# QGIS Application - Bug report #21085
## Processing time for GeoJSON 10 times slower in 3.4

2019-01-24 11:03 AM - Peter Gipper

| | | | |
|---|---|---|---|
| **Status:** | Closed | | |
| **Priority:** | High | | |
| **Assignee:** | Even Rouault | | |
| **Category:** | Processing/Core | | |
| **Affected QGIS version:** | 3.4.4 | **Regression?:** | Yes |
| **Operating System:** | Windows 7, Windows 10 | **Easy fix?:** | No |
| **Pull Request or Patch supplied:** | | **Resolution:** | up/downstream |
| **Crashes QGIS or corrupts data:** | | **Copied to github as #:** | 28903 |

**Description**

When i use the processing algorithm to create voronoi polygons from a GeoJSON file, the processing time became more than 10x longer in 3.4 compared to 3.2.

Attached is test data (osm) with 3442 point features. In QGIS 3.2 it takes 5 seconds to process. In QGIS 3.4 it takes 67 seconds.

I think this might also be observed with other processing algorithms. It is the same for geographic and projected CRS.

Would be nice to get this fixed since GeoJSON is an awesome format and became somewhat popular.

**Related issues:**

| | | |
|---|---|---|
| Duplicated by QGIS Application - Bug report # 21088: QGIS 3.4 much much slowe... | **Closed** | **2019-01-24** |

## History

**#1 - 2019-01-24 01:34 PM - Jürgen Fischer**

*- Duplicated by Bug report #21088: QGIS 3.4 much much slower to delete shapes added*

**#2 - 2019-01-25 01:04 PM - Giovanni Manghi**

*- Regression? changed from No to Yes*

*- Affected QGIS version changed from 3.4.0 to 3.4.4*

*- Priority changed from Normal to High*

**#3 - 2019-02-01 01:41 PM - Peter Gipper**

To clarify, it is no problem to wait 1 minute for an algorithm to finish, the problem is when the feature count goes up towards 100 000, then it takes an hour instead of a minute to process.

**#4 - 2019-02-07 09:00 PM - Even Rouault**

*- Assignee set to Even Rouault*

The issue is that in recent GDAL version we have switched to a streaming reader for GeoJSON, which enables to sequentially read arbitrarily large GeoJSON files, instead of ingesting everything in memory. The adverse consequence for that use case which uses random reading is that getting a feature by FID requires to read statistically half the file each time a feature is asked b FID. I'm working on some improvement regarding this.
A workaround is to convert the file priorly to another format like GeoPackage or shapefiles that have efficient random reading by design.

**#5 - 2019-02-08 02:08 PM - Even Rouault**

*- Resolution set to up/downstream*

*- Status changed from Open to Closed*

Fixed in GDAL master per https://github.com/OSGeo/gdal/commit/bd668db37eb6f176226ebbe7efe34cfac86a3cf6a and in release/2.4 per https://github.com/OSGeo/gdal/commit/d6c38adfa28f75da0630f3e3ac26dbb501fc361e

## Files

| | | | |
|---|---|---|---|
| voronoi_test_points_4326.geojson | 766 KB | 2019-01-24 | Peter Gipper |