

QGIS Application - Bug report #21718

Inconsistent layer extent between ST_Extent and ST_Estimated_Extent

2019-04-01 08:46 AM - Andreas Neumann

Status:	Closed	
Priority:	High	
Assignee:	Jürgen Fischer	
Category:	Data Provider/PostGIS	
Affected QGIS version:	3.7(master)	Regression?: No
Operating System:	all	Easy fix?: No
Pull Request or Patch supplied:	No	Resolution:
Crashes QGIS or corrupts data:	No	Copied to github as #: 29533
Description		
<p>We have a strange situation of a WMS service where there is only one layer in the project.</p> <p>If the layer is loaded directly as the leaf layer, then not all data is displayed, if the layer is loaded as root (top level) layer, then all data is displayed.</p> <p>For comparison reasons, in the attached project and screenshot the leaf layer is displayed in gray, and the top level layer in color (yellow).</p> <p>Project is in EPSG 2056 and should also be loaded as such.</p> <p>Data source is PostgreSQL (data attached).</p>		

Associated revisions

Revision 95bd7480 - 2019-04-02 03:59 PM - Jürgen Fischer

postgres provider: use st_estimatedextent only when 'use estimated metadata' is enabled (fixes #21718)

Revision d3ec6b59 - 2019-04-05 04:11 PM - Jürgen Fischer

postgres provider: use st_estimatedextent only when 'use estimated metadata' is enabled (fixes #21718)

(cherry picked from commit 95bd7480d0f0b65359e23bee5454e066fb712cab)

Revision ff166628 - 2019-04-06 07:20 PM - Jürgen Fischer

- backports
- c77e393ea4 legend: truncate excessively long tooltips (fixes #21737)
 - 88513284d4 fix removal of temporary directories (fixes #21292)
 - 95bd7480d0 postgres provider: use st_estimatedextent only when 'use estimated metadata' is enabled (fixes #21718)
 - f1dbcc3c53 processing: fix grass' r.quantile parameter (fixes #21751)
 - c6382553cd don't relink object if just the shared objects/DLL is changed
 - 9aaaad3ab8 trim crssync include directories (followup 8851328)

History

#1 - 2019-04-01 08:52 AM - Andreas Neumann

Forgot to add the URL of the WMS:

<https://services.geo.zg.ch/ows/TestClip>

#2 - 2019-04-01 08:56 AM - Alessandro Pasotti

- Assignee set to Alessandro Pasotti

#3 - 2019-04-01 09:25 AM - Andreas Neumann

This issue might not be a server issue.

We are experiencing strange things on the Desktop with this data as well (e.g. the extent that QGIS calculates (invalid) is different from the extent that Postgis is calculating (coorect))

Will post more infos when I find out more.

#4 - 2019-04-01 09:59 AM - Andreas Neumann

More info:

```
SELECT 1 AS pk, st_estimated_extent('ameisen_1700','ameisenschutz','geom') AS geom
```

run in Postgis

gives the invalid result that also QGIS uses.

#5 - 2019-04-01 10:35 AM - Andreas Neumann

Issue is that QGIS uses the ST_Estimated_Extent() from Postgis as layer extent, which may differ from real extent given from ST_Extent()

The question is, if QGIS should be using ST_Estimated_Extent(), because it may differe substantially from the real extent given back by ST_Extent()

Note that I did not enable the "use estimated table metadata" checkbox.

#6 - 2019-04-01 10:35 AM - Marco Bernasocchi

the culprit might be here <https://github.com/qgis/QGIS/blob/master/src/providers/postgres/qgspostgresprovider.cpp#L3240>

#7 - 2019-04-01 10:42 AM - Andreas Neumann

Maybe this line should be:

```
if ( !mIsQuery && mUseEstimatedMetadata && mSqlWhereClause.isEmpty() )
```

#8 - 2019-04-01 10:46 AM - Andreas Neumann

- Assignee changed from Alessandro Pasotti to Jürgen Fischer

- Operating System changed from Linux to all

- Affected QGIS version changed from 3.6.1 to 3.7(master)

- Category changed from QGIS Server to Data Provider/PostGIS

- Subject changed from QGIS Server WMS: inconsistent layer extent when viewing leaf layer vs root layer to Inconsistent layer extent between ST_Extent and ST_Estimated_Extent

#9 - 2019-04-01 12:39 PM - Jürgen Fischer

Andreas Neumann wrote:

Maybe this line should be:

```
if ( !mIsQuery && mUseEstimatedMetadata && mSqlWhereClause.isEmpty() )
```

Depends. mUseEstimatedMetadata was meant to trade performance for accuracy - and not to avoid usage of inaccurated stats. So the original version was to use the stats even if there is a where clause (which otherwise would alter the returned extent), if mUseEstimatedMetadata is on.

Wouldn't it be better to just analyze the table to update the stats? You want reliable stats for other queries too.

#10 - 2019-04-01 01:04 PM - Andreas Neumann

Hi Jürgen,

Thanks for having a look at the issue. In my case (PostgreSQL 10, Postgis 2.4) the vacuum analyze did not help unfortunately.

However, Marco B. imported the data into his PostgreSQL 11 and did not reproduce the issue.

#11 - 2019-04-01 01:40 PM - Jürgen Fischer

Andreas Neumann wrote:

Thanks for having a look at the issue. In my case (PostgreSQL 10, Postgis 2.4) the vacuum analyze did not help unfortunately.

However, Marco B. imported the data into his PostgreSQL 11 and did not reproduce the issue.

Interesting - 9.6/2.3 and 10/2.4 have the issue - 11/2.5 is fine.

```
test=# select version(),postgis_version(),st_estimatedextent('ameisen_1700','ameisenschutz','geom'),st_extent(geom) from
ameisen_1700.ameisenschutz;
```

```
-[ RECORD 1 ]-----+-----
```

```
version      | PostgreSQL 9.6.5 on x86_64-pc-linux-gnu, compiled by gcc (Debian 7.2.0-1) 7.2.0, 64-bit
```

```
postgis_version | 2.3 USE_GEOS=1 USE_PROJ=1 USE_STATS=1
```

```
st_estimatedextent | BOX(2681941.25 1219439.125,2683874.5 1222586.5)
```

```
st_extent      | BOX(2680800.79 1219175.85,2683864.79 1222570.85)
```

```
test=# select version(),postgis_version(),st_estimatedextent('ameisen_1700','ameisenschutz','geom'),st_extent(geom) from
ameisen_1700.ameisenschutz;
```

```
-[ RECORD 1 ]-----+-----
```

```
version      | PostgreSQL 10.5 (Debian 10.5-1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 8.2.0-3) 8.2.0, 64-bit
```

```
postgis_version | 2.4 USE_GEOS=1 USE_PROJ=1 USE_STATS=1
```

st_estimatedextent | BOX(2681941.25 1219439.125,2683874.5 1222586.5)
st_extent | BOX(2680800.79 1219175.85,2683864.79 1222570.85)

```
test=# select version(),postgis_version(),st_estimatedextent('ameisen_1700','ameisenschutz','geom'),st_extent(geom) from
ameisen_1700.ameisenschutz;
-[ RECORD 1 ]-----+-----
version      | PostgreSQL 11.2 (Debian 11.2-2) on x86_64-pc-linux-gnu, compiled by gcc (Debian 8.3.0-2) 8.3.0, 64-bit
postgis_version | 2.5 USE_GEOS=1 USE_PROJ=1 USE_STATS=1
st_estimatedextent | BOX(2680800.75 1219175.75,2683865 1222570.875)
st_extent      | BOX(2680800.79 1219175.85,2683864.79 1222570.85)
```

#12 - 2019-04-01 05:04 PM - Andreas Neumann

hm - strange.

What do you suggest?

I am happy to upgrade our server one day (not immediately) - but I guess that I am not the only one with such a combo (10/2.4) ;-(

#13 - 2019-04-02 04:00 PM - Jürgen Fischer

- Status changed from Open to Closed
- % Done changed from 0 to 100

Applied in changeset commit:qgis|95bd7480d0f0b65359e23bee5454e066fb712cab.

#14 - 2019-04-02 04:04 PM - Jürgen Fischer

Andreas Neumann wrote:

| I am happy to upgrade our server one day (not immediately) - but I guess that I am not the only one with such a combo (10/2.4) ;-(

Now st_estimatedextent is only used when "use estimated metadata" is enabled, but in that case the where clause is still ignored. So it should still have the intended boost when enabled, but not be inaccurate anymore when disabled.

Files			
wms_strange_clipping_problem.png	148 KB	2019-04-01	Andreas Neumann
clipped_wms.qgs	16.4 KB	2019-04-01	Andreas Neumann
TestClip.qgs	36.8 KB	2019-04-01	Andreas Neumann
ameisennester_1700.backup	6.39 KB	2019-04-01	Andreas Neumann