Redmine - Feature #25625

Turn issue numbers into UUIDs

2017-04-17 19:34 - Pavel Kretov

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Issues	**Estimated time:**	0.00 hour
Target version:			
Resolution:			

Description

Problem with issue numbering

It seems like global sequental issue numbering in Redmine makes a lot of people unhappy; they filled a number of feature requests to advocate PROJECT-01 scheme support (that's Jira-style, I guess), for example: #282, #1926, and #2885.

The core developers were always reluctant to adopt such a scheme, because it doesn't allow issues to be easily moved between projects. Somebody even wrote a plugin which does exactly that thing (I don't know how reliable it is).

However, there is a feature which is worth considering important: the ability to export and import individual projects, as well as transfer them between different Redmine installations (see #3647). Since issue numbers are often put in commit messages, it's critical for them to never change after moving to another Redmine instance. Unfortunately, neither global numbering, nor PROJECT-01 scheme allows that, since:

- small global numbers (#1, #2, and so on) are likely to be already taken on the target server;
- people tend to use short project names which are not unlikely to collide.

Proposed solution

I think that the only reliable approach to the problem would is to use UUIDs instead of plain numbers. Nobody is surprised about Git's SHA-1 revision hashes, why issue numbers are different? Strings like

```
http://www.redmine.org/issues/af666b6a-4245-43dc-a42b-f3cd44a58e7c
```

are unlikely to ever clash with another Redmine issue in the world, but still short enough to serve as URLs; they can be easily put into commit messages on a separate line. These UUIDs can be abbreviated when presented to the user on the web interface, while commit messages, comments and wiki sources should always refer to full UUIDs.

Of course, these IDs are harder to memorize than plain numbers, but they have an inherent beauty of some kind:

- they are all created equal (most people will find #42 better then #97014),
- they are not leaving the impression of an imperfection when a project has its issues numbered with large gaps.

Unfortunately, this would apparently be a breaking change. Maybe, 4.x could afford it?

P.S.

BTW, the UUID approach automatically solves #6642 at the same time.

Related issues:

Related to Redmine - Feature #2885: A segregated numbering per project	**New**	**2009-03-02**
Related to Redmine - Feature #3647: Data Import/Export System	**New**	**2009-07-21**
Related to Redmine - Feature #6642: anonymized issue numbers	**New**	**2010-10-12**

History

#1 - 2017-05-17 14:52 - Toshi MARUYAMA

- Related to Feature #2885: A segregated numbering per project added

#2 - 2017-05-17 14:53 - Toshi MARUYAMA

- Related to Feature #3647: Data Import/Export System added

#3 - 2017-05-17 14:53 - Toshi MARUYAMA

- Related to Feature #6642: anonymized issue numbers added