avail Use% Mounted on
/dev/mapper/vegbiendev-root   2.0T  1.8T  160G  92% /
none                          4.0K     0  4.0K   0% /sys/fs/cgroup
udev                           16G  4.0K   16G   1% /dev
tmpfs                         3.2G  400K  3.2G   1% /run
none                          5.0M     0  5.0M   0% /run/lock
none                           16G  152K   16G   1% /run/shm
none                          100M     0  100M   0% /run/user
/dev/vda1                     485M   66M  394M  15% /boot

# after stopping import (516 GB free):
df
Filesystem                    Size  Used Avail Use% Mounted on
/dev/mapper/vegbiendev-root   2.0T  1.4T  516G  73% /
none                          4.0K     0  4.0K   0% /sys/fs/cgroup
udev                           16G  4.0K   16G   1% /dev
tmpfs                         3.2G  400K  3.2G   1% /run
none                          5.0M     0  5.0M   0% /run/lock
none                           16G  152K   16G   1% /run/shm
none                          100M     0  100M   0% /run/user
/dev/vda1                     485M   66M  394M  15% /boot
```

- may be related to the same problem that caused joining on the wrong columns (#902)
- the problem is not necessarily caused by #884, because sort temp files consume disk space only in proportion to the table size, which for a small import would not be nearly enough to fill the disk

fix

see proposed steps in #905

Subtasks:

Task # 902: fix bug that causes joining on the wrong columns in the import	**Resolved**
Task # 905: narrow down the cause of the import bug (incorrect join columns and disk sp...	**Resolved**

History

#1 - 04/11/2014 04:14 PM - Aaron Marcuse-Kubitza

- Description updated

#2 - 04/11/2014 04:15 PM - Aaron Marcuse-Kubitza

- Description updated

#3 - 04/11/2014 04:43 PM - Aaron Marcuse-Kubitza

- Subject changed from fix Postgres bug that causes all the available disk space to be used up to fix disk space leak that causes all the available disk space to be used up

- Description updated

#4 - 04/17/2014 01:33 AM - Aaron Marcuse-Kubitza

- Description updated

#5 - 04/17/2014 03:01 AM - Aaron Marcuse-Kubitza

- Description updated

#6 - 04/17/2014 10:16 AM - Aaron Marcuse-Kubitza

- Subject changed from fix disk space leak that causes all the available disk space to be used up to fix disk space leak that fills the disk and crashes the import

#7 - 04/17/2014 10:31 AM - Aaron Marcuse-Kubitza

- % Done changed from 0 to 20

submitted support request to restore vegbiendev to last working configuration and install a past revision of Postgres 9.3

#8 - 04/17/2014 10:42 AM - Aaron Marcuse-Kubitza

- Description updated

#9 - 04/17/2014 10:43 AM - Aaron Marcuse-Kubitza

- Description updated

#10 - 04/17/2014 10:46 AM - Aaron Marcuse-Kubitza

- Description updated

#11 - 04/17/2014 10:47 AM - Aaron Marcuse-Kubitza

- Description updated

#12 - 04/17/2014 08:05 PM - Aaron Marcuse-Kubitza

- Description updated

#13 - 04/17/2014 08:14 PM - Aaron Marcuse-Kubitza

- % Done changed from 20 to 30

main DB backed up, (close to?) ready to roll back and/or upgrade the VM

#14 - 04/23/2014 03:45 PM - Aaron Marcuse-Kubitza

- Description updated

#15 - 04/23/2014 03:46 PM - Aaron Marcuse-Kubitza

- Description updated

#16 - 04/23/2014 03:57 PM - Aaron Marcuse-Kubitza

- Description updated

#17 - 04/23/2014 07:21 PM - Aaron Marcuse-Kubitza

- Description updated

#18 - 04/23/2014 07:22 PM - Aaron Marcuse-Kubitza

- % Done changed from 30 to 40

VM upgraded to Ubuntu 14.04 and using the official Ubuntu version of Postgres, but problem still occurs

#19 - 04/23/2014 07:23 PM - Aaron Marcuse-Kubitza

- Description updated

#20 - 04/23/2014 07:30 PM - Aaron Marcuse-Kubitza

- Description updated

#21 - 04/23/2014 07:31 PM - Aaron Marcuse-Kubitza

- Description updated

#22 - 04/24/2014 04:48 PM - Aaron Marcuse-Kubitza

- Description updated

#23 - 04/25/2014 05:16 AM - Aaron Marcuse-Kubitza

- Description updated

#24 - 04/25/2014 05:17 AM - Aaron Marcuse-Kubitza

- Description updated

#25 - 04/25/2014 05:42 AM - Aaron Marcuse-Kubitza

- Description updated

#26 - 04/25/2014 05:58 AM - Aaron Marcuse-Kubitza

- Description updated

#27 - 04/25/2014 06:00 AM - Aaron Marcuse-Kubitza

- Description updated

#28 - 05/10/2014 09:30 AM - Aaron Marcuse-Kubitza

- Description updated

#29 - 05/28/2014 02:51 PM - Aaron Marcuse-Kubitza

- Parent task set to #905

#30 - 05/28/2014 03:00 PM - Aaron Marcuse-Kubitza

- Parent task deleted (#905)

#31 - 07/15/2014 05:50 PM - Aaron Marcuse-Kubitza

- Description updated

- Status changed from New to Rejected

#32 - 09/10/2014 04:10 PM - Aaron Marcuse-Kubitza

not fixing this because we fixed the bug in our code that was triggering this Postgres bug