

Redmine - Defect #32971

New line between list items break a list

2020-02-08 15:03 - Ji-Hyeon Gim

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Go MAEDA	% Done:	0%
Category:	Text formatting	Estimated time:	0.00 hour
Target version:	4.1.1	Affected version:	4.1.0
Resolution:	Fixed		
Description			
New line between list items break a list when I try to write ordered/unordered list with textile.			
- Example			
* list1			
* list2			
* list3			
# list4			
# list5			
# list6			
- In 3.2.1			
3.2.1.png			
- In 4.1.0			
4.1.0.png			
Related issues:			
Related to Redmine - Defect # 29756: \f or \v character in Textile markup may...			Closed

Associated revisions

Revision 19536 - 2020-02-29 05:21 - Go MAEDA

Newline between list items break a list (#32971).

Patch by Yuichi HARADA.

Revision 19542 - 2020-02-29 12:18 - Go MAEDA

Fix a RuboCop offence Style/NegatedWhile (#32971).

Revision 19545 - 2020-02-29 15:35 - Go MAEDA

Merged r19536 and r19542 from trunk to 4.1-stable (#32971).

History

#1 - 2020-02-17 07:59 - Yuichi HARADA

- File 32971-fixed.patch added

I think the behavior has probably changed by applying r17603 ([Defect !#29756: \f or \v character in Textile markup may cause RegexpError exception](#)).

https://www.redmine.org/projects/redmine/repository/revisions/17603/diff/trunk/lib/redmine/wiki_formatting/textile/redcloth3.rb

It could be resolved with the following patch.

```
diff --git a/lib/redmine/wiki_formatting/textile/redcloth3.rb b/lib/redmine/wiki_formatting/textile/redcloth3.rb
index 80e0a3626..ff4687f7e 100644
--- a/lib/redmine/wiki_formatting/textile/redcloth3.rb
+++ b/lib/redmine/wiki_formatting/textile/redcloth3.rb
@@ -1020,10 +1020,13 @@ class RedCloth3 < String
  end

  def flush_left( text )
-   indt = 0
-   if text =~ /^ /
+   if /(?![\r\n\t ])[[:cntrl:]]/.match?(text)
+     text.gsub!(?![\r\n\t ])[[:cntrl:]]/, " )
+   end
+   if /^ /.match?(text) && text.length > 1
+     indt = 0
+     unless text.empty?
-     indt += 1 while text !~ /^ #{indt}[^ ]/
+     indt += 1 while text !~ /^ #{indt}\S/
+   end
+   if indt.nonzero?
+     text.gsub!( /^ #{indt}$/, " )
```

#2 - 2020-02-18 05:41 - Ji-Hyeon Gim

Thanks for your help Yuichi! :)

This patch works gracefully well!

Thank you again for patching it :)

#3 - 2020-02-18 13:18 - Go MAEDA

- Related to Defect #29756: \f or \v character in Textile markup may cause RegexpError exception added

#4 - 2020-02-18 13:19 - Go MAEDA

- Target version set to Candidate for next minor release

#5 - 2020-02-19 03:47 - Yuichi HARADA

- File 32971-fixed-v2.patch added

Yuichi HARADA wrote:

I think the behavior has probably changed by applying r17603 ([Defect !#29756: \f or \v character in Textile markup may cause RegexpError exception](#)).

https://www.redmine.org/projects/redmine/repository/revisions/17603/diff/trunk/lib/redmine/wiki_formatting/textile/redcloth3.rb

It could be resolved with the following patch.

[...]

Sorry, I found that String#lstrip! could do the same.

https://www.rubydoc.info/stdlib/core/String#lstrip!-instance_method

```
diff --git a/lib/redmine/wiki_formatting/textile/redcloth3.rb b/lib/redmine/wiki_formatting/textile/redcloth3.rb
index 80e0a3626..815f860e6 100644
--- a/lib/redmine/wiki_formatting/textile/redcloth3.rb
+++ b/lib/redmine/wiki_formatting/textile/redcloth3.rb
@@ -1020,15 +1020,7 @@ class RedCloth3 < String
   end

   def flush_left( text )
-     indt = 0
-     if text =~ /^ /
-       unless text.empty?
-         indt += 1 while text !~ /^ #{indt}[^ ]/
-       end
-       if indt.nonzero?
-         text.gsub!( /^ #{indt}/, " )
-       end
-     end
+     text&.lstrip!
   end

   def footnote_ref( text )
```

#6 - 2020-02-19 05:18 - Yuichi HARADA

Yuichi HARADA wrote:

Yuichi HARADA wrote:

I think the behavior has probably changed by applying r17603 ([Defect !#29756: \f or \v character in Textile markup may cause RegexpError exception](#)).

https://www.redmine.org/projects/redmine/repository/revisions/17603/diff/trunk/lib/redmine/wiki_formatting/textile/redcloth3.rb

It could be resolved with the following patch.

[...]

Sorry, I found that `String#strip!` could do the same.

https://www.rubydoc.info/stdlib/core/String#strip!-instance_method

[...]

It did not work well with attachment:32971-fixed-v2.patch . The same problem occurred as in the previous situation.
Please delete attachment:32971-fixed-v2.patch .

#7 - 2020-02-19 05:42 - Go MAEDA

- File deleted (32971-fixed-v2.patch)

#8 - 2020-02-19 07:17 - Yuichi HARADA

- File 32971-fixed-v3.patch added

I fixed the attachment:32971-fixed.patch because sometimes the `flush_left` method did not end.

```
diff --git a/lib/redmine/wiki_formatting/textile/redcloth3.rb b/lib/redmine/wiki_formatting/textile/redcloth3.rb
index 80e0a3626..c9bdd0aeb 100644
--- a/lib/redmine/wiki_formatting/textile/redcloth3.rb
+++ b/lib/redmine/wiki_formatting/textile/redcloth3.rb
@@ -1020,11 +1020,12 @@ class RedCloth3 < String
  end

  def flush_left( text )
-   indt = 0
-   if text =~ /^ /
-     unless text.empty?
-       indt += 1 while text !~ /^ #{indt}[^ ]/
-     end
+   if /^(?![\r\n\t ])[[:cntrl:]]/.match?(text)
+     text.gsub!(/^(?![\r\n\t ])[[:cntrl:]]/, " )
+   end
+   if /^ +S/.match?(text)
+     indt = 0
+     indt += 1 while !/^ #{indt}\S/.match?(text)
+     if indt.nonzero?
+       text.gsub!(/^ #{indt}/, " )
+     end
  end
```

#9 - 2020-02-19 11:40 - Kevin Fischer

How about adding a unit test to prevent future regressions of this conversion?

#10 - 2020-02-20 04:10 - Yuichi HARADA

Ji-Hyeon Gim wrote:

|

2021-10-26

New line between list items break a list when I try to write ordered/unordered list with textile.

- Example

* list1

* list2

* list3

list4

list5

list6

- In 3.2.1

3.2.1.png

I noticed when looking at source:trunk/test/unit/lib/redmine/wiki_formatting/textile_formatter_test.rb#L148 that this nested list is probably wrong for Textile.

```
def test_nested_lists
  raw = <<~RAW
  # Item 1
  # Item 2
  ** Item 2a
  ** Item 2b
  # Item 3
  ** Item 3a
  RAW
  expected = <<~EXPECTED
  <ol>
  <li>Item 1</li>
  <li>Item 2
  <ul>
  <li>Item 2a</li>
  <li>Item 2b</li>
  </ul>
  </li>
  <li>Item 3
  <ul>
  <li>Item 3a</li>
  </ul>
  </li>
  </ol>
  EXPECTED
  assert_equal expected.gsub(%r{\s+}, ""), to_html(raw).gsub(%r{\s+}, "")
end
```

#11 - 2020-02-20 09:02 - Yuichi HARADA

- File 32971-fixed-v4.patch added

Kevin Fischer wrote:

| *How about adding a unit test to prevent future regressions of this conversion?*

Thank you for pointing it out.

I have added the following test to attachment:32971-fixed-v3.patch .

```
diff --git a/test/unit/lib/redmine/wiki_formatting/textile_formatter_test.rb b/test/unit/lib/redmine/wiki_formatting/textile_formatter_test.rb
index 19128524e..2358ded58 100644
--- a/test/unit/lib/redmine/wiki_formatting/textile_formatter_test.rb
+++ b/test/unit/lib/redmine/wiki_formatting/textile_formatter_test.rb
@@ -171,6 +171,24 @@ class Redmine::WikiFormatting::TextileFormatterTest < ActionView::TestCase
  </ol>
  EXPECTED
  assert_equal expected.gsub(%r{\s+}, ""), to_html(raw).gsub(%r{\s+}, "")
+
+  raw = <<~RAW
+  * Item-1
+
+  * Item-1a
+  * Item-1b
+  RAW
+  expected = <<~EXPECTED
+  <ul>
+  <li>Item-1
+  <ul>
+  <li>Item-1a</li>
+  <li>Item-1b</li>
+  </ul>
+  </li>
+  </ul>
+  EXPECTED
+  assert_equal expected.gsub(%r{\s+}, ""), to_html(raw).gsub(%r{\s+}, "")
  end

  def test_escaping
```

#12 - 2020-02-25 03:22 - Kevin Fischer

Tried it out. The test passed, so looks good to me!

#13 - 2020-02-26 17:51 - Go MAEDA

- Target version changed from Candidate for next minor release to 4.1.1

Setting the target version to 4.1.1.

#14 - 2020-02-29 05:22 - Go MAEDA

- Status changed from New to Resolved
- Assignee set to Go MAEDA
- Resolution set to Fixed

Committed the patch. Thank you.

#15 - 2020-02-29 15:37 - Go MAEDA

- Status changed from Resolved to Closed

Files

3.2.1.png	20.4 KB	2020-02-08	Ji-Hyeon Gim
4.1.0.png	13.6 KB	2020-02-08	Ji-Hyeon Gim
32971-fixed.patch	824 Bytes	2020-02-17	Yuichi HARADA
32971-fixed-v3.patch	827 Bytes	2020-02-19	Yuichi HARADA
32971-fixed-v4.patch	1.69 KB	2020-02-20	Yuichi HARADA