redmineorg-copy202205 - Vote #79613

add_working_days returns wrong date

2022/05/09 18:38 - Admin Redmine

ステータス:	Confirmed	**開始日:**	2022/05/09
優先度:	通常	**期日:**	
担当者:		**進捗率:**	0%
カテゴリ:	Issues_2	**予定工数:**	0.00時間
対象バージョン:	Candidate for next minor release_33	**作業時間:**	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/29855	**status_id:**	9
category_id:	2	**tracker_id:**	1
version_id:	33	**plus1:**	0
issue_org_id:	29855	**affected_version:**	
author_id:	367447	**closed_on:**	
assigned_to_id:	332	**affected_version_id:**	138
comments:	14		

説明

@Redmine::Util::DateCalculation#add_working_days(date, n)@ returns wrong date when @date@ is holiday and @n@ is a multiple of 5.

Example:

```
irb(main):004:0> Setting.non_working_week_days
=> ["6", "7"]
irb(main):001:0> include Redmine::Utils::DateCalculation
irb(main):002:0> add_working_days(Date.new(2018, 10, 27), 5)
=> Mon, 05 Nov 2018       # Expected Fri, 02 Nov 2018
irb(main):003:0> add_working_days(Date.new(2018, 10, 28), 5)
=> Mon, 05 Nov 2018       # Expected Fri, 02 Nov 2018
```

Tested with @trunk@17598@

journals

I have confirmed that 3.3-stable and 3.4-stable are also affected.

I think that applying this patch will solve the problem.
The code of the add_working_days method changes quite a bit, but all the tests succeed.

Any feedback is welcome.

The suggested fix works fine but it is much slower than the current code. I think we need to consider whether this will affect the performance of Redmine.

```
$ bin/rails r bench-29855.rb
Warming up --------------------------------------
                              before    12.236k i/100ms
                               after   997.000  i/100ms
Calculating -------------------------------------
                              before    159.524k (± 4.7%) i/s -    807.576k in   5.073660s
                               after     10.597k (± 3.4%) i/s -     53.838k in   5.086474s

Comparison:
                              before:   159524.1 i/s
```

```
               after:      10597.0  i/s  -  15.05x    slower

require  'benchmark/ips'

include  Redmine::Utils::DateCalculation

Benchmark.ips  do  |x|
  x.report('before')  do
    add_working_days(Date.today,  30)
  end

  x.report('after')  do
    result  =  Date.today
    30.times  do
      result  =  next_working_date(result  +  1)
    end
    result
  end

  x.compare!
end
```

-- Jean-Philippe, do you think we can accept this performance deterioration? I think it is OK because 'add_working_days' method will not be executed hundreds of times by the user's single operation. So, it does not affect the performance of Redmine. -- Mizuki ISHIKAWA wrote: > Any feedback is welcome. @DateCalculation#working_days@ should be fixed in a similar way to be consistent with the proposed fix. These new assertions should pass:

```
Index:  test/unit/lib/redmine/utils/date_calculation.rb
===================================================================
---  test/unit/lib/redmine/utils/date_calculation.rb  (revision  17671)
+++  test/unit/lib/redmine/utils/date_calculation.rb  (working  copy)
@@  -41,6  +41,8  @@
       assert_working_days   8,  '2012-10-11',  '2012-10-23'
       assert_working_days   2,  '2012-10-14',  '2012-10-17'
       assert_working_days  11,  '2012-10-14',  '2012-10-30'
+      assert_working_days   5,  '2012-10-20',  '2012-10-26'
+      assert_working_days   5,  '2012-10-21',  '2012-10-26'
     end
   end
```

I took a look and there are some strange (or wrong) test cases the we should review before changing anything else.

Taking the following test scenario:

```
  def  test_working_days_with_non_working_week_days
    with_settings  :non_working_week_days  =>  %w(6  7)  do
      assert_working_days  14,  '2012-10-09',  '2012-10-27'
      assert_working_days   4,  '2012-10-09',  '2012-10-15'
      assert_working_days   4,  '2012-10-09',  '2012-10-14'
      assert_working_days   3,  '2012-10-09',  '2012-10-12'
      assert_working_days   8,  '2012-10-09',  '2012-10-19'
      assert_working_days   8,  '2012-10-11',  '2012-10-23'
      assert_working_days   2,  '2012-10-14',  '2012-10-17'
      assert_working_days  11,  '2012-10-14',  '2012-10-30'
    end
  end
```

@assert_working_days 4, '2012-10-09', '2012-10-15'@
2012-10-09 was Tuesday
2012-10-15 was Monday
The number of the expected working days according to the test is 4. But in my opinion, it should be 5 days (Tuesday, Wednesday, Thursday, Friday and Monday). 4 could be only if we exclude the end date from the count. if we do this, than the number of the expected days for the 2 assertions proposed by Jean-Philippe should be 4 because we need to exclude Friday (2012-10-26).

Also, it sound incorrect to say that between '2012-10-09 - 2012-10-15 (Tuesday - Monday)' and '2012-10-09 - 2012-10-14 (Tuesday - Sunday)' are the same number of working days (4).

Jean-Philippe, what do you think? I'm in favour of including the end date in the count.

Marius BALTEANU wrote:

> The number of the expected working days according to the test is 4. But in my opinion, it should be 5 days (Tuesday, Wednesday, Thursday, Friday and Monday). 4 could be only if we exclude the end date from the count. if we do this, than the number of the expected days for the 2 assertions proposed by Jean-Philippe should be 4 because we need to exclude Friday (2012-10-26).

@#working_days@ and @#add_working_days@ are used to reschedule an issue when the start date is changed. Its duration is calculated with @#working_days@ and the new due date is calculated with @#add_working_days@. If there is no "non working day", they should behave like @Date#-@ and @Date#+@.

Jean-Philippe Lang wrote:

> @#working_days@ and @#add_working_days@ are used to reschedule an issue when the start date is changed. Its duration is calculated with @#working_days@ and the new due date is calculated with @#add_working_days@. If there is no "non working day", they should behave like @Date#-@ and @Date#+@.

Thanks, but are still not clear for me the expected results so I'll leave Go Maeda or Mizuki ISHIKAWA to fix this issue.

---

related_issues

relates,Closed,14846,Calculation of the start date of following issues ignores the "non-working days" setting

履歴

#1 - 2022/05/10 17:04 - Admin Redmine

- カテゴリ を Issues_2 にセット

- 対象バージョン を Candidate for next minor release_33 にセット