

Status:	Closed	
Priority:	Normal	
Assignee:	Alessandro Pasotti	
Category:	GUI	
Affected QGIS version:	3.4.5	Regression?: No
Operating System:	Windows	Easy fix?: No
Pull Request or Patch applied:	No	Resolution: fixed/implemented
Crashes QGIS or corrupts data:	No	Copied to github as #: 29226
Description		
<p>I'm not sure it's a bug, but it's a problem for centralized parameter management.</p> <p>if the user adds a custom variable, all variables are rewritten in QGIS.ini</p> <p>Any future changes to the relevant variables in the global_settings are no longer taken into consideration.</p> <p>QGIS reading the QGIS.ini after the global_settings.ini</p>		

History

#1 - 2019-02-27 04:35 PM - Alessandro Pasotti

- Status changed from Open to Rejected
- Resolution set to invalid

That is by design: global\_settings is meant to provide default values, not to prevent users to set/override a setting with their own values.

#2 - 2019-02-28 07:29 AM - Alain FERRATON

It is not a question of preventing a user from overwriting a parameter with his own values if it is a deliberate act.

But the system as it is currently designed freezes all parameters in the QGIS.INI if only one change is made by a user. Any later changes to the default settings in the global\_settings are then no longer taken into account, except for the addition of new variables.

#3 - 2019-02-28 07:38 AM - Alessandro Pasotti

Do you mean array parameters?

Can you please show an example?

#4 - 2019-02-28 07:48 AM - Alessandro Pasotti

- Status changed from Rejected to Feedback
- Resolution deleted (invalid)

#5 - 2019-02-28 07:50 AM - Alessandro Pasotti

- Assignee set to Alessandro Pasotti

**#6 - 2019-02-28 07:53 AM - Alain FERRATON**

The example I am confronted with is the following:

we add in global\_settings a path to a plugin directory:

```
customEnvVars=customEnvVars, "overwrite|QGIS_PLUGINPATH=T:\\configuration_qgis\\plugins"
```

if the user adds a variable to describe a service name for its layout. All variables, including QGIS\_PLUGINPATH are written in QGIS.INI

If QGIS\_PLUGINPATH is modified in global\_settings, the new value is overwritten by the value stored in QGIS.INI

**#7 - 2019-02-28 09:05 AM - Alessandro Pasotti**

Ok, I see what's happening, I'm not sure this case was in the original specification but it makes sense to me: if a value has not changed from the default value it should not be stored in user settings.

Would this solve your issue?

**#8 - 2019-02-28 09:44 AM - Alessandro Pasotti**

- Status changed from Feedback to In Progress

- Pull Request or Patch supplied changed from No to Yes

PR <https://github.com/qgis/QGIS/pull/9309>

If you want you can add your opinion there: this is really a change in the behavior of the QgsSettings system more than a bug fix.

**#9 - 2019-02-28 10:28 AM - Alain FERRATON**

Thank you. I think this new approach solves problems.

As we focus on the deployment of QGIS 3.4 LTR, is it possible to backport the modification?

**#10 - 2019-02-28 12:54 PM - Alain FERRATON**

Finally I am not sure that the proposed patch solves my particular case.

Indeed, if a user adds a custom variable, he modifies custEnvVars.

The test line 287 will therefore be true and QGIS\_PLUGINPATH will be stored in QGIS.ini.

What do you think about?

**#11 - 2019-02-28 01:15 PM - Alessandro Pasotti**

Sorry but I don't understand: if the user adds/modify custEnvVars (on any other setting) it will be stored in the user settings and the defaults from global settings will not be used.

That's the idea behind the global settings: to be able to provide default values.

For any other settings manipulation you'd need more sophisticated approaches, I did some work on that in the past, you need to start from a startup.py script that manipulates the user settings before QGIS starts.

**#12 - 2019-02-28 01:43 PM - Alain FERRATON**

The difficulty with custom environment variables is that you may want some to be stored in global\_setting.ini while leaving the user the option to add their own.

Unfortunately QGIS stores all environment variables as a single string in CustEnvVars which is therefore necessarily in QGIS.INI as soon as the user adds a local variable.

I understand that this is sophisticated approach.

Startup.py may be a solution...

I am interested by examples if they exist....

**#13 - 2019-02-28 02:11 PM - Alessandro Pasotti**

This is my old work: [https://connect.boundlessgeo.com/docs/desktop/latest/system\\_admins/init\\_scripts.html](https://connect.boundlessgeo.com/docs/desktop/latest/system_admins/init_scripts.html) it was developed for QGIS 2 but you get the idea.

**#14 - 2019-03-04 12:52 PM - Alain FERRATON**

Je crains que le correctif n'ait produit un effet de bord indésirable  
see #21464

**#15 - 2019-03-04 12:53 PM - Alain FERRATON**

I'm afraid the patch has produced an undesirable side effect  
see #21464

**#16 - 2019-04-01 05:31 PM - Alessandro Pasotti**

- Status changed from In Progress to Closed
- Resolution set to fixed/implemented