

## subtle - Bug #293

### view.next and view.prev are broken?

05/22/2012 05:39 PM - Sae Hirak

<b>Status:</b>	Not a bug	<b>Start date:</b>	05/22/2012
<b>Priority:</b>	Urgent	<b>Due date:</b>	
<b>Assignee:</b>	Christoph Kappel	<b>% Done:</b>	0%
<b>Category:</b>	Unconfirmed	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Description</b>			
<p>I'm probably seriously misunderstanding the purpose of view.next and view.prev but I'm expecting them to return the next and previous views, respectively. In other words, these should be true:</p> <pre>Subtlex::View[0].next == Subtlex::View[1] Subtlex::View[1].prev == Subtlex::View[0]</pre> <p>Instead, from what I can tell, it's like this:</p> <pre>Subtlex::View[0].next == Subtlex::View[2] Subtlex::View[1].prev == Subtlex::View[1]</pre> <p>Basically, view.next 'skips' a view, so if you call the next method on the 1st view, it'll return the 3rd view, if you call it on the 3rd view, it'll return the 5th view, etc.</p> <p>And view.prev doesn't seem to work at all: it just returns the same view, so calling the prev method on the 2nd view just returns the 2nd view.</p> <p>Is this intentional, or am I missing something?</p>			

### History

#### #1 - 05/22/2012 11:39 PM - Christoph Kappel

I don't see any problems here?

```
>> Subtlex::View.current
=> test
>> Subtlex::View.current.prev.prev.prev.prev.next == Subtlex::View["terms"].next
=> true
>> Subtlex::View.all
=> [terms, www, void, sketch, test, editor]
```

#### #2 - 05/22/2012 11:41 PM - Christoph Kappel

- Category set to Unconfirmed
- Assignee set to Non member users
- Target version set to Xi

#### #3 - 05/29/2012 12:17 AM - Sae Hirak

Still not working for me. I tried using this to debug it:

```
on :client_create do |c|
  p Subtlex::View[:all]
  puts
  p Subtlex::View[4]
  p Subtlex::View[0]
  puts
  p Subtlex::View[4].prev.prev.prev.prev.next
  p Subtlex::View[0].next
end
```

Which gave me this:

```
[1, 2, 3, 4, 5, 6, 7, 8, , ]
```

5  
1

6  
2

I only have 8 views, so what's with the two empty ones at the end of the array? Also, why are your views named and mine aren't?

View[0] and View[4] are correct

View[4].prev.prev.prev.prev.next is **incorrect**, apparently because prev returns the same view rather than the previous one

View[0].next is correct

---

By the way, this:

```
on :client_create do |c|
  p Subtlex::View[1] == Subtlex::View[0].next
  p Subtlex::View[0] == Subtlex::View[1].prev
end
```

Still prints this:

```
false
false
```

Shouldn't those both be true?

#### #4 - 05/31/2012 01:15 PM - Christoph Kappel

I added your hook and both values are true as expected. Can you post your whole config? Must be something there, doesn't happen with either my or the default config.

#### #5 - 05/31/2012 10:42 PM - Sae Hirak

It's not my config. I just copied /etc/xdg/subtle/subtle.rb -> \$HOME/.config/subtle/subtle.rb and then added this at the end:

```
on :client_create do |c|
  p Subtlex::View[:all]
  puts
  p Subtlex::View[3]
  p Subtlex::View[0]
  puts
  p Subtlex::View[3].prev.prev.prev.next
  p Subtlex::View[0].next
end

on :client_create do |c|
  p Subtlex::View[1] == Subtlex::View[0].next
  p Subtlex::View[0] == Subtlex::View[1].prev
end
```

Which then printed this:

```
[1, 2, 3, 4, , ]

4
1

2
false
false
```

- There's two extra (empty) views.
- The views are unnamed.
- prev.prev.prev.next is incorrect.
- next is correct.
- Both print false.

Very odd that you aren't having the same problem... I'm using Debian Sid 64-bit Xfce4 with subtle 0.10.3212.

Just in case, here's my config, though I already pointed out that it happens with the default config: <http://pastebin.com/81EAMmMD>

**#6 - 05/31/2012 11:52 PM - Christoph Kappel**

Weird, Ruby version?

**#7 - 06/01/2012 02:57 AM - Sae Hirak**

```
$ ruby --version
ruby 1.9.3p0 (2011-10-30 revision 33570) [x86_64-linux]
```

**#8 - 06/01/2012 08:31 PM - Christoph Kappel**

I've asked several users in the project channel, no problems there. I have no idea why [subtext](#) behaves differently for you, I can't even explain why the #to\_str method returns numbers instead of names.

**#9 - 06/01/2012 10:59 PM - Sae Hirak**

Well, I guess you better mark this as WontFix until I can get some more information on why it happens.

**#10 - 06/15/2012 07:29 PM - Christoph Kappel**

- Target version changed from Xi to Omicron

**#11 - 12/24/2017 07:49 PM - Christoph Kappel**

- Target version deleted (Omicron)

**#12 - 01/04/2018 03:54 PM - Christoph Kappel**

- Status changed from Unconfirmed to Not a bug

Cannot reproduce this.