

Getting rendered images to the browsers in Rails

Asked 16 years, 1 month ago Modified 16 years ago Viewed 4k times



2



I am writing a Rails app that processes data into a graph (using Scruffy). I am wondering how can I render the graph to a blog/string and then send the blog/string directly to the browser to be displayed (without saving it to a file)? Or do I need to render it, save it to a file, then display the saved image file in the browser?

[ruby-on-rails](#)

[ruby](#)

Share

Improve this question

Follow

edited Nov 6, 2008 at 20:12



[John Topley](#)

115k ● 47 ● 199 ● 240

asked Nov 6, 2008 at 7:18



[Josh Moore](#)

13.5k ● 15 ● 61 ● 74

3 Answers

Sorted by:

Highest score (default)





I think you will be able to use [send_data](#) for this purpose:

7

```
send_data data_string, :filename => 'icon.jpg', :type  
:disposition => 'inline'
```



If you put this in a controller action - say show on a picture controller, then all you need do is include the following in your view (assuming RESTful routes):



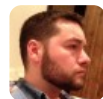
```
<%= image_tag picture_path(@picture) %>
```

Share Improve this answer

edited Nov 6, 2008 at 9:53

Follow

answered Nov 6, 2008 at 9:18



Codebeef

43.9k ● 23 ● 88 ● 119

Matt, how do I reference this file in the view? – [Josh Moore](#)

Nov 6, 2008 at 9:22



I wonder if sending direct to the browser is the best way? If there is the possibility that users will reload the page would this short circuit any cache possibilities? I ask because I really don't know.

0



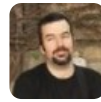
answered Nov 6, 2008 at 11:55





Share Improve this answer

Follow



srboisvert

12.7k ● 15 ● 65 ● 87



0



"If there is the possibility that users will reload the page would this short circuit any cache possibilities?"



No - whether you're serving from a file system or send_data doesn't matter. The browser is getting the data from your server anyway. Just make sure you've got your HTTP caching directives sorted out.

Share Improve this answer

Follow

answered Nov 25, 2008 at 8:58



Xavier Shay