

Replace images with icon fonts

2022/05/09 18:06 - Admin Redmine

| | | | |
|-----------------|--------------------------------------|----------------------|------------|
| ステータス: | New | 開始日: | 2022/05/09 |
| 優先度: | 通常 | 期日: | |
| 担当者: | | 進捗率: | 0% |
| カテゴリ: | UI_10 | 予定工数: | 0.00時間 |
| 対象バージョン: | Candidate for next major release_32 | 作業時間: | 0.00時間 |
| Redmineorg_URL: | https://www.redmine.org/issues/23980 | status_id: | 1 |
| category_id: | 10 | tracker_id: | 3 |
| version_id: | 32 | plus1: | 1 |
| issue_org_id: | 23980 | affected_version: | |
| author_id: | 107353 | closed_on: | |
| assigned_to_id: | 107353 | affected_version_id: | |
| comments: | 49 | | |

説明

Icon fonts have some advantages over the classical images:

- being vector graphics, they are scalable and can be resized without losing quality.
- can be customized directly from CSS (size, colour, etc)
- less HTTP requests to server because they are loaded only with one or a few requests. Now, Redmine make a request for each image.
- some of the current custom themes already use icon fonts (Abacus theme, Minelab, PurpleMine2, our custom theme and I think the theme from plan.io).

We're interested to contribute with a patch that implements the FontAwesome icons, but because there are at least two ways to implement them, we want some feedback before from Redmine contributors and/or users.

1. Using i tags:

Advantages:

- Is the recommended way on their "official page": <http://fontawesome.io/examples/#basic>
- We can use all the build-in rules available in the FontAwesome CSS.

Disadvantages:

- it'll be required to add the i elements in views.

1. Only from css

Advantages:

- the views will not be modified

Disadvantages:

- The build-in rules must be reimplemented in the current CSS
- The icons will be defined using their unicode. For example, the fa-pencil icon (similar with the current images for icon-edit) has the unicode f040.

Only for demo purposes, I've attached a patch that replaces the icons from issue page with font awesome icons (using i tags).

journals

Marius BALTEANU wrote:

1. Only from css

Advantages:

- the views will not be modified

Disadvantages:

- The build-in rules must be reimplemented in the current CSS
- The icons will be defined using their unicode. For example, the fa-pencil icon (similar with the current images for icon-edit) has the unicode f040.

The "disadvantages" word should be after the first advantage. Like is in this comment.

Attached a screenshot also.

!font_awesome.png!

Hi Marius, thanks for proposing this. I think replacing the current icon set with an icon font is a great idea. As you noticed, we're using Font Awesome for the "new Planio design": <http://plan.io/blog/post/135376091228/the-challenge-of-redesigning-redmine>.

Marius BALTEANU wrote:

because there are at least two ways to implement them, we want some feedback

Before implementing our new design, we were discussing this intensively. We decided against using @@ tags and in favour of adding new CSS rules, for the following reasons:

- Changing all icons in Redmine to @@ tags would be a very large patch touching almost all views in Redmine (as you probably already noticed while preparing your demo patch).
- Using @@ tags would require all plugin developers to change plugins as well and there would be a (potentially long) period where plugins would still use the "old" icons while Redmine is already using "new" icons making the overall user experience inconsistent.

Implementing new CSS rules for all icons using Font Awesome, however, would be less of a change to existing Redmine code (we would be able to leave views alone) and would apply to all existing plugins using standard Redmine icons (like edit, delete, etc.) at once, so plugin developers would have few things to change or nothing at all.

To respond to the disadvantages mentioned:

The build-in rules must be reimplemented in the current CSS

I actually see this as an advantage: implementing Icon style can be encapsulated in a defined place within the CSS and does not happen all over the place.

The icons will be defined using their unicode. For example, the fa-pencil icon (similar with the current images for icon-edit) has the unicode f040.

From our experience in working with Font Awesome in our new UI, it isn't much of a problem. The icons can be looked up easily via <http://fontawesome.io/icons/>, and we could use CSS comments to note the Font Awesome icon names next to the unicodes.

One sidenote: At Planio, we're using "Rails' Asset Pipeline": http://guides.rubyonrails.org/asset_pipeline.html with Redmine and are thus using "SCSS": <http://sass-lang.com/> and the "fontawesome-rails gem": <https://github.com/bokmann/font-awesome-rails> which is a tremendous help.

Jan from Planio www.plan.io wrote:

Hi Marius, thanks for proposing this. I think replacing the current icon set with an icon font is a great idea. As you noticed, we're using Font Awesome for the "new Planio design": <http://plan.io/blog/post/135376091228/the-challenge-of-redesigning-redmine>.

Marius BALTEANU wrote:

because there are at least two ways to implement them, we want some feedback

Before implementing our new design, we were discussing this intensively. We decided against using @@ tags and in favour of adding new CSS rules, for the following reasons:

- Changing all icons in Redmine to @@ tags would be a very large patch touching almost all views in Redmine (as you probably already noticed while preparing your demo patch).
- Using @@ tags would require all plugin developers to change plugins as well and there would be a (potentially long) period where plugins would still use the "old" icons while Redmine is already using "new" icons making the overall user experience inconsistent.

Implementing new CSS rules for all icons using Font Awesome, however, would be less of a change to existing Redmine code (we would be able to leave views alone) and would apply to all existing plugins using standard Redmine icons (like edit, delete, etc.) at once, so plugin developers would have few things to change or nothing at all.

To respond to the disadvantages mentioned:

The build-in rules must be reimplemented in the current CSS

I actually see this as an advantage: implementing Icon style can be encapsulated in a defined place within the CSS and does not happen all over the place.

The icons will be defined using their unicode. For example, the fa-pencil icon (similar with the current images for icon-edit) has the unicode f040.

From our experience in working with Font Awesome in our new UI, it isn't much of a problem. The icons can be looked up easily via <http://fontawesome.io/icons/> and we could use CSS comments to note the Font Awesome icon names next to the unicodes.

Totally agree with you, I'll try to create a patch using only CSS.

One sidenote: At Planio, we're using "Rails' Asset Pipeline": http://guides.rubyonrails.org/asset_pipeline.html with Redmine and are thus using "SCSS": <http://sass-lang.com/> and the "fontawesome-rails gem": <https://github.com/bokmann/font-awesome-rails> which is a tremendous help.

I know that were some discussions about the Rails' Asset Pipeline (I can't find the ticket now) and enabling this feature is not an option because will add some complexity. If I'm wrong, I'll be very happy to use that gem.

Thanks for your feedback.

I've attached a first patch which:

- adds the Font Awesome fonts
- replaces the images icons with font awesome icons for all elements that have icon and icon-* classes

Some observations:

1. Currently, there are some specific programming images for file types like text-x-php, text-x-c, text-x-java and so on. Because FontAwesome doesn't have these types of icons, I see 3 ways to solve this problem: 1.1 use the current images 1.2 use the "generic file code icon": <http://fontawesome.io/icon/file-code-o/> provided by FontAwesome for all of these files. 1.3 add "Font-Mfizz": <http://fizzed.com/oss/font-mfizz> which is an extension of FontAwesome icons and provides the required icons.

IMHO, I prefer the second solution because:

- it doesn't add new fonts
- it doesn't require to keep the rules related to background images in icon class
- it doesn't require maintenance when a new type of files appears (in this patch I've added a new class named "application-javascript" in order to recognize the javascript files)

1. I'll add separate patches that replaces the current images with FontAwesome icons in other pages like: administration, project overview, etc..

2. In the last patch, I'll remove the unused images.

Any feedback is appreciated.

I would prefer the Font-Mfizz solution. It does not require much more server load using it (there are a lot less server requests compared to the background image solution). Main reason for this solutions is, that the usability and the user acceptance would be higher. A lot of technical teams are working with Redmine and SCM, and these people will be certainly happy to see file type specific icons.

We discussed internally in "Zitec":<https://zitec.com> and we are going to implement the solution with Font-Mfizz but for the moment we are waiting for some feedback on issue #24313 because having that ticket committed, it will be much easier for us to implement this one.

#24313 has been implemented.

I am looking forward to seeing this feature in 3.4.0. Setting target version to "Candidate for next major release".

Go MAEDA wrote:

#24313 has been implemented.

I am looking forward to seeing this feature in 3.4.0. Setting target version to "Candidate for next major release".

The patch that replaces the images with fa icons is ready, but I want to take some feedback from our users regarding the icons color. Now all the icons have the same color (# 169). Please let me know if you want to test it as it is now.

The attached patch (replace_images_with_fa_icons.patch) replaces the images for all regular icons (that have the classes "icon icon-*") with font-awesome icons.

Because the git binary diffs are not supported by the patch command, I wasn't able to add the required fonts as patches, but I've documented below the necessary steps:

1. Create a folder fonts in the public directory from redmine.
2. Unzip the attached font-mfizz.zip in the new fonts folder.
3. Because the font-awesome archive is bigger than the maximum allowed size for attachments (600kb), the zip can be downloaded from "here":<https://drive.google.com/file/d/0BymeVD9kRWHEc003a3pKX0ZxcGc/view?usp=sharing>. If the official "link":<http://fontawesome.io/assets/font-awesome-4.7.0.zip> is preferred, the fonts must be extracted from the archive (folder fonts) to a folder named font-awesome in the same folder fonts from public directory.

At the end, the fonts folder from public should have 2 directories:

- font-awesome
- font-mfizz

!fonts_folder.png!

There are still some elements that are using images (like classes expander, asc, desc), but I think that is safer to replace them with FA icons in another ticket (requires more changes). Also, I think the images should be removed in a later major release (for plugins compatibility).

Any feedback on the chosen icons is welcome.

Some screenshots.

Updated to include a fix for the responsive mode (the left padding is no more required).

Any feedback on this patch? FTR, I tried to create a single patch with all the changes with Subversion too but it didn't work.

Screenshots looks nice for me.

Jean-Phillipe, is there any possibility that this patch will be merged?

Jean-Philippe, do you have any feedback on this?

This patch is a great improvement to Redmine. It extends the variety of icons available to developers of plugins and Redmine itself. It's a pity that there hasn't been more discussion about this improvement for a year now.

Unfortunately this patch doesn't work anymore.

@Marius BALTEANU is it possible to update this patch to a recent revision of Redmine? I would really love to see this patch applied in version:4.0.0 or version:4.1.0.

+1 for this!

Bernhard Rohloff wrote:

@Marius BALTEANU is it possible to update this patch to a recent revision of Redmine? I would really love to see this patch applied in version:4.0.0 or version:4.1.0.

Sure, I'll try to do it this month (including the update to Font Awesome 5).

"GitHub changed from Icon fonts to SVG icons.":<https://github.com/blog/2112-delivering-octicons-with-svg>

Icon fonts is not bad, but I think it will be better if you change to SVG icons.

Taiki IKEGAME wrote:

"GitHub changed from Icon fonts to SVG icons.":<https://github.com/blog/2112-delivering-octicons-with-svg>
Icon fonts is not bad, but I think it will be better if you change to SVG icons.

This also sounds very interesting to me and seems to be more versatile than icon fonts. It could be nice to load custom icons in a plugin and use them with the same helper method as the core icons. Or just to install a custom icon pack as a plugin. Just a few thoughts. I'm not so well informed about this topic right now. So are there some other opinions about preferring SVG icons over icon fonts?

Fontawesome 5 recommends "JS with SVG":<https://fontawesome.com/get-started/svg-with-js>.

Sorry for not checking well.

Yey! This asap pls!  <https://cdn.discordapp.com/emojis/471854100984758274.png>!

+1

Max Johansson wrote:

Yey! This asap pls!  <https://cdn.discordapp.com/emojis/471854100984758274.png>!
+1

Max, I'll be very happy to update my patch if Jean-Philippe agrees to replace the current

icons with FA icons.

Marius,  <https://cdn.discordapp.com/emojis/468702863795683328.png>!

Jean-Philippe, could you please let us know your opinion about this?

Marius, maybe it will be easier to update the patch and sneak-commit it with help of Go? Then we could cross fingers and hope that Jean-Philippe might not notice it :D

Max Johansson wrote:

Marius, maybe it will be easier to update the patch and sneak-commit it with help of Go?

I will not commit such a big change without Jean-Philippe's consent. I never forget that he is the owner of this software and we can use Redmine thanks to him and his many years of efforts.

Go MAEDA wrote:

I will not commit such a big change without Jean-Philippe's consent. I never forget that he is the owner of this software and we can use Redmine thanks to him and his many years of efforts.

Absolutely agree!

Basically I agree with Go and Jan. It's not an option to make decisions behind JP's back. As much as I share this opinion I can understand Max's intention. This issue is nearly three years old and no decisions are made. So has anybody a better idea on how to get at least a yes, no, maybe from Redmine's kahuna?

So has anybody a better idea on how to get at least a yes, no, maybe from Redmine's kahuna?

I guess the only way I could come up with is to bump this ticket with a message mentioning JPL and urgent to be visible in the activity log of Redmine.org next time a commit streak from JPL will occur. And it can be detected with help of #redmine-commits channel on our Discord, but if RSS tracker bot will not logout again (Discord been having problems with bots logging out lately, but they are working on fixing it afaiK) and there will be anyone available or online to bump this message in that perfect moment :)

Bump! Jean-Philippe, this ticket is ultra-urgent to many, please share your consent.

Max Johansson wrote:

Bump! Jean-Philippe, this ticket is ultra-urgent to many, please share your consent.

Max, let be honest, it is not so ultra-urgent, it is just a nice UI improvement for the core and for the theme developers. I'll update my patch to Font Awesome 5 and then I'll share my thoughts regarding this feature.

But I partially agree, a feedback from Jean-Philippe could be useful to not work in the wrong direction.

"Here": https://gitlab.com/redmine-org/redmine/merge_requests/3 is an updated version of my patch. Because the size of the patch is too big, you need to download it from my gitlab instance (screenshot attached for those who are not familiar with GitLab -

attachment:gitlab.png).

The patch includes now the latest version of Font Awesome (5.8.2) and replaces the images with font icons only for elements with 'icon icon-*' classes. I'll add very soon another patch for the rest of the elements.

Please test it and let me know what do you think.

Marius BALTEANU wrote:

"Here":https://gitlab.com/redmine-org/redmine/merge_requests/3 is an updated version of my patch.

Since the page requires authentication, I could not get the patch.

Go MAEDA wrote:

Marius BALTEANU wrote:

"Here":https://gitlab.com/redmine-org/redmine/merge_requests/3 is an updated version of my patch.

Since the page requires authentication, I could not get the patch.

Sorry, I didn't know that the project is private. I made it public now, you should have access.

I've have updated my patch to include the latest changes made in #31433 and #31496.

Considering that most of the icons are now replaced with font awesome icons, I'm assigning this to version:"4.1.0" in order to take a feedback from Jean-Philippe.

There are still some images that can be replaced with FA icons, but we can do it in a later version. I'm really sure that it'll be a great improvement to deliver this for everyone (users, theme developers, plugin developers, 3rd party saas providers like Plan.io).

+1

I hope this feature will be released as soon as possible.

The current icon is hard to see due to its small size and is not kind to the user.

Updated the patch and added a new commit to replace the images from JS Toolbar with FA icons.

Toolbar with current images:

!before.png!

Toolbar with FA icons*:

!after.png!

*The icons for Inline code, H1, H2 and H3 are not really icons because FA doesn't have such icons, are just text.

The patches (2 commits) can be found "here":

https://gitlab.com/redmine-org/redmine/merge_requests/3/commits.

This is a major change for plugins and themes, reassigning to 5.0.

I've started to look again on this and I think it's better to move from icon fonts to svgs if we really do this step. Most of the major apps that I use did this and Github wrote a very nice article about this (<https://github.blog/2016-02-22-delivering-octicons-with-svg/>).

I will post in a few days a PoC for this change which allows easy customisation by plugin

developers or to keep the existing functionality.

I made a demo patch that adds SVGs instead of icon fonts for issue page:

- rendering SVG inline (in HTML) make them fully customisable from CSS (size, fill, stroke, etc)
- added a Helper class to render all icons/SVGs. This means that are easily customisable/extendable.
- plugin developers can use this new class to render the icons using the existing SVGs or they can add new SVGs
- the SVGs used are from Font Awesome (latest 5.* version).

!{border: 1px solid grey;}issue.png!

Marius BALTEANU wrote:

I made a demo patch that adds SVGs instead of icon fonts for issue page:

Marius I've tested your patch. IMHO it looks really nice. I would love to see this landing in v5.0.0.

Marius BALTEANU wrote:

I made a demo patch that adds SVGs instead of icon fonts for issue page

That looks very good.

I know of a few themes that replace icon images with icon fonts.

For those themes, you can continue to use your own icons without making any major changes by applying the following CSS.

```
.icon-svg svg {display: none;}
```

This change looks like it could be released without breaking the theme too much.

I'm very much looking forward to its introduction.

Although svg icons may require more effort than icon fonts, they can be implemented using only standard technology.

On the other hand, I see the following issues with the current proposal (I understand that the attached patch is for verification purposes).

Inline svg images do not cache svg files.

The new helper would require significant modifications, including plugins.

We need a mechanism to add icons in themes and plugins.

We need a way to add/remove svg icons with javascript.

It is tedious to check and obtain updates of external resources that distribute svg icons.

I am attaching patches to solve the problem. Icons are from fontawesome and material design using a service called iconify. Note that it does not work in IE11.

For consistency of icons with themes and plugins, this feature can be disabled in the configuration file. After applying patches, We must set `config.redmine_enable_svg_icon` to true in "config/application.rb" to enable it.

1 Inline svg images do not cache svg files.

- We can use svg from another file by adding a symbol element to svg and using "href" in the use element.

- Stylesheets can control the size and the color of the icons.

2 The new helper would require significant modifications, including plugins.

- I patched to tag helper to add icons, check classes and add necessary svg icons if icon class or icon-only class is present.
- link_to and content_tag do not need to be modified since they use the tag helper internally.
- Rewrite the part of the html in the view with a helper (we must notify the plugin authors).

3 We need a mechanism to add icons from themes and plugins.

- Prepare common.yml ("class name" "svg" correspondence).
- common.yml prepared by plugin and theme will be merged into icon data in Redmine at startup.
- svg icon is generated by default when we generate a new plugin.

4 We need a way to add/remove svg icons with javascript.

- create @addSVGIcon@ and @removeSVGIcon@ method.
- It is necessary to announce to plugin/theme authors that the method of toggling icon display by simply adding/removing CSS classes is not available.

1. It is tedious to check and obtain updates of external resources that distribute svg icons.

- @rake icon:fetch@ to fetch required icons at once.

patches can be applied to r21487

related_issues

relates,New,5830,Replace famfamfam icons with the fugue set

relates,New,11757,Add support for HDPI screens (retina)

relates,Closed,30229,Optimization: all UI icons collapsed into a single sprite

履歴

#1 - 2022/05/10 17:06 - Admin Redmine

- カテゴリをUI_10にセット

- 対象バージョンをCandidate for next major release_32にセット