

## Redmine - Patch #40798

### Optimize Version model

2024-06-05 17:28 - Go MAEDA

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Go MAEDA	<b>% Done:</b>	0%
<b>Category:</b>	Performance	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	6.0.0		
<b>Description</b>			
<p>The attached patch improves the performance of several methods in the Version model such as Version#issues_count, FixedIssuesExtension#completed_percent, FixedIssuesExtension#closed_percent, and FixedIssuesExtension#issues_progress.</p> <p>These methods use count method to get the total number of fixed issues in the version. It performs an SQL count query every time it is called. This patch replaces count with open_count + closed_count in order to use cached values instead of performing SQL count queries.</p> <p>The change has an effect on Gantt as well as Roadmap.</p>			

#### Associated revisions

##### Revision 22880 - 2024-06-17 08:55 - Go MAEDA

Optimize Version model by reducing redundant SQL count queries (#40798).

Patch by Go MAEDA (@maeda).

##### Revision 22881 - 2024-06-17 08:57 - Go MAEDA

Optimize FixedIssuesExtension#closed\_percent by bypassing issues\_progress when all issues are closed (#40798).

Patch by Go MAEDA (@maeda).

##### Revision 22882 - 2024-06-17 09:10 - Go MAEDA

Optimize Version#behind\_schedule? by avoiding the call to completed\_percent when due\_date or start\_date is nil (#40798).

Patch by Go MAEDA (@maeda).

#### History

##### #1 - 2024-06-06 01:45 - Go MAEDA

The following are SQL queries triggered by Version#behind\_schedule?.

Before:

```
irb(main):001> Version.find(2).behind_schedule?  
  Version Load (0.1ms)  SELECT "versions".* FROM "versions" WHERE "versions"."id" = ? LIMIT ?  [{"id", 2}, [{"LIMIT", 1}]]  
  Issue Count (0.1ms)  SELECT COUNT(*) FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]]  
  Issue Count (0.1ms)  SELECT COUNT(*) AS "count_all", "issues"."status_id" AS "issues_status_id" FROM "issues" WHERE "issues"."fixed_version_id" = ? GROUP BY "issues"."status_id" [{"fixed_version_id", 2}]]  
  IssueStatus Load (0.1ms)  SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" IN (?, ?) [{"id", 2}, [{"id", 5}]]  
  Issue Count (0.1ms)  SELECT COUNT(*) FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]]  
  Issue Load (0.2ms)  SELECT "issues".* FROM "issues" INNER JOIN "issue_statuses" ON "issue_statuses"."id" = "issues"."status_id" WHERE "issues"."fixed_version_id" = ? AND "issue_statuses"."is_closed" = ? [{"fixed_version_id", 2}, [{"is_closed", 1}]]  
  Issue Load (0.1ms)  SELECT "issues".* FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]]  
  IssueStatus Load (0.1ms)  SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" = ? LIMIT ? [{"id", 5}, [{"LIMIT", 1}]]  
  Issue Count (0.0ms)  SELECT COUNT(*) FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]]
```

```

Issue Count (0.0ms) SELECT COUNT(*) FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]
Issue Load (0.0ms) SELECT "issues".* FROM "issues" INNER JOIN "issue_statuses" ON "issue_statuses"."id" = "issues"."status_id" WHERE "issues"."fixed_version_id" = ? AND "issue_statuses"."is_closed" = ? [{"fixed_version_id", 2}, {"is_closed", 0}]
IssueStatus Load (0.0ms) SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" = ? LIMIT ? [{"id", 2}, {"LIMIT", 1}]
Setting Load (0.0ms) SELECT "settings".* FROM "settings" WHERE "settings"."name" = ? ORDER BY "settings"."id" DESC LIMIT ? [{"name", "issue_done_ratio"}, {"LIMIT", 1}]
Issue Count (0.0ms) SELECT COUNT(*) FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]
Issue Minimum (0.0ms) SELECT MIN("issues"."start_date") FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]
Issue Count (0.0ms) SELECT COUNT(*) FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]
AnonymousUser Load (0.1ms) SELECT "users".* FROM "users" WHERE "users"."type" = ? AND "users"."lastname" = ? LIMIT ? [{"type", "AnonymousUser"}, {"lastname", "Anonymous"}, {"LIMIT", 1}]
=> false

```

After:

```

irb(main):001> Version.find(2).behind_schedule?
Version Load (0.1ms) SELECT "versions".* FROM "versions" WHERE "versions"."id" = ? LIMIT ? [{"id", 2}, {"LIMIT", 1}]
Issue Count (0.1ms) SELECT COUNT(*) AS "count_all", "issues"."status_id" AS "issues_status_id" FROM "issues" WHERE "issues"."fixed_version_id" = ? GROUP BY "issues"."status_id" [{"fixed_version_id", 2}]
IssueStatus Load (0.1ms) SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" IN (?, ?) [{"id", 2}, {"id", 5}]
Issue Load (0.1ms) SELECT "issues".* FROM "issues" INNER JOIN "issue_statuses" ON "issue_statuses"."id" = "issues"."status_id" WHERE "issues"."fixed_version_id" = ? AND "issue_statuses"."is_closed" = ? [{"fixed_version_id", 2}, {"is_closed", 1}]
Issue Load (0.5ms) SELECT "issues".* FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]
IssueStatus Load (0.0ms) SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" = ? LIMIT ? [{"id", 5}, {"LIMIT", 1}]
Issue Load (0.0ms) SELECT "issues".* FROM "issues" INNER JOIN "issue_statuses" ON "issue_statuses"."id" = "issues"."status_id" WHERE "issues"."fixed_version_id" = ? AND "issue_statuses"."is_closed" = ? [{"fixed_version_id", 2}, {"is_closed", 0}]
IssueStatus Load (0.0ms) SELECT "issue_statuses".* FROM "issue_statuses" WHERE "issue_statuses"."id" = ? LIMIT ? [{"id", 2}, {"LIMIT", 1}]
Setting Load (0.0ms) SELECT "settings".* FROM "settings" WHERE "settings"."name" = ? ORDER BY "settings"."id" DESC LIMIT ? [{"name", "issue_done_ratio"}, {"LIMIT", 1}]
Issue Minimum (0.1ms) SELECT MIN("issues"."start_date") FROM "issues" WHERE "issues"."fixed_version_id" = ? [{"fixed_version_id", 2}]
AnonymousUser Load (0.1ms) SELECT "users".* FROM "users" WHERE "users"."type" = ? AND "users"."lastname" = ? LIMIT ? [{"type", "AnonymousUser"}, {"lastname", "Anonymous"}, {"LIMIT", 1}]
=> false

```

## #2 - 2024-06-13 10:55 - Go MAEDA

- File 0001-Improve-performance-of-Version-model-by-reducing-red.patch added
- File 0002-Optimize-FixedIssuesExtension-closed\_percent-by-bypa.patch added
- File 0003-Optimize-Version-behind\_schedule-by-avoiding-the-cal.patch added

Slightly improved the patches.

## #3 - 2024-06-13 10:56 - Go MAEDA

- File deleted (0001-Improve-performance-of-Version-model-by-reducing-red.patch)

## #4 - 2024-06-13 10:56 - Go MAEDA

- File deleted (0002-Optimize-FixedIssuesExtension-closed\_percent-by-bypa.patch)

## #5 - 2024-06-15 09:44 - Go MAEDA

- Target version changed from Candidate for next major release to 6.0.0

Setting the target version to 6.0.0.

## #6 - 2024-06-17 09:12 - Go MAEDA

- Subject changed from *Reduce redundant SQL count queries in Version model* to *Optimize Version model*
- Status changed from *New* to *Closed*
- Assignee set to *Go MAEDA*

Committed the patches in [r22880](#), [r22881](#), and [r22882](#).

### Files

---

0001-Improve-performance-of-Version-model-by-reducing-red.patch	1.88 KB	2024-06-13	Go MAEDA
0002-Optimize-FixedIssuesExtension-closed_percent-by-bypa.patch	754 Bytes	2024-06-13	Go MAEDA
0003-Optimize-Version-behind_schedule-by-avoiding-the-cal.patch	1.27 KB	2024-06-13	Go MAEDA