

Redmine - Feature #5490

Option for independent subtask priority/start date/due date/done ratio

2010-05-10 10:07 - Anders Aagaard

Status:	Closed	Start date:	2010-05-10
Priority:	High	Due date:	
Assignee:	Jean-Philippe Lang	% Done:	0%
Category:	Issues	Estimated time:	0.00 hour
Target version:	3.1.0		
Resolution:	Fixed		
Description			
<p>In the subtasks patch in trunk a subtask controls the parent task. Priority/start date/due date is completely controlled by subtasks.</p> <p>I'd like some discussion on whether or not this is a good idea. While on some part it's practical to have priority tracked like this. Doing it takes away the "identity" of the original issue.</p> <p>It feels like subtasks control the parent task, not the other way around. This is probably practical when you have a parent task with loads of subtasks, but when I have a parent task with 1 subtask to "do some little side thing" it's very unpractical.</p> <p>I'd like to suggest freely changing priority/start date/due date for parent tasks and subtasks. But subtasks priority/due date can not be set "under" the parent task. Meaning if a parent task has priority high the subtask can have whatever priority it wants, but not under high.</p>			
Related issues:			
Related to Redmine - Feature #6687: Making an issue a subtask leads to loss o...		New	2010-10-18
Related to Redmine - Feature #5875: Changes to child estimates should trigger...		New	2010-07-12
Related to Redmine - Defect #6847: Parent priorities not dropping when subtas...		Closed	2010-11-08
Related to Redmine - Defect #5880: Only consider open subtasks when computing...		Closed	2010-07-14
Related to Redmine - Feature #6167: Add option to weight parent task %-done b...		Closed	2010-08-18
Related to Redmine - Defect #16092: Parent/subtask: calculation of estimated ...		Closed	
Related to Redmine - Defect #19914: Conflicting Fields Permission X Workflow ...		New	
Has duplicate Redmine - Defect #5905: "Due date" from a ticket is readonly af...		Closed	2010-07-19
Has duplicate Redmine - Defect #11188: Unable to change "priority", "start da...		Closed	
Has duplicate Redmine - Defect #11598: Priority automatically changes when yo...		Closed	
Has duplicate Redmine - Defect #14118: When creating a subtask, the priority ...		Closed	

Associated revisions

Revision 14269 - 2015-05-25 11:53 - Jean-Philippe Lang

Adds settings to control start/due dates and priority on parent tasks (#5490).

Revision 14270 - 2015-05-25 12:03 - Jean-Philippe Lang

Adds a setting to control done ratio on parent tasks (#5490).

History

#1 - 2010-05-18 10:57 - Anders Aagaard

Been working more with this functionality, and I find that limit less and less practical.

When we have one person assigned to the parent task, that's responsible for tracking it. It becomes unpractical to manage.

#2 - 2010-05-21 09:34 - Anders Aagaard

Another thing is tasks where all the subtasks are closed/rejected, and there's only the main task left. Then you have no way of setting priority/start date/end date.

#3 - 2010-05-31 12:44 - Daniel N

Anders Aagaard wrote:

Another thing is tasks where all the subtasks are closed/rejected, and there's only the main task left. Then you have no way of setting priority/start date/end date.

Exactly. Another point is that only the first subtask sets parents due dates. Other subtasks with earlier due dates didn't update parents due date. And of course closing child tickets makes updating the parent task impossible. Any workaround or ideas how to "unlink" both dates?

#4 - 2010-05-31 14:59 - Mischa The Evil

- Priority changed from Normal to High

#5 - 2010-06-24 12:48 - Anders Aagaard

Any updates on this? This isn't remotely close to logical behavior.

#6 - 2010-06-24 17:39 - Robert Chady

I also find this behavior to be illogical and counter productive. In fact, in our production system I had to hack this out for the priority as it was completely destroying our workflow. My suggestion is this should be made optional, allowing the admin to select which things are controlled by the child subtask.

#7 - 2010-06-24 18:20 - Anders Aagaard

Could you make a diff for that hack? I don't have time to do it myself right now. And it's a rather big annoyance to both me and my coworkers.

#8 - 2010-06-26 20:04 - Michael Hellein

I strongly agree that this is illogical behavior for a lot of use cases. I'd be happier without the feature, but I'd be satisfied with a way to turn it off (enabling status and due dates on the parent).

#9 - 2010-06-26 20:40 - Michael Hellein

- File `defang_children.diff` added

Here's an artless diff that I believe lets you do whatever you want to an issue. Please be wary of it - I wasn't all that careful!

#10 - 2010-06-26 20:41 - Michael Hellein

Oops. Accidentally left something unrelated in that diff. You ought to be able to figure it out!

#11 - 2010-06-27 09:24 - Anders Aagaard

I see this has "unintended consequences". Have you used this for a while? I don't wanna risk anything that might screw up the database.

If children aren't deleted/moved correctly for example, how easy is it to fix that manually?
Will children with parents deleted still work properly?

#12 - 2010-09-26 17:34 - Enderson Maia

Anders Aagaard wrote:

Been working more with this functionality, and I find that limit less and less practical.

When we have one person assigned to the parent task, that's responsible for tracking it. It becomes unpractical to manage.

In the case of a user *Foo* responsible for the parent tasks and want to delegate some subtasks for other user *Bar*, I suggest in this ticket [#6460](#) that changes in subtasks should send a notification for users associated with the parent task.

I know it's an issue not related to this one, but it involves subtasks, and it's good to discuss ways to improve subtasks in Redmine.

#13 - 2010-11-10 10:39 - Andreas Bosch

+1 - I also think this feature is more painful than useful.

In my eyes, the best solution would be if you could specify in the parent task how its properties should be handled:

1. **independent** (*completely independent parent and child tasks, there is just a relation between them*)
2. **derived from children** (*the way it is now*)
3. **restricted children** (*subtasks' priority/due date cannot be set "under" the parent task's, see last paragraph of issue description*)

#14 - 2010-11-30 11:49 - Eric Voisard

I fully agree.

We just have updated from 0.9.2 to 1.0.4 to benefit from the Subtasks feature, but how it's implemented now, we simply can't use it. We can't have a

subtask changing (shortening) the due date of the main task.

Say we have a main task which is "Development of the new RS-232 interface" which is officially due on 01/01/2011, having a subtask "Make a null modem cable for tests" that shortens the real due date to something closer is nonsense.

I like Andreas proposal. Though, I don't know if these properties should be handled at parent task level, projet level or application level...

Thanks, Eric

#15 - 2010-12-03 22:57 - yannick quenec'hdu

Hi,

+1 for Andreas

The total relation between subtask and task parent is a nonsense for my project.

I use redmine in agile project, I need different priority beetwen user stories (task parent) for my product backlog or sprint. The subtask (a task for development) priority for each story don't have related to the priorities of my user stories.

But the estimated time from each subtask is a total time for my parent task.

So, I need to choose the relation (date not/and priority not/and time) with subtask and parent task.

Thanks
Yannick

#16 - 2011-02-02 05:40 - Shahriar Prosciuttini

Yes Please!

+1 for Andreas

It would be great if this issue was addressed. As it stands, it looks like we'll have to avoid using sub-tasks since they contort the parent's due date.

Otherwise, the system looks pretty slick.

Thanks,
Shahriar.

#17 - 2011-03-22 09:08 - Martin Bächtold

+1

I've been using Redmine for years now but I've got to admit that the subtask module is not usable the way it is now.

#18 - 2011-03-25 16:45 - Eric Voisard

+1

I don't understand the logics behind current implementation. Anyway, subtasks definitely are unusable in our case.

And unfortunately, it looks Michael's patch makes task -> subtasks relations disappear (e.g. from the list in the issue description).

Eric

#19 - 2011-03-28 13:43 - Cassiano Monteiro

+1

I had a due date for a parent task, and it got lost when I created a subtask with a earlier due date. I also think that child tasks should not control the parent task.

#20 - 2011-03-28 14:35 - Andreas Bosch

Any idea when this might be integrated? I think this is a major problem because it makes the subtasks feature completely useless or unusable. It's interesting that noone from the core team seems to care about it... I would implement it myself, but I have no knowledge of Ruby.

#21 - 2011-04-05 12:05 - Nigel Jones

+1

The way subtasks control their parents at the moment is very odd indeed.

#22 - 2011-04-21 13:47 - Evgeny Brednya

+1

Would be great if it can be set on global and by project level as:

- **independent** (completely independent parent and child tasks, there is just a relation between them)
- **derived from children** (the way it is now)

#23 - 2011-04-21 15:38 - Etienne Massip

- Target version set to Unplanned backlogs

#24 - 2011-05-05 18:51 - Siegfried Vogel

+1

Waiting for a better solution, too.

#25 - 2011-05-26 15:38 - Svein-Tore Griff With

+1

Vote for the option where the current solution is kept, but we also keep a separate estimate for the parent task, so that the parent task would have the properties:

Estimated time: 30 hours(editable)
Total estimated time from children: 32 hours

We can do the same for the other properties.

#26 - 2011-09-16 19:53 - Nick Nguyen

I've applied the defang_children.diff to Redmine 1.2.0, but it doesn't seem to work. All subtasks are still setting the parent task's dates and estimated hours. Is there an updated diff? Here is what i did in my issues.rb. I also added the line to the show.rhtml as specified in the diff.

```
# 9/16/2011 Hack to remove subtask dependency
#after_save :reschedule_following_issues, :update_nested_set_attributes, :update_parent_attributes, :create_
journal
after_save :update_nested_set_attributes, :create_journal
# 9/16/2011 Hack to remove subtask dependency
#after_destroy :update_parent_attributes

# HACK!
# This forces Redmine to believe the issue has no children,
# allowing us to update priority and due dates manually.
# WARNING: There are some unintended consequences: children are not deleted, etc.
def leaf
  true
end
```

#27 - 2011-10-13 08:16 - Sølve Monteiro

+1 to this (the whole discussion), I find this feature so counterproductive that I end up not using subtasks, and instead organize them by category, version or even subprojects.

#28 - 2011-10-27 15:40 - Fred Zimmerman

Agreed.

#29 - 2011-10-27 15:41 - Fred Zimmerman

I like the option [#25](#) with dual estimated time.

#30 - 2012-01-04 12:18 - Sridhar P

- File issue5490.patch added

I have tried my hand at this. What it does:

- Creating/updating the sub task doesn't update the start_date, due_date and priority_id
- Allows editing of priority_id, start_date and due_date of an issue even when all of the sub-tasks are closed

Problem is that the unit tests are failing. Failure messages are given below:

```
=====
Loaded suite test/unit/issue_nested_set_test
Started
.....F...F.F
Finished in 15.987187 seconds.
```

1) Failure:

```
test_parent_dates_should_be_lowest_start_and_highest_due_dates (IssueNestedSetTest) [test/unit/issue_nested_set_test.rb:310]:
<Mon, 25 Jan 2010> expected but was
<nil>.
```

2) Failure:

```
test_parent_priority_should_be_the_highest_child_priority (IssueNestedSetTest) [test/unit/issue_nested_set_test.rb:289]:
<"High"> expected but was
<"Normal">.
```

3) Failure:

```
test_reschuling_a_parent_should_reschedule_subtasks (IssueNestedSetTest) [test/unit/issue_nested_set_test.rb:368]:
<[Wed, 02 Jun 2010, Thu, 10 Jun 2010]> expected but was
<[nil, nil]>.
```

24 tests, 77 assertions, 3 failures, 0 errors, 0 skips

Test run options: --seed 28610

=====
All the functional and integration tests have passed though. Attached is the patch.

#31 - 2012-02-29 16:21 - Sven Ludwig

Here is a patch for redmine 1.3, which filters subtasks if closed. So a parent issue would have the highest priority of all active subtasks.

```
--- app/models/issue.rb.orig      2012-02-29 14:49:31.103061065 +0000
+++ app/models/issue.rb         2012-02-29 14:49:56.147060558 +0000
@@ -798,7 +798,7 @@
   def recalculate_attributes_for(issue_id)
     if issue_id && p = Issue.find_by_id(issue_id)
       # priority = highest priority of children
-      if priority_position = p.children.maximum("#{IssuePriority.table_name}.position", :include => :priority)
+      if priority_position = p.children.maximum("#{IssuePriority.table_name}.position", :include => :priority
+ , :joins => "LEFT JOIN #{IssueStatus.table_name} ON #{IssueStatus.table_name}.id = #{Issue.table_name}.status_
id", :conditions => ["#{IssueStatus.table_name}.is_closed = ?", false])
         p.priority = IssuePriority.find_by_position(priority_position)
       end
     end
   end
end
```

If all childs are closed the last priority given to the parent survives. You may add a patch which disables the roll-up completely if all childs are closed.

#32 - 2012-03-14 01:54 - Jason Palmer

Having the subtasks take over the parent task is a complete pain. We want to use them (and are trying), but the current implementation is not workable for us. I can see the logic that if a subtask has a high priority, then the priority of the parent task must elevate for visibility, and that if a subtask has a date off in the future that the due date on the parent task has to be at least this far out.

The parent issue start/end dates, or priority should not be controlled by CLOSED subtasks - we have some work which is maintained in the parent issue as well as key areas of work extracted to subtasks, and when these subtasks are closed the parent task is left in limbo unless you edit each of the subtasks and remove the priority and dates. Ideally we should be able to set the due date/priority on the parent task at least as high as the highest open subtask, and if a subtask is added with a date that is beyond the parent task it should offer the choice of pulling back the due date on the subtask or extending the parent task.

#33 - 2012-05-15 07:56 - pamela stevens

That was something very informative for me, the information that you have provided has added a lot to my knowledge about the subject. Thank you

[website design agency](#)

#34 - 2012-05-22 13:30 - Jacq Jacq

+1

The way subtasks change the parent task is not practical.

#35 - 2012-06-08 19:07 - Josef Grahn

This has become one of my top grievances with Redmine. It makes a very useful feature (sub-tasks) almost practically unusable with our workflow, and almost any other workflow I can think of actually. As soon as you have added a sub-task you lose the ability to control the priority of the parent, which is where you really want to have the control, in a sort of supervisory role.

Suppose we have the following task, for example:

- Fix boat
 - Plug leaking hole (very urgent)
 - Lubricate squeaky hinge (not so important)

Both sub-tasks are logically part of fixing the boat. Plugging the hole is very urgent, however making sure the boat is fixed in its entirety is not (especially once the hole is plugged! #5880). I can see how one can make the argument that drawing attention to the "fix boat" issue is good if there are important (open!) sub-tasks, but since each sub-task is already an issue in its own regard in Redmine there is really no need.

There is also a rather dangerous reverse problem. Suppose we have the critical "plug hole" issue and someone adds a sub-task to it:

- Plug leaking hole (our lives depend on it)
 - Even off the plug surface with sandpaper, once in hole (not so important) <-- *added later, but while still leaking*

Oops, suddenly it is **not important to fix the leak any more** (it has now inherited the low priority of the "finishing touches" task). Logically, it made sense to add it as a sub-task to the "plug hole" issue, at least enough that someone could do it accidentally, but the consequence is that we have now *altered* the main task in doing so.

(Incidentally, perhaps Redmine should have a voting system instead of all these +1s.)

#36 - 2012-06-08 19:39 - Eric Voisard

Thanks for providing such a colorful example. If only developers would read it and perceive how nonsensical current implementation is...

#37 - 2012-06-30 19:07 - Jerome Revillard

+1 please add a property which can enable this behaviour

#38 - 2012-07-04 09:23 - Prof. Dr. YoMan

+1

#39 - 2012-07-26 13:45 - Chris R.

+1

#40 - 2012-08-20 09:24 - Anonymous

+1

#41 - 2012-08-28 19:50 - Moritz Lungershausen

+1

#42 - 2012-08-28 23:44 - Terence Mill

Priority shall be most important/Strongest of the priority all subtasks.

#43 - 2012-08-29 01:25 - louie santos

-

#44 - 2012-08-29 18:44 - Josef Grahn

Terence Mill wrote:

Priority shall be most important/Strongest of the priority all subtasks.

I disagree for the reasons I have stated above.

It would be interesting to hear which workflows people are using in practical, real-world projects of non-trivial size where the current behaviour is desirable, and why.

Unfortunately, having subtasks overruling parent tasks makes the subtask feature essentially useless for us. Even a bit risky, in fact. Evidently, several other Redmine users are of similar opinion, so if the aim is to make Redmine as useful as possible to as many people as possible, I think at least a setting for this should be considered.

#45 - 2012-09-05 22:48 - K. Scott Tripp

+1

This is a must I think, and Josef's example shows exactly why.

From an ease of use perspective, it also makes sense. If I have a large task that several people need to work on different parts of, I really don't want to have to manually set the due date for every single one.

What I think would be really nice is to inherit the parent's properties (due date, priority) as a default set of constraints, but then allow further modification within those constraints. For example, a child task cannot be due *later* than the parent.

Here is an example:

Parent task 123 is to deliver a doohickey. However, that requires someone to design it and build it, which are tasks 124 and 125 respectively. The doohicky is due at the end of next week, but clearly the design subtask needs a due date earlier than the build sub task.

#46 - 2012-10-17 09:58 - Johannes Pfeifer

+1

I would also like this.

#47 - 2012-10-22 12:08 - Roger Müller

+1

#48 - 2013-01-17 15:19 - Erny Revilla

+1. I just installed 2.2-stable and found the same issue.

#49 - 2013-01-17 15:55 - Erny Revilla

- *File issue_no_parent_updates-2.2-stable.diff added*

Erny Revilla wrote:

+1. I just installed 2.2-stable and found the same issue.

Here I have a quick & dirty patch for 2.2.1 (stable). It avoids update of the parent issue and makes the fields editable again. Enjoy.

#50 - 2013-02-28 21:56 - Zbynek Drlik

+1

#51 - 2013-04-05 10:53 - Sebastien Gaspard

+1

#52 - 2013-04-15 09:48 - André Schloemp

+1

It should give a choice on issue level if the issue controls the subtasks, the subtasks controls the parent issue or no control all must be done manually. Perhaps a per project setting for a default value on new issues.

We have projects where issues are used as "master backlogs", they have an estimation and start/due date. If someone now creates a subtask all that information are lost.

Other projects working more with "small" issues. So we have to group that issues and now the parent issues should calculated from the subtasks.

We have also projects where this is mixed together.

I hope I could give enough feedback, that this is really needed for us.

#53 - 2013-04-19 02:01 - Sebastian Bertram

Hi I installed 2.3 3 Weeks ago. At the moment I am facing the same problem as mentioned here.

Can I use the patch which was posted under [#49](#) also vor the new version?

Is it possible to easily remove the patch, if it won't work?

#54 - 2013-04-19 09:03 - Marc Pasteur

Sebastian Bertram wrote:

Hi I installed 2.3 3 Weeks ago. At the moment I am facing the same problem as mentioned here.

Can I use the patch which was posted under [#49](#) also vor the new version?

Is it possible to easily remove the patch, if it won't work?

i've patched my 2.2.3 version with this patch without any problem, but can't say with a 2.3...

You can patch with a -b option to backup impacted files.

#55 - 2013-04-19 09:44 - Ernesto Revilla

Marc Pasteur wrote:

Sebastian Bertram wrote:

Hi I installed 2.3 3 Weeks ago. At the moment I am facing the same problem as mentioned here.
Can I use the patch which was posted under [#49](#) also vor the new version?
Is it possible to easily remove the patch, if it won't work?

i've patched my 2.2.3 version with this patch without any problem, but can't say with a 2.3...
You can patch with a -b option to backup impacted files.

You can also reverse patch: patch -R

Regards.

#56 - 2013-04-19 17:36 - Sebastian Bertram

Thanks for the replay. I also found this patch. It also sound very nice --> [#6687](#)

Is there another diskription howto patch redmine than this one? --> <http://www.redmine.org/projects/redmine/wiki/Patch>

Because some things aren't clear for me.
Is there already a patch file in redmine?
If not, where do I have to put the patch-programm that is mentioned?
An at last, where do I have to put the .diff file?

Sorry for the questions but I is my first time working with redmine.

#57 - 2013-04-19 22:17 - Sebastian Bertram

- File *issue_no_parent_updates-2.3-stable.diff* added

I managed to patch the file. First tried the patch form [#6687](#). Patching process went fine, but I hat a problem with an internal error 500 when I wanted to display an issue. So I loaded the old .rb again.

Now I wanted to patch the above .diff. But I says, that it cant find the issue.rb. I think it is because of a/app/models/issue.rb and b/app/models/issue.rb. In my installation files the issue.rb is in app/models/issue.rb.
So I edit the file. But now I am not sure if this is right or not. Maybe somebody could look if that what I have done is making sense.

Thanks and best regards

#58 - 2013-04-20 13:55 - Ernesto Revilla

Hi.

When you do a git diff the first item in the path is always "a" and "b". To apply git patches, you could use:

```
git apply patch.diff
```

But you could also use:

```
patch -p1 < patch.diff
```

The number after -p parameters specifies how many path items to remove from the paths given in the diff file.

Regards.

Sebastian Bertram wrote:

I managed to patch the file. First tried the patch form [#6687](#). Patching process went fine, but I hat a problem with an internal error 500 when I wanted to display an issue. So I loaded the old .rb again.

Now I wanted to patch the above .diff. But I says, that it cant find the issue.rb. I think it is because of a/app/models/issue.rb and b/app/models/issue.rb. In my installation files the issue.rb is in app/models/issue.rb.
So I edit the file. But now I am not sure if this is right or not. Maybe somebody could look if that what I have done is making sense.

Thanks and best regards

#59 - 2013-04-26 11:35 - Bernd Winter

Erny Revilla wrote:

Here I have a quick & dirty patch for 2.2.1 (stable). It avoids update of the parent issue and makes the fields editable again. Enjoy.

Hi, yesterday I tried to use that patch as guideline to enable the independent parent task priority again (so basically I only used the priority-relevant parts of your patch). It did have the effect that modifications among child tasks didn't affect the parent priority any more. However, it seems that I **could still not modify the priority in the parent task** itself. The field is editable, I can alter the value and save/update the task, but the priority will not be changed.

After quite some code reading (but not being fluent in Ruby), I could not find an obvious reason for the priority update being ignored.

Maybe you have an idea where to look more thoroughly?

#60 - 2013-04-29 09:44 - Ernesto Revilla

Bernd Winter wrote:

Erny Revilla wrote:

Here I have a quick & dirty patch for 2.2.1 (stable). It avoids update of the parent issue and makes the fields editable again. Enjoy.

Hi, yesterday I tried to use that patch as guideline to enable the independent parent task priority again (so basically I only used the priority-relevant parts of your patch). It did have the effect that modifications among child tasks didn't affect the parent priority any more. However, it seems that I **could still not modify the priority in the parent task** itself. The field is editable, I can alter the value and save/update the task, but the priority will not be changed.

Hi there. I can't remember well, but I think the parent data is recalculated from children whenever it is updated. I.e., your changes will be overwritten.

Regards

After quite some code reading (but not being fluent in Ruby), I could not find an obvious reason for the priority update being ignored.

Maybe you have an idea where to look more thoroughly?

#61 - 2013-05-29 15:51 - Alexey Ivanov

Bernd Winter wrote:

Hi, yesterday I tried to use that patch as guideline to enable the independent parent task priority again (so basically I only used the priority-relevant parts of your patch). It did have the effect that modifications among child tasks didn't affect the parent priority any more. However, it seems that I **could still not modify the priority in the parent task** itself. The field is editable, I can alter the value and save/update the task, but the priority will not be changed.

After quite some code reading (but not being fluent in Ruby), I could not find an obvious reason for the priority update being ignored.

Maybe you have an idea where to look more thoroughly?

Hi!

I've solved it for 2.3!

After patch "issue_no_parent_updates-2.3-stable.diff", just find these lines (in app/models/issue.rb:436)

```
unless leaf?  
  attrs.reject! {|k,v| %w(priority_id done_ratio start_date due_date estimated_hours).include?(k) }  
end
```

and replace it for:

```
unless leaf?  
  attrs.reject! {|k,v| %w(done_ratio).include?(k) }  
end
```

#62 - 2013-06-27 13:33 - not available

+1 Completely illogical behavior.

issue_no_parent_updates-2.3-stable.diff + <http://www.redmine.org/issues/5490#note-61> = issue_no_parent_updates-2.4-stable.diff needed.

#63 - 2013-10-07 15:27 - Deoren Moor

+1 to have the behavior swapped. There is probably a good reason to have the child tasks control the parent task, but I'm not seeing it.

#64 - 2014-01-30 16:30 - Bernd Worsch

Hi there! I stumbled upon the estimates-due-subtasks problem and as this page seems stalled and somewhat long went for the forum: <http://www.redmine.org/boards/1/topics/41138>. Hopefully the ideas presented there help to move the topic along.

#65 - 2014-01-31 14:25 - Peter Englmaier

Please fix this issue. Maybe the following strategy might help solving it:

- Make parent ticket have it's own 'estimated_hours' value.
- State TOTAL estimated_hours value behind the estimated_hours.
- Make parent have it's own start_date / due_date and have a red flag/warning behind it subtask begins/starts before that date.

#66 - 2014-02-19 23:37 - John Pisani

+1

#67 - 2014-02-22 11:16 - Toshi MARUYAMA

- Related to Defect #16092: Parent/subtask: calculation of estimated hours added

#68 - 2014-04-22 18:56 - Ismael Barros²

+1

Pretty please with sugar on it.

#69 - 2014-06-24 17:05 - Fran Fabrizio

+1

Definitely need to at least allow the option for the parent fields to remain editable and independent of the subtasks. Reasons well-documented by others before me.

#70 - 2014-06-25 16:53 - Hackkyy Hack

- File *SubtaskOverview.png* added

+1

Current version also unusable for me because of start and due date.

Child issues should not have start or due date outside the interval defined on the parent, but that interval can be later modified (not beyond first start date and last due date of subtasks).

#71 - 2014-07-18 01:53 - James H

+1

Josef's Example

I say it would be best if you have two fields per issue when issues have sub-tasks.

One read-only field that displays the sub-task's properties (as it does now) and an input field next to the first field where you can either override or add dates, hours, etc.

This gives us flexibility to use the current style of sub-task handling or to input our own needs.

Sub-task properties should not always override parent task properties.

#72 - 2014-10-23 23:11 - Joel SCHAAL

+1

#73 - 2014-12-03 13:13 - Sergey Startsev

+1

Very-very needed.

Idealy it will be as extention to option of autocalc/manual updating of done percentage by task state.

It's awful, while I can override percentage, and canot override begin and end date, priority, and even estimated horus.

#74 - 2014-12-03 15:06 - Nick Maximenko

+1

it will be good if we can set up for tasks:

- independent (completely independent parent and child tasks, there is just a relation between them)
- derived from children (the way it is now)

- restricted children

#75 - 2014-12-17 00:35 - Moritz Koehler

- File Diff Calender for.png added

+1

I had to change two additional lines in order to get my calendar buttons back in app/views/issues/_attributes.html.erb

Ruby 2.6

issue_no_parent_updates-2.3-stable.diff but manually not with patch and the change form <http://www.redmine.org/issues/5490#note-61>

seems to work

[_600](#)

#76 - 2015-01-08 16:48 - Jon Bruckman

+1

Parent tasks should accumulate subtask estimated time, not be overridden by them. If I create a feature that I think will take 10 hours to complete, and implementing this feature creates an unforeseen bug, I want the bug to be a subtask that needs to be fixed as part of implementing the feature. I do not want the 0.5h expected time on the bug to completely eradicate the 10h expected time on the feature... Also, the start/end times don't work in this context, either. The bug would start later than the feature, and possibly end at a different time as well.

#77 - 2015-01-19 13:39 - Marc D.

+1

#78 - 2015-01-31 20:01 - Alexander Kupchenko

+1 it is survival need

#79 - 2015-02-09 08:48 - Ismael Barros²

+11, current behavior (subtasks taking control of parent task) is a huge deal breaker and is wreaking havoc when we try to use Redmine to organize work

#80 - 2015-02-16 20:19 - WDS D

+1, the following suggestion would be perfect. I would also like to add that these options should be for each property (start date, due date, spent time, priority ...) because even though I don't want start and due date of subtasks to affect parent task, I would still like the subtasks' spent time added to the parent's.

Nick Maximenko wrote:

+1

it will be good if we can set up for tasks:

- independent (completely independent parent and child tasks, there is just a relation between them)
- derived from children (the way it is now)
- restricted children

#81 - 2015-02-19 17:44 - Lucile Quirion

+1

It is also a important feature for our usage of Redmine.

I'll start working on this.

#82 - 2015-02-23 20:02 - Lucile Quirion

- File 0001-settings-configurable-issue-independent-time-trackin.patch added

The patch is ready (it applies onto v3.0.0 [r14043](#))(its formatted with git)

I've named the setting 'issue_independent_time_tracking'.

Its available via the Administration > Settings > Issue tracking > "Estimate and track time for parent issue independently from children issues"

Parent attributes 'priority', 'start_date', 'due_date', 'estimated_hours' and 'done_ratio' will not be overridden by subtask.
Child attributes 'start_date', 'due_date' will not be updated when the parent task is rescheduled.

I would like to have feedback on this patch.

Thanks

#83 - 2015-03-09 14:25 - Mötz Jensen

Lucile Quirion wrote:

The patch is ready (it applies onto v3.0.0 [r14043](#))(its formatted with git)

...

I would like to have feedback on this patch.

Thanks

I would like to know how to apply this patch on an installed and running redmine server?
Should I make a clone of the repo onto my local computer, apply your patch and see what files that are changed by the patch? Then copy the changed files into my redmine installation, overwrite the files already in place?

Thanks

#84 - 2015-04-17 10:32 - Michael Gielda

[Lucile Laine](#) Quirion

awesome! will this work with Redmine 2.6.0/2.6.2? I have been looking for that for a long time...

#85 - 2015-05-05 10:31 - Systems Administration

Lucile Quirion wrote:

The patch is ready (it applies onto v3.0.0 [r14043](#))(its formatted with git)

I would like to have feedback on this patch.

+1 for the issue.

Unfortunately I am running on 2.5.1 so I'm not sure your patch would apply cleanly...

#86 - 2015-05-07 13:15 - Kevin Palm

+1 (fields of parent-issue should not be restricted by closed/rejected child-issues)

#87 - 2015-05-23 09:32 - Jean-Philippe Lang

- *Related to Defect #19914: Conflicting Fields Permission X Workflow X Subtask added*

#88 - 2015-05-26 03:51 - Go MAEDA

- *Target version changed from Unplanned backlogs to 3.1.0*

#89 - 2015-05-26 10:53 - Sebastian Paluch

Lucile Quirion wrote:

The patch is ready (it applies onto v3.0.0 [r14043](#))(its formatted with git)

I've named the setting 'issue_independent_time_tracking'.

Its available via the Administration > Settings > Issue tracking > "Estimate and track time for parent issue independently from children issues"

Parent attributes 'priority', 'start_date', 'due_date', 'estimated_hours' and 'done_ratio' will not be overridden by subtask.

Child attributes 'start_date', 'due_date' will not be updated when the parent task is rescheduled.

I would like to have feedback on this patch.

Thanks

I don't get this...

If you eliminate all the dependencies, as suggested, this leads to simple "Related" relation and breaks entire idea of subtasks.

If you have a need for such loose relation why are you trying to use (and change) subtasks instead just using relations?

#90 - 2015-05-27 19:09 - James H

Sebastian Paluch wrote:

Lucile Quirion wrote:

The patch is ready (it applies onto v3.0.0 [r14043](#))(its formatted with git)

I've named the setting 'issue_independent_time_tracking'.

Its available via the Administration > Settings > Issue tracking > "Estimate and track time for parent issue independently from children issues"

Parent attributes 'priority', 'start_date', 'due_date', 'estimated_hours' and 'done_ratio' will not be overridden by subtask. Child attributes 'start_date', 'due_date' will not be updated when the parent task is rescheduled.

I would like to have feedback on this patch.

Thanks

I don't get this...

If you eliminate all the dependencies, as suggested, this leads to simple "Related" relation and breaks entire idea of subtasks.

If you have a need for such loose relation why are you trying to use (and change) subtasks instead just using relations?

People have different views on what a parent/child issue is and how they should act in relation to each other.

Its one thing for a parent to be made up of **ONLY their child issues/tasks** and another thing for a parent to be made up of **itself AND their child issues/tasks**.

Do parents have an identity as well? Or are they just a "Folder" for subtasks?

- At the moment, they act like "Folder"s for subtasks. This essentially makes the parent issue have no function or substance other than to be a container/label for its subtasks.
- On the other hand, my natural understanding of how parent/child works was that parents have their own task or issue, and any subtasks that are added onto it are **additional** tasks (get it? "added" onto the issue are "additional" things to the issue, not the **ONLY** things).

The natural workflow of Redmine also perpetuates one way more than the other because when you create a parent/child relationship, (usually) you first create the parent and then create "subtasks" to that parent. For me particularly, I tend to create parent issues (without knowing they are parents first) and use them. As I am using the first issue, I find that I need subtask issues to track specific parts of the first issue. I create a subtask and apply dates and whatever that are specific to **ONLY** that subtask. Now my parent issue's data (dates and whatnot) are being overwritten by the subtask and the subtask data pertains **ONLY** to its specific scope. This is unnatural for the user.

Its important to be able to identify and understand the different viewpoints of the user, especially on a total user experience-level.

Additionally, if they had a Issue Relation called "Subtask" or something similar, I might use it instead of creating subtasks and having parent/child issues because it doesn't work the way I naturally think it would work.

Otherwise, it isn't really the same to label something as a "related issue" vs a "subtask" (or whatever you call it).

#91 - 2015-05-30 10:03 - Jean-Philippe Lang

- *Tracker changed from Defect to Feature*

- *Subject changed from subtask priority/start date/due date to Option for independent subtask priority/start date/due date*

- *Status changed from New to Closed*

- *Assignee set to Jean-Philippe Lang*

- *Resolution set to Fixed*

Nick Maximenko wrote:

it will be good if we can set up for tasks:

- independent (completely independent parent and child tasks, there is just a relation between them)
- derived from children (the way it is now)
- restricted children

You can now choose between the 2 first options how start/due dates, priority and done ratio are handled. The last one is way too vague, please open a separate feature request with the exact rules that are needed.

#92 - 2015-07-11 11:11 - Mischa The Evil

- Subject changed from Option for independent subtask priority/start date/due date to Option for independent subtask priority/start date/due date/done ratio

#93 - 2015-07-28 12:31 - Petr Pospisil

Are you sure to delete user's data without asking?

```
_def self.up
  # Clears estimated hours on parent issues
  Issue.where("rgt > lft + 1 AND estimated_hours > 0").update_all :estimated_hours => nil
end_
```

#94 - 2015-07-28 12:37 - Jean-Philippe Lang

Petr Pospisil wrote:

Are you sure to delete user's data without asking?

Yes, this was not data entered by users but calculated by Redmine (sum of leaves estimates)

#95 - 2015-09-11 20:51 - Julius Friedrich

James H wrote:

The natural workflow of Redmine also perpetuates one way more than the other because when you create a parent/child relationship, (usually) you first create the parent and then create "subtasks" to that parent. For me particularly, I tend to create parent issues (without knowing they are parents first) and use them. As I am using the first issue, I find that I need subtask issues to track specific parts of the first issue. I create a subtask and apply dates and whatever that are specific to ONLY that subtask. Now my parent issue's data (dates and whatnot) are being overwritten by the subtask and the subtask data pertains ONLY to its specific scope. This is unnatural for the user. Its important to be able to identify and understand the different viewpoints of the user, especially on a total user experience-level.

Same for me.

Jean-Philippe Lang wrote:

You can now choose between the 2 first options how start/due dates, priority and done ratio are handled. The last one is way too vague, please open a separate feature request with the exact rules that are needed.

We're talking about Redmine core functionality, I wish such issues would be met with a higher priority and be thoroughly discussed and evaluated before getting (somewhat) fixed. Next fix/feature will probably be several years ahead.

#96 - 2015-10-13 16:42 - Robert Chady

I know this ticket is closed, but wanted to interject one point in to it anyhow. First off, I am very happy to see the ability to change this behavior being added to Redmine core. I used to hack this out of my version because turning an established ticket in to a parent would blindly clobber data that was needed, breaking many things.

With that said, I see a need for having both pieces of information. Towards that, would it be an option to add a couple new fields to parent tasks that would indicate the summed information from the sub-tasks? For example, Estimated Hours and Spent Time are very useful to have summed up.

#97 - 2015-10-14 12:08 - Ismael Barros²

- File Screenshot from 2015-10-14 11-59-18.png added

- File Screenshot from 2015-10-14 11-59-28.png added

I've just updated from Redmine 3.0 to Redmine 3.1 just to get this fix, but it doesn't seem to be working for me.

I've configured "Parent task attributes > Priority" to "Independent of subtasks" (see [Screenshot from 2015-10-14 11-59-18.png](#)), but when I want to change the priority of a parent, the field is still grayed out (see [Screenshot from 2015-10-14 11-59-28.png](#)). Am I missing something?

Versions and stuff: [unfoldunfold](#)

```
Environment:
  Redmine version      3.1.1.stable
  Ruby version         1.9.3-p0 (2011-10-30) [x86_64-linux]
  Rails version        4.2.4
  Environment          production
  Database adapter     Mysql2
```

SCM:

```
Mercurial          2.0.2
Git                1.7.9.5
Filesystem
Redmine plugins:
clipboard_image_paste 1.10
redmine_embedded_video 0.0.3.1
redmine_inline_note_images_plugin 0.0.1
redmine_ldap_sync    2.0.7.devel.g202f7c74b9
redmine_lightbox     0.0.1
redmine_my_page_queries 2.1.6
redmine_zenedit      0.0.2
sidebar_hide        0.0.7
time_logger         0.5.1
```

#98 - 2015-10-14 16:51 - Deoren Moor

Ismael Barros² wrote:

I've just updated from Redmine 3.0 to Redmine 3.1 just to get this fix, but it doesn't seem to be working for me.

Confirmed here as well (same settings). I suspect that the devs would prefer to have a new ticket opened for this regression so it can be tracked/fixed in a new release. Do you want to do that or should I?

#99 - 2015-10-14 17:31 - Ismael Barros²

Done: [#20992](#)

Thanks :)

#100 - 2016-12-05 15:04 - Janeks Kamerovskis

Is this available for:

```
Environment:
Redmine version 3.2.1.stable
Ruby version 2.3.1-p112 (2016-04-26) [x86_64-linux-gnu]
Rails version 4.2.7.1
Environment production
Database adapter PostgreSQL
SCM:
Git 2.9.3
Filesystem
Redmine plugins:
event_notifications 3.2.1
planner 0.5
redmine_better_gantt_chart 0.9.0
redmine_checklists 3.1.5
redmine_dashboard 2.7.1
redmine_mail_reminder 3.0.0.0001
redmine_mentions 0.0.1
redmine_percent_done 1.0.0
redmine_spent_time_in_issue_description 2.8.0
redmine_tags 3.1.1
redmine_work_time 0.3.3
timelog_timer 2.0.0
```

?

#101 - 2017-01-03 08:25 - Ismael Barros²

Janeks Kamerovskis wrote:

Is this available for:

[...]

?

According to [#20992-2](#), it was set to be released in 3.1.2, so it *should* be available in your 3.2.1

#102 - 2021-10-19 16:53 - Holger Mößinger

FYI: I am working on the "restricted children" option mentioned in [#5490-91](#) in a new issue [#34609](#). Input is appreciated.

Files

defang_children.diff	1.73 KB	2010-06-26	Michael Hellein
issue5490.patch	2.52 KB	2012-01-04	Sridhar P
issue_no_parent_updates-2.2-stable.diff	5.03 KB	2013-01-17	Erny Revilla
issue_no_parent_updates-2.3-stable.diff	5.01 KB	2013-04-19	Sebastian Bertram
SubtaskOverview.png	20.8 KB	2014-06-25	Hackkyy Hack
Diff Calender for.png	64 KB	2014-12-16	Moritz Koehler
0001-settings-configurable-issue-independent-time-trackin.patch	20.9 KB	2015-02-23	Lucile Quirion
Screenshot from 2015-10-14 11-59-28.png	7.87 KB	2015-10-14	Ismael Barros ²
Screenshot from 2015-10-14 11-59-18.png	12.2 KB	2015-10-14	Ismael Barros ²