

## Redmine - Defect #23511

### Progress of parent task should be calculated using total estimated hours of children

2016-08-04 22:59 - Zouheir Najai

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Jean-Philippe Lang	<b>% Done:</b>	0%
<b>Category:</b>	Issues	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	3.4.0	<b>Affected version:</b>	3.2.3
<b>Resolution:</b>	Fixed		

#### Description

Problem of calculating the percentage of completion of a Issue(see the attached file).

When "child" has "sub-child": estimated\_hours is nil, therefore is not accounted in "average" (line 1534).

Issue.total\_estimated\_hours has to be accounted in this estimate.

app/model/issue.rb:

```
1511 def recalculate_attributes_for(issue_id)
1512   if issue_id && p = Issue.find_by_id(issue_id)
1513     if p.priority_derived?
1514       # priority = highest priority of children
1515       if priority_position = p.children.joins(:priority).maximum("#{IssuePriority.table_name}.position")
1516         p.priority = IssuePriority.find_by_position(priority_position)
1517       end
1518     end
1519
1520     if p.dates_derived?
1521       # start/due dates = lowest/highest dates of children
1522       p.start_date = p.children.minimum(:start_date)
1523       p.due_date = p.children.maximum(:due_date)
1524       if p.start_date && p.due_date && p.due_date < p.start_date
1525         p.start_date, p.due_date = p.due_date, p.start_date
1526       end
1527     end
1528
1529     if p.done_ratio_derived?
1530       # done ratio = weighted average ratio of leaves
1531       unless Issue.use_status_for_done_ratio? && p.status && p.status.default_done_ratio
1532         child_count = p.children.count
1533         if child_count > 0
1534           average = p.children.where("estimated_hours > 0").average(:estimated_hours).to_f
1535           if average == 0
1536             average = 1
1537           end
1538           done = p.children.joins(:status).
1539             sum("COALESCE(CASE WHEN estimated_hours > 0 THEN estimated_hours ELSE NULL END, #{average}) "
+
1540             "** (CASE WHEN is_closed = #{self.class.connection.quoted_true} THEN 100 ELSE
COALESCE(done_ratio, 0) END)").to_f
1541           progress = done / (average * child_count)
1542           p.done_ratio = progress.round
```

```

1543     end
1544     end
1545     end
1546
1547     # ancestors will be recursively updated
1548     p.save(:validate => false)
1549     end
1550 end

```

Possible Fix:

```

--- issue.rb.ori      2016-08-01 11:44:53.220332666 -0400
+++ issue.rb      2016-08-01 11:45:07.395332666 -0400
@@ -1531,13 +1531,12 @@
     unless Issue.use_status_for_done_ratio? && p.status && p.status.default_done_ratio
       child_count = p.children.count
       if child_count > 0
-         average = p.children.where("estimated_hours > 0").average(:estimated_hours).to_f
+         average = p.total_estimated_hours / child_count
         if average == 0
           average = 1
         end
-         done = p.children.joins(:status).
-         sum("COALESCE(CASE WHEN estimated_hours > 0 THEN estimated_hours ELSE NULL END, #{average}) " +
-         " * (CASE WHEN is_closed = #{self.class.connection.quoted_true} THEN 100 ELSE COALESCE(done_ratio, 0)
END)").to_f
+         done = 0
+         p.children.each { |c| done += (c.total_estimated_hours * c.done_ratio) }
         progress = done / (average * child_count)
         p.done_ratio = progress.round
       end

```

When one manually enter the estimated time.

In the original program, based on the average of other tasks to calculate progress. I keep the same approach in the patch :

```

--- issue.rb.ori      2016-08-01 11:44:53.220332666 -0400
+++ issue.rb      2016-08-01 13:42:18.138330460 -0400
@@ -1531,13 +1531,19 @@
     unless Issue.use_status_for_done_ratio? && p.status && p.status.default_done_ratio
       child_count = p.children.count
       if child_count > 0
-         average = p.children.where("estimated_hours > 0").average(:estimated_hours).to_f
+         average = (p.total_estimated_hours||=0) / child_count
         if average == 0
           average = 1
         end
-         done = p.children.joins(:status).
-         sum("COALESCE(CASE WHEN estimated_hours > 0 THEN estimated_hours ELSE NULL END, #{average}) " +
-         " * (CASE WHEN is_closed = #{self.class.connection.quoted_true} THEN 100 ELSE COALESCE(done_ratio, 0)
END)").to_f
+         done = 0
+         p.children.each do |c|
+           dr = c.closed_on.nil? ? (c.done_ratio||=0) : 100

```

```
+ unless c.total_estimated_hours.nil? || c.total_estimated_hours == 0
+   done += (c.total_estimated_hours * c.done_ratio)
+ else
+   done += average * c.done_ratio
+ end
+ end
progress = done / (average * child_count)
p.done_ratio = progress.round
end
```

#### Related issues:

Related to Redmine - Defect # 20995: Automatic done ratio calculation in issu...	<b>Closed</b>
Related to Redmine - Defect # 24457: Progress of version should be calculated...	<b>New</b>
Duplicated by Redmine - Defect # 24213: % Done is not correct for parent to p...	<b>Closed</b>
Duplicated by Redmine - Defect # 23151: done_ratio calculation with multi-lev...	<b>Closed</b>

#### Associated revisions

**Revision 15802 - 2016-09-01 19:03 - Jean-Philippe Lang**

Fixed that progress of parent should be calculated with total estimated hours of children (#23511).

#### History

**#1 - 2016-08-23 04:41 - Jens Krämer**

- File 0001-test-case-for-grandchildren-s-done-percentages.patch added
- File 0002-fix-attempt-for-23511.patch added

This is a valid problem.

Given

```
Settings.issue_done_ratio == 'issue_field'
Settings.parent_issue_done_ratio == 'derived'
```

and issues like this:

- parent issue
  - child 1 (2h estd, 0% done)
  - child 2
    - child a (2h estd, 50% done)
    - child b (2h estd, 50% done)

so in total 2h of 6h are done, which means the parent should have a done ratio of 33%. Redmine 3.2 and current master show 25% instead (average of 50% from child 2 and 0% from child 1).

It looks like this new issue has been introduced with my fix for #20995 which changed the calculation from leaves to children in the first place to solve another problem.

I don't have an idea how to solve this without breaking anything else right now. For now I'll attach a failing test case illustrating the problem and a second patch containing the fix attempt from this issue's description, which solves the problem at hand but breaks some other test cases.

**#2 - 2016-08-23 15:20 - Toshi MARUYAMA**

- Description updated

**#3 - 2016-08-23 15:24 - Toshi MARUYAMA**

- Related to Defect #20995: Automatic done ratio calculation in issue tree is wrong in some cases added

**#4 - 2016-08-23 15:24 - Toshi MARUYAMA**

- Target version set to 3.4.0

**#5 - 2016-09-01 19:05 - Jean-Philippe Lang**

- Subject changed from Problem of calculating the percentage of completion of a Issue to Progress of parent task should be calculated using total estimated hours of children

- Status changed from New to Closed

- Assignee set to Jean-Philippe Lang

- Resolution set to Fixed

Fixed in r15802 by calculating the progress of parent task using the total estimated hours of children as proposed by Zouheir Najai. I've included the test provided by Jens.

Thanks to both of you.

**#6 - 2016-11-21 14:02 - Toshi MARUYAMA**

- Duplicated by Defect #24213: % Done is not correct for parent to parents when calculated from subtasks added

**#7 - 2016-11-25 15:41 - Toshi MARUYAMA**

- Related to Defect #24457: Progress of version should be calculated the same way as parent tasks added

**#8 - 2017-07-09 05:24 - Go MAEDA**

- Duplicated by Defect #23151: done\_ratio calculation with multi-level sub tasks and estimated hours added

**Files**

---

Problem in calculating percentage.pdf	96 KB	2016-08-04	Zouheir Najai
0001-test-case-for-grandchildren-s-done-percentages.patch	1.29 KB	2016-08-23	Jens Krämer
0002-fix-attempt-for-23511.patch	1.76 KB	2016-08-23	Jens Krämer