QGIS Application - Bug report #13502
Processing dissolve now discards all attributes

2015-10-03 10:10 AM - Anita Graser

Status:	Closed		
Priority:	High		
Assignee:	Bernhard Ströbl		
Category:	Processing/QGIS		
Affected QGIS version:	master	**Regression?:**	No
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	fixed/implemented
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	21546

Description

Dissolve used to retain all attributes. Now all attributes are lost and only a count column is left. Probably happened in commit:116089e

This breaks existing models and imho should be reverted.

History

#1 - 2015-10-04 12:14 PM - Bernhard Ströbl

This was done on purpose because the attribute values assigned to the output feature were totally random (values of the first input feature processed), thus useless (IMHO) and therefore dropped. If a dissolve field is entered it is maintained, though.

#2 - 2015-10-04 01:04 PM - Anita Graser

It's not useless if the user ensured that all dissolved featues would have the same values in the attributes of interest (thus it made no difference which feature's values the algorithm happened to pick).

Imho, it's bad practice to change the behavior of tools that other users might have built their workflows with if there is no need to. Keeping Processing training material up to date is a pain because of all the tiny changes breaking models and scripts :(

#3 - 2015-10-06 04:53 AM - Matthias Kuhn

Bernhard, would it be hard to resurrect the old behavior?

For the future it would be great to be able to choose an aggregate function (max, min, mean, median, first) so the generated fields are no longer random (or if first is chosen, at least they are random by choice :)).

#4 - 2015-10-18 07:06 AM - Bernhard Ströbl

Sorry for keeping silent, I was on holidays...

Anita Graser wrote:

> *It's not useless if the user ensured that all dissolved featues would have the same values in the attributes of interest (thus it made no difference which feature's values the algorithm happened to pick).*

Agreed but I think that the normal use case is **one** dissolve field and this is how all dissolve algorithms I am aware of work (that is ArcInfo 8.? and GRASS: https://grass.osgeo.org/grass70/manuals/v.dissolve.html). See it the other way round: What if the fields don't contain the same values? An unaware user *could* think the values in the result are "correct", there is no way to prevent him or her from this mistake. If features have the same value in e.g. three fields you could use one of them to dissolve and simply join the original layer to the result. So I do not really understand the need for keeping all attributes given that in the majority of cases it is useless and error-prone for the unaware.
Breaking existing models is bad and I did not think of it. We could reestablish the old behavior and

1. add another dissolve algorithm and call it e.g. "real dissolve", "new dissolve" or similar
2. wait with the change until a new major version of either QGIS or Processing (and break things then)

What do you think?

Matthias Kuhn wrote:

> *Bernhard, would it be hard to resurrect the old behavior?*

No, see above

> *For the future it would be great to be able to choose an aggregate function (max, min, mean, median, first) so the generated fields are no longer random (or if first is chosen, at least they are random by choice :)).*

The problem I see is that when defining an algorithm you have to define all input. AFAIK processing is not flexible in a way that in analyses the data source and adapts itself. So you could predefine e.g. 5 fields or 10 fields (meaning two inputs for each fields: one text line for the field (because if used in a model you cannot use a combo), the second for the aggregate function (combo)). Would be a great feature, though, if we could agree on e.g. two or three fields.

#5 - 2015-10-18 08:40 AM - Harrissou Santanna

Bernhard Ströbl wrote:

> *Breaking existing models is bad and I did not think of it. We could reestablish the old behavior and 1. add another dissolve algorithm and call it e.g. "real dissolve", "new dissolve" or similar*
> *2. wait with the change until a new major version of either QGIS or Processing (and break things then)*

FWIW, there's a plugin DissolveWithStats that may be worth integrating Processing tools (see issue 2)

#6 - 2015-10-18 11:44 AM - Anita Graser

Bernhard Ströbl wrote:

> *See it the other way round: What if the fields don't contain the same values? An unaware user could* think the values in the result are "correct", there is no way to prevent him or her from this mistake.

True, there are a lot of possibilities to make mistakes with almost any tool we provide.

> *Breaking existing models is bad and I did not think of it. We could reestablish the old behavior*

Yes please.

> *add another dissolve algorithm and call it e.g. "real dissolve", "new dissolve" or similar*

Let's try to find a descriptive name, e.g.

- "dissolve with aggregates" and
- "dissolve dropping attributes"

> *wait with the change until a new major version of either QGIS or Processing (and break things then)*

I'd rather not break things if there's no need to.

> *Matthias Kuhn wrote:*
>
> > *For the future it would be great to be able to choose an aggregate function (max, min, mean, median, first) so the generated fields are no longer random (or if first is chosen, at least they are random by choice :)).*
>
> *The problem I see is that when defining an algorithm you have to define all input. AFAIK processing is not flexible in a way that in analyses the data source and adapts itself. So you could predefine e.g. 5 fields or 10 fields (meaning two inputs for each fields: one text line for the field (because if used in a model you cannot use a combo), the second for the aggregate function (combo)). Would be a great feature, though, if we could agree on e.g. two or three fields.*

Maybe we could have choices for the aggregates, e.g.

- compute max for all numeric fields (yes/no)
- compute min for all numeric fields (yes/no)
- first for all fields (not just numeric) (yes/no)
- ...

#7 - 2015-10-18 11:37 PM - Bernhard Ströbl

Anita Graser wrote:

> *True, there are a lot of possibilities to make mistakes with almost any tool we provide.*

agreed, but we still should improve the software to prevent users from making mistakes

> > *Breaking existing models is bad and I did not think of it. We could reestablish the old behavior*
>
> *Yes please.*
>
> > *add another dissolve algorithm and call it e.g. "real dissolve", "new dissolve" or similar*
>
> *Let's try to find a descriptive name, e.g.*
>
> - *"dissolve with aggregates" and*
> - *"dissolve dropping attributes"*

renaming old Dissolve would break things again, wouldn't it?

> > *wait with the change until a new major version of either QGIS or Processing (and break things then)*
>
> *I'd rather not break things if there's no need to.*

Agreed but IMHO there is a need to break things in this case because 1. Dissolve's result is not as expected (not dropping fields as in other GIS)
2. The tickbox "Dissolve all" is redundand
3. we should not have several algorithms that do similar things and are named similar (IMHO)

I have been thinking further and for me the whole problem arose because a new version of processing (bugfix version because the third number has been raised) has been released which included a new feature. If the new feature would have been released with QGIS 2.12 then it would have been ok. So we should look on this, too. As long as Processing's repo is identical with QGIS' repo then releasing seperate versions of processing does not make sense. Any bugfixes to Processing must go in QGIS bugfix versions any new features must wait until next QGIS release - or: Have processing in a seperate repo and release it independently from QGIS; when preparing a QGIS release include the latest Processing release.

> *Maybe we could have choices for the aggregates, e.g.*
> - *compute max for all numeric fields (yes/no)*
> - *compute min for all numeric fields (yes/no)*
> - *first for all fields (not just numeric) (yes/no)*
> - *...*

I am not convinced that that would be what users might need, I assume they would like to choose the aggregate function for every field seperately as Matthias described it

#8 - 2015-10-19 12:42 AM - Anita Graser

Bernhard Ströbl wrote:

> *I have been thinking further and for me the whole problem arose because a new version of processing (bugfix version because the third number has been raised) has been released which included a new feature. If the new feature would have been released with QGIS 2.12 then it would have been ok.*

Imho, breaking the API, even "only" Processing API should be done as rarely as possible and for majority of cases limited to major releases (so next time for QGIS 3.x) because the possibility of broken models with each minor release (3 times a year!) reduces the the usefulness of Processing severely.

#9 - 2015-10-19 01:42 AM - Bernhard Ströbl

I think we agree that - first number changed = api break
- second number changed = new feature
- third number changed = bug fix

However I experienced api break e.g. in snapping api (Python) from QGIS 2.6 to 2.8
I experienced my processing models not being readable anymore in (was it QGIS 2.6?) that I saved in the previous version. Therefore my impression is that this scheme is not handled that strictly in the project, which would allow the new Dissolve to be introduced with QGIS 2.12. That is what I thought would happen because it was merged in the current release cycle, I was not aware that it will be included in a bug-fix release of the processing plugin. That is where the problem comes from and it should be well considered when releasing Processing independently from QGIS.
So the proposal would be to:
1. reestablish old Dissolve (naming question still not decided, AFAIK to not break anything it should be "Dissolve")
2. mark this as being deprecated (is there a special way to do so in Processing?)
3. reintroduce new dissolve under what name?
4. remove old dissolve and rename new dissolve to "Dissolve" when changing to QGIS 3.0 (who will be responsible for that?)

#10 - 2015-10-19 02:36 AM - Anita Graser

Bernhard Ströbl wrote:

> *So the proposal would be to: 1. reestablish old Dissolve (naming question still not decided, AFAIK to not break anything it should be "Dissolve")*

+1

> *1. mark this as being deprecated (is there a special way to do so in Processing?)*

Not possible currently as far as I know.

> *1. reintroduce new dissolve under what name?*

Maybe something like "Dissolve with aggregates (will replace normal Dissolve in QGIS 3.0)" if it will contain such functionality.

> *1. remove old dissolve and rename new dissolve to "Dissolve" when changing to QGIS 3.0 (who will be responsible for that?)*

I suggest opening a ticket for this an assigning it to you or me or Victor or ...

#11 - 2015-10-19 02:49 AM - Bernhard Ströbl

Anita Graser wrote:

> > *1. mark this as being deprecated (is there a special way to do so in Processing?)*
>
> *Not possible currently as far as I know.*

So I add it to processing log

> > *1. reintroduce new dissolve under what name?*
>
> *Maybe something like "Dissolve with aggregates (will replace normal Dissolve in QGIS 3.0)" if it will contain such functionality.*

I doubt we will have aggregates in processing because when defining dissolve it is unknown how many suitable fields there are in the input layer

> > *1. remove old dissolve and rename new dissolve to "Dissolve" when changing to QGIS 3.0 (who will be responsible for that?)*
>
> *I suggest opening a ticket for this an assigning it to you or me or Victor or ...*

ok, will do that, should be a blocker then.

#12 - 2015-10-19 04:39 AM - ujaval gandhi

I agree with Anita that we should not change the current behavior.

For the 'dissolve with aggregate' algorithm, why not let the user pick the field(s) on which to compute the aggregates? It is a popular user request and I had written a processing script to do something similar.
https://github.com/spatialthoughts/qgis-tutorials/blob/master/resources/en/docs/processing_python_scripts/scripts/dissolve_with_sum.py

#13 - 2015-10-19 06:31 AM - Bernhard Ströbl

The problem is not to have *one* aggregate field but to enable the user to aggregate every field with the aggregate function he wants because you have to define every input for processing when you code the algorithm. If you provide e.g. 10 fields the UI will be too large and still someone will ask for 11 :-)
Furthermore: you cannot pick fields when the algorithm is used in a model because the input layer is undefined at the point when you build the model.

#14 - 2015-10-22 06:09 AM - Bernhard Ströbl

- Resolution set to fixed/implemented

- Status changed from Open to Closed

changes have been reverted in commit:f705154