QGIS Application - Feature request #6269
Better tokens for text replacement in Print Composer labels

2012-08-28 11:01 AM - Larry Shaffer

Status:	Closed		
Priority:	Normal		
Assignee:	Marco Hugentobler		
Category:	Map Composer/Printing		
Pull Request or Patch supplied:	No	**Resolution:**	fixed/implemented
Easy fix?:	No	**Copied to github as #:**	15574

Description

See also: Issue #1385

While there have been several requests for new 'tokens' for Composer labels that get replaced with data (e.g. $CURRENT_DATE), the concept of adding support for new tokens in code could be supplanted with a more extensible solution.

I have outlined two such solutions here: http://gis.stackexchange.com/questions/31090

To summarize from that post, the first solution is to have output from a customized Expression Editor be the label's text, and the second is to have a Python snippet, or specifically named functions in a snippet, that replace tokens.

I'm in favor of the Python snippet with tokens being replaced with the returned value of a similarly named function.

For example:

```
import getpass
from qgis.core import *

def pytoken_user():
  return getpass.getuser()

def pytoken_title():
  return QgsProject.instance().title()
```

would replace any number of in-text tokens named $pytoken_user or $pytoken_title.

For security reasons, the code would probably not be auto-run on Composer launch (could be an option to allow always, though, like new Project Macros).

History

#1 - 2012-08-29 01:00 AM - Regis Haubourg

Hi Larry,
I'm about to support solution 1 using expression builder. What about having both solutions? Having Expression builder for common users, and python snippet for advanced dev's ?
Régis

#2 - 2012-08-29 05:42 AM - Hugo Mercier

Hi.

If my understanding is correct, we can still stick with something that stay compliant with the two approaches: why not consider a new special token, that may be edited via the Expression Editor, named for instance '$CUSTOM_PYTHON()', with an inline Python string as argument or a reference to an external Python file ?

#3 - 2012-08-29 11:08 PM - Marco Hugentobler

It will be great to have support for advanced replacement options in QgsComposerLabel.

One issue with the python approach is that QgsComposerLabel is also used for QGIS server printing (GetPrint request). In a server environment, it is desirable to have a solution working without python (old LTS distros are common where everything needs to be compiled from source).

#4 - 2012-08-30 02:17 AM - Larry Shaffer

- File label-tokens_treeview.png added
- File label-tokens_button.png added

Marco Hugentobler wrote:

> *It will be great to have support for advanced replacement options in QgsComposerLabel.*

Yes, especially if it is extensible, even if that means adding new sections/functions to a custom Expression Editor, as they are requested.

> *One issue with the python approach is that QgsComposerLabel is also used for QGIS server printing (GetPrint request). In a server environment, it is desirable to have a solution working without python (old LTS distros are common where everything needs to be compiled from source).*

This did not occur to me. Best to skip Python for now, until some Expression Editor solution is in place, which should work well for (I'm guessing) 95% of general use of custom tokens. Then, adding the ability to inline Python in the Editor's output, or as an inline token in the label's text, would help, if it makes sense to add, with respects to QGIS Server's base requirements.

Concerning an implementation, I figured tokens would be something stored/available at the project level. Then, they could be reused across multiple Composers, and imported from other project files. If purely custom, named tokens were used, instead of in-code support (like CURRENT_DATE is now), only new sections/functions for the Expression Editor would need to be focused on, after the initial implementation is completed.

Here are some quick mock-ups of GUI elements:

label-tokens_button: shows access to tokens from main toolbox panel.

label-tokens_treeview: shows a tree view widget listing available project tokens. Double-clicking on row would open the Expression Editor (like with rule-based symbols tree view). Selecting a row and clicking Paste would paste the token in the text edit field for the label at the current cursor, e.g. paste $ProjectTitle. Or, the Paste button could be Edit instead, and double-click would paste a token.

#5 - 2012-08-30 08:08 AM - Marco Hugentobler

> *Concerning an implementation, I figured tokens would be something stored/available at the project level. Then, they could be >reused across multiple Composers, and imported from other project files. If purely custom, named tokens were used, instead of >in-code support (like CURRENT_DATE is now), only new sections/functions for the Expression Editor would need to be focused on, >after the initial implementation is*

> *completed.*

The GUI and the expression management looks good. Maybe the expressions can be shared between the composer labels and other parts that use expressions (though the label cannot easily deal with attribute columns / feature area / coordinates. Don't know how that should be handled).

#6 - 2012-11-19 02:49 AM - Regis Haubourg

Now that expression is commited in master, I understand that very few things are missing to close this issue. Maybe adding some extra properties un expression builder could help. (Project properties for title , OS properties for UserName...)
Do we still keep this for 2.0 or can we postpone?

#7 - 2014-04-06 06:04 AM - Jürgen Fischer
- Category changed from 33 to Map Composer/Printing

#8 - 2014-06-28 07:42 AM - Jürgen Fischer
- Target version changed from Version 2.0.0 to Future Release - Lower Priority

#9 - 2015-12-13 11:31 AM - Médéric RIBREUX
- Status changed from Open to Feedback

Hello, bug triage...

Let sum it up. On QGIS 2.13 master we have:

- Can insert expressions in the composer.
- In the expressions, we can use:
  - all expression functions.
  - all atlas variables.
  - all project/QGIS variables.
  - a function written in the editor.

With all of these features, I think we can close this bug ?

#10 - 2015-12-14 01:12 PM - Nyall Dawson
- Resolution set to fixed/implemented
- Status changed from Feedback to Closed

Files

label-tokens_button.png	4.16 KB	2012-08-30	Larry Shaffer
label-tokens_treeview.png	22.5 KB	2012-08-30	Larry Shaffer