

Redmine - Patch #39857

Optimize users visibility check

2023-12-15 15:40 - Pavel Rosický

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Go MAEDA	% Done:	0%
Category:	Performance	Estimated time:	0.00 hour
Target version:	6.0.0		
Description			
<pre>diff --git a/app/models/principal.rb b/app/models/principal.rb index 4cce97e26..25a79d768 100644 --- a/app/models/principal.rb +++ b/app/models/principal.rb @@ -48,8 +48,8 @@ class Principal < ActiveRecord::Base all else view_all_active = false - if user.memberships.to_a.any? - view_all_active = user.memberships.any? { m m.roles.any? { r r.users_visibility == 'all' + if user.memberships.any? + view_all_active = User.where(id: user.id).joins(memberships: :roles).where("#{Role.table_ name}.users_visibility = ?", 'all').any? else view_all_active = user.builtin_role.users_visibility == 'all' end end</pre>			
<p>in a bad scenario <code>user.memberships.any? { m m.roles...</code> is (n * memberships without permissions + 1 queries) which is very slow. Let the database do the job.</p> <p>unfortunately in an ideal scenario (where the very first membership has the <code>users_visibility == 'all'</code>) performance could be slightly worse, but I think it's an acceptable tradeoff</p>			

Associated revisions

Revision 22567 - 2023-12-27 09:07 - Go MAEDA

Optimize users visibility check (#39857).

Patch by Pavel Rosický.

History

#1 - 2023-12-25 16:06 - Go MAEDA

- Target version set to 6.0.0

#2 - 2023-12-27 09:07 - Go MAEDA

- Subject changed from *Optimize users visibility* to *Optimize users visibility check*

- Status changed from *New* to *Closed*

- Assignee set to *Go MAEDA*

Committed the patch. Thank you.