subtle - Bug #292

:name selector sometimes doesn't match?

05/22/2012 05:06 PM - Sae Hirak

Status:	Not a bug	**Start date:**	05/22/2012
Priority:	Urgent	**Due date:**	
Assignee:	Christoph Kappel	**% Done:**	0%
Category:	Unconfirmed	**Estimated time:**	0.00 hour
Target version:	Xi		

Description

I have tried the following, and **neither** of them match:

```
tag '5:bsnes' do
  match :instance => 'phoenix', :name => '^bsnes v\\d+$'
  geometry [ 508, 195, 584, 510 ]
  fixed true
end

tag '5:bsnes' do
  match :instance => 'phoenix', :name => 'bsnes v088'
  geometry [ 508, 195, 584, 510 ]
  fixed true
end
```

If I remove the :name selector it works fine. Here's the xprop output for the client:

```
WM_CLASS(STRING) = 'phoenix', 'Phoenix'
WM_ICON_NAME(STRING) = 'bsnes v088'
_NET_WM_ICON_NAME(UTF8_STRING) = 'bsnes v088'
WM_NAME(STRING) = 'bsnes v088'
_NET_WM_NAME(UTF8_STRING) = 'bsnes v088'
```

Unless I'm doing something wrong, it should work, shouldn't it? What makes it even weirder is that this (different) tag **does** work:

```
tag 'chat:ll' do
  match :instance => 'Steam.exe', :name => 'Friends'
  gravity :g15_ll
end
```

But this one doesn't:

```
tag 'chat:lr' do
  match :instance => 'Steam.exe', :name => ' - Chat$'
  gravity :g15_lr
end
```

So, it seems to me that :name works sometimes but not other times. I can't figure out why, though.

History

#1 - 05/22/2012 05:07 PM - Sae Hirak

Oh, the formatting broke it. It should have said:

```
match :instance => "phoenix", :name => "^bsnes v\\\\d+$"
```

That is, two backslashes, not one.

#2 - 05/22/2012 05:12 PM - Sae Hirak

Okay, I narrowed it down a little. This works:

```
match :instance => "phoenix", :name => /bsnes /
```

But this doesn't:

```
match :instance => "phoenix", :name => /bsnes v/
```

Does subtle do some sort of string manipulation on the regexps? If so, my best guess is that it's barfing on spaces in the regexp.

#3 - 05/22/2012 10:46 PM - Christoph Kappel

- Category set to Unconfirmed

- Status changed from Unconfirmed to Not a bug

- Assignee set to Non member users

- Target version set to Xi

You need to escape whitespaces, I don't do any manipulation at all.

Also keep in mind that matching WM_NAME is unreliable, because clients might change their WM_NAME during runtime after subtle applies tags.

#4 - 05/23/2012 10:36 AM - Sae Hirak

That's very unusual. I'm unaware of any regexp implementations that require whitespace escaping. Ah well, as long as it works, I don't care, though perhaps I should put a note somewhere in the wiki mentioning that.

#5 - 05/23/2012 02:24 PM - Christoph Kappel

Tbh, I don't understand it either. I have to use oniguruma, the regexp engine that is included in ruby, because they overwrite the symbols (e.g. struct regexp_t) from glibc..