

UPCOMING: Build.GE GitHub Infrastructure Migration. Please read [the Build.GE GitHub Yammer announcement](#).

cicada / ui

212750376 requested your review on this pull request.

Add your review

Update Deployments Page Tabs to Use Router #320

Edit

Open

212750376 wants to merge 23 commits into dev from feature/breadcrumb_trace_in_services-joe-recommendations

Conversation 32

Commits 23

Checks 0

Files changed 8

+169 -160

joe

212750376 commented 8 days ago • edited

Member

User Story (or Defect or Whatever Else It Is):
<https://rally1.rallydev.com/#/255730482452d/iterationstatus?qdp=%2Fdetail%2Fuserstory%2F348297202096&sharedViewId=344816267284>

Pull Request Administrativa

PR Opener Admin TDL:

☐

As part of this PR, update the [Unreleased] section of [the Changelog](#) to summarize the changes in this PR in an end-user relevant manner.

- If this PR is pure documentation or otherwise does not affect the build artifact then this task can be skipped/checked off.

PR Reviewer Admin TDL:

☐

On Approval, bump the version by the appropriate increment with npm version ...

- If this PR is pure documentation or otherwise does not affect the build artifact then this task can be skipped/checked off.

Pull Request Details

Changes:

- Update the Deployments Menu to use the Router.
- Give each Deployments Tab its own route.

TODO:

☐

Update the breadcrumb to display the current route when on tabs.

Reviewers

503152573

Assignees

503152573

Labels

wip

Projects

None yet

Milestone

No milestone

Notifications

2 participants

joe

- 212750376 added some commits 8 days ago
- joe

IServiceTab and predicate ...5e0e032
- joe

Define tabs in separate file ...5a5f4dc
- joe

DeploymentsMenu: Update tabs to use routera35490f
- joe


DeploymentsPage: Render router view415d131
- joe

router: add routes for deployments page tabs6e7837a
- joe


StyleguidePage: More rigorous logic in route defs34b3c46
- joe

DeploymentsPage: Render tab components with router-viewc589793
- joe

store/Deployments: delete tab state✓ a5bf992

 joe 212750376 added the `do not merge yet` label 8 days ago

 joe 212750376 requested a review from 503152573 8 days ago

 joe DeploymentsMenu: Remove unused prop in v-tabs ✓ 8238061

joe

 212750376 reviewed 8 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

...

45

46

47

48

...

45

46

47

48

@@ -45,21 +45,21 @@

</v-flex>

</v-layout>

- <v-tabs v-model="active">


+ <v-tabs>

joe 212750376 8 days ago

Author

Member

Because each `<v-tab>` is now pointing at a route using `<v-tab :to="routeObject">`, we don't need to track the state using `<v-tabs v-model="whatever">` ourselves. Instead, we can let `<v-tabs>` use its internal state tracking to keep the current tab synched with the current route.



Reply...

Resolve conversation

joe

 212750376 reviewed 8 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

112

85

86

87

-];

+

+ @Prop({ type: Array, required: true })

+ public visibleTabs!: IServiceTab[];

joe 212750376 8 days ago


Author

Member

This is made into a prop, and changed to be the visible tabs, for two reasons:

- The DeploymentsPage component already has to have the tabs in order to be able to show certain error messages.
- The DeploymentsPage component already must compute what tabs to actually show (`visibleTabs`) in order to know when to show those error messages and when not to.

Therefore, it's a duplication of computation to redo those in this component, so we just have the DeploymentsPage do that instead and pass them down.



503152573 7 days ago

Member

I inlined `IServiceTab[]` based on `IServiceTab.interface.ts` found in Build Infra Services: Services Tabs #304

joe 212750376 7 days ago

Author

Member

Indeed. I just checked out the file directly from #304 instead so that it would be in the same place.



Reply...

Resolve conversation

joe

212750376 reviewed 8 days ago

View changes

src/components/DeploymentsPage/DeploymentsPage.vue

...

6

7

8

9

...

6

7

8

9

@@ -6,11 +6,13 @@

<v-fade-transition>

<dynamic-container v-if="deploymentsPageState === 'READY'" key="data">

<!-- Deployments Menu Bar -->

<deployments-menu></deployments-menu>

<deployments-menu :visible-tabs="visibleTabs" />

joe

212750376 8 days ago

Author

Member

Again, since this component DeploymentsPage must already calculate the visible tabs, we just pass that same value down to the DeploymentsMenu component.

Reply...

Resolve conversation

joe router: Wildcard redirect for service tabs

7644bd4

joe

212750376 reviewed 8 days ago

View changes

src/components/DeploymentsPage/DeploymentsPage.vue

11

12

13

-

-

-

<keep-alive v-if="isSelectedDeploymentTabSupported">

<component

:is="selectedDeploymentTab"

<template v-if="!selectedTab">

joe

212750376 8 days ago

Author

Member

Because selectedTab is based on the current route and may be null or undefined, we must now account for that case in the template.

Since the only reason we shouldn't have a selected tab is if we haven't yet redirect the user to one of those tabs, we just show a loading indicator.

503152573 7 days ago

Member

It seems to be previous version, as the last one included router-view:

How should I modify it ?

Reply...

Resolve conversation

joe

212750376 reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsPage.vue

...

126

127

128

...

131

132

133

@@ -126,6 +131,11 @@ export default class DeploymentsPage extends Vue {

*/

protected isLoadingInitialData: boolean = false;

134

+

// We have to deep-clone it because vue-router mutates the objects you pass to

135

+

protected renderableTabs: IServiceTab[] = deepClone(

joe

212750376 7 days ago

Author

Member

About the only difference here is that this is now pulled from another file. Also of course we have to account for the noted Vue Router idiosyncrasy.

503152573 7 days ago

Member

Are some lines are missing ?

joe

212750376 7 days ago

Author

Member

They are still present after this line. They are not shown in this snippet because GH only lets you make comments on one line and not sets of lines, and I placed the comment on the first line of the thing I was commenting on.

Reply...

Resolve conversation

⚙

joe

deploymentsPageTabs: Remove getters in favor of cloning

✓ 242e6a2

joe

👁

212750376 reviewed 7 days ago

View changes

src/interfaces/IServiceTab.interface.ts

Outdated

Show outdated

joe

👁

212750376 reviewed 7 days ago

View changes

src/util/router.ts

...

...

@@ -52,7 +54,56 @@ export const routes = [

52

54

import(

53

55

/* webpackChunkName: "DeploymentsPage.vue"*/ '@/components/DeploymentsPag

54

56

),

55

-

props: true,

joe

212750376 7 days ago

Author

Member

props: true had to be removed because it was causing strange errors with trying to set an HTML attribute named `0`, which is not a valid HTML attribute name.

503152573 7 days ago

Member

Seems that this was the previous version of the file, since my current one does not include that line ('props: true')

joe

212750376 7 days ago

Author

Member

Dev still has that line, and this PR was made against dev, which is why this diff shows the line being removed.

Reply...

Resolve conversation

joe

👁

212750376 reviewed 7 days ago

View changes

src/util/router.ts

100

+

),

101

+

},

102

+

{

103

+

path: '*',

joe


212750376 7 days ago

Author

Member

This is just a nice-to-have to make sure entering a tab that does not have a defined route will kick the user over to a known location.

An alternative is to have a component that shows an "unknown feature" message, but that's not currently implemented.



503152573 6 days ago

Member

Fo now, I added below, to be redirected to DeploymentsPage rather than ProductsPage

```
{
  path: '',
  component: () =>
    import(
      / webpackChunkName: "DeploymentsPage.vue"*/
      '@components/DeploymentsPage/DeploymentsTab/DeploymentsTab.vue'
    ),
}
```

joe

212750376 6 days ago • edited

Author

Member


That presumes that the DeploymentsTab is:

- Always supported by that service
- Always the default tab for that service

This is not the proper behavior, though:

- We do not know what tab is the "default tab" for a given service until after we have loaded the supported features for that service.
 - Note also that a service may support no features, in which case *no tab* can be the "default tab".
- Therefore we cannot assume that the DeploymentsTab will be the "default tab".

This is why my redirect goes to the index page, and why I do not specify a no-sub-path route.



Reply...

Resolve conversation

joe

212750376 reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

77

78

79

80

81

82

83

84

78

79

80

81

82

83

84

```
@Component({
  name: 'DeploymentsMenu',
})
export default class DeploymentsMenu extends Vue {
  public isLoading: boolean = true;
  public links: IEntityLink = {} as IEntityLink;
  - public active: number = 0;
```


joe

212750376 7 days ago

Author

Member

Since we no longer need to manage the state of the tabs, we can just delete this property.



Reply...

Resolve conversation

joe

212750376 reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

7778

7879

7980

8081

8182

8283

8384

8485

@Component({

name: 'DeploymentsMenu',

})

export default class DeploymentsMenu extends Vue {

public isLoading: boolean = true;

public links: IEntityLink = {} as IEntityLink;

- public active: number = 0;

- public text: string = '';

joe

212750376 7 days ago

Author

Member

This wasn't being used anywhere, so it also was deleted.



Reply...

Resolve conversation

joe

212750376 reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

...131132133134

...106107108

@@ -131,14 +106,6 @@ export default class DeploymentsMenu extends Vue {

return null;

}

- get serviceFeaturesData() {

joe

212750376 7 days ago

Author

Member

Since we're getting the tabs to render from our parent, we no longer need to get anything from the store, so this gets deleted.



503152573 6 days ago

Member

I need to work on that a bit, as when I detete it, its parents and children, service tabs bar disappears

joe

212750376 6 days ago

Author

Member

Note what prop the <v-tabs><v-tab> components in the template are drawing from in this PR, and work backwards from there.

If I had to guess, you might have deleted it but not anything that depended on it, leading to both type and reference errors. The JS console in the browser should show those.



Reply...

Resolve conversation

joe

212750376 reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

165

-

);

166

-

}

167

-

168

-

get tabsVisible(): Array<{

joe

212750376

7 days ago

Author

Member

This is now a prop that gets passed down from the parent, so it and the data to compute it can all be deleted.



Reply...

Resolve conversation

joe

212750376

reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

...

...

@@ -197,25 +136,10 @@ export default class DeploymentsMenu extends Vue {

197

136

return 'this service';

198

137

}

199

138

200

-

public handleClickTab(selectedComponent: ServiceFeature): void {

joe


212750376

7 days ago

Author



Member

No more need for this handler since <v-tab :to="routeObject"> is being used.




Reply...

Resolve conversation

-  212750376 added some commits 7 days ago
- 

joe DeploymentsMenu: Almost always redirect from no tab to tab

✓ 4d4c6ea
- 

joe DeploymentsMenu: Use router.replace so users can use back button

✓ dc4de97

joe

212750376

reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

216

-

this.\$store,

217

-

targetTab

218

-

);

147

+

@Watch('\$route')

joe

212750376


7 days ago • edited

Author

Member

We need to watch the route here to catch cases where the user may navigate to the index page (that is, not a tab route) without actually leaving the page itself.

We can't redirect if we have no tabs to redirect to, though, so we just don't even bother in that case.



Reply...

Resolve conversation

joe

212750376

reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsMenu.vue

140

+

protected

selectInitialTab() {

141

+

const

firstTab = this.visibleTabs[0];

142

+

if

(firstTab !== undefined) {

143

+

this.\$router.replace

(firstTab.route);

joe

212750376

7 days ago

Author

Member

We use `$router.replace` here so that the user can actually use the back button to get out of the Deployments Page and go back to the Services Page.



Reply...

Resolve conversation

 joe DeploymentsPage: Delete redundant imports 1d8f9e8

joe

 212750376 reviewed 7 days ago

View changes

src/components/DeploymentsPage/DeploymentsPage.vue

...

...

@@ -89,26 +94,6 @@

enum DeploymentsPageState {

89

94

import

(

90

95

/* webpackChunkName: "DeploymentsPage.vue" */

'@/components/DeploymentsPage

91

96

),

92

-

DeploymentsTab: () =>

joe

212750376

7 days ago

Author

Member

Since all the tabs' components are imported by the router, we no longer need to reference them here.



Reply...

Resolve conversation

 joe DeploymentsMenu: Explain why to use `$router.replace()` 03a850f

joe

212750376 commented 7 days ago • edited

Author

Member

NOTE: This was tested to work as expected but hasn't been tested by anyone else. There may be issues in it, but I assert that it at least functions as I expected it to.

When I was giving you earlier feedback on your PR, I think I had the right ideas when starting out, but didn't actually settle on a good implementation until later so some of the code I pointed you to is actually out of date. That's my bad.

In this example PR, I've implemented changes to enact the same behavior as I made on the Build Infra Service Details PR, though I had to checkout one file directly from my branch which is kinda dirty. Ah well, I'll deal with that conflict when we get to it.

I've added [line comments](#) that I hope will be useful for explaining particular things, and want to give a general overview of my approach here.

Also, as a caveat, the approach I describe below in the order it's described doesn't exactly line up with the order I did things in the commits on this PR. This is a sort of cleaned up high level summary.

Moving the "Currently Selected Tab" State

So, the first thing is this: since whatever current tab the user selects is going to be in the route, the route must now be the source of truth on the matter. This means we must delete that entirely from the state, it's no longer useful. Before we can do that, however, we need to have some way of associating a certain route with a certain tab.

Service Tab Definitions

To solve that, I created a [separate file that has the description of each tab, as well as an array that is currently used for both the order of tabs in the menu and for their priority for being the default tab](#). The priority, of course, is the order in the array: the first tab in the array that is supported by the given service is the default tab for that service.

So, that file gives us globally defined associations of Service Feature, Tab Title, and Route. By having a defined Title, we can also avoid having to do any string manipulation which may go awry in the future.

The Tab Routes

Now that we have an association between Tabs and Routes, we can [mark the Routes in the Router config](#):

```
ui/src/util/router.ts
Lines 57 to 106 in dc4de97

57   children: [
58     {
59       path: 'deployments',
60       name: 'service-details-deployments',
61       meta: {
62         serviceTab: deploymentsPageTabs.deploymentsTab,
63       },
64       component: () =>
65         import(
66           /* webpackChunkName: "DeploymentsPage.vue" */ '@components/DeploymentsPage/Dep
67         ),
```

I'd originally done this using just the Service Feature, but it became easier in the rest of the code to just use the whole ServiceTab objects. Currently however this makes things a bit redundant, what with each ServiceTab object also having a Route on it, but that's fine for now.

As an aside, one other reason I did it this particular way is because when I tried to import the route config in my PR, I got a circular dependency error. I felt a bit of redundancy was better than trying to figure out how to break that.

With those associations defined, we can now move on updating the rest of the code.

The Old State

This is the easiest part.

As stated before, we're moving the state tracking of "what tab is currently selected" out of the Vuex Store and into the Route. This means we can and should delete the old store state and associated accessors.

- <https://github.build.ge.com/cicada/ui/pull/320/files?file-filters%5B%5D=.ts#diff-0d57c579ad3c2c3c1e8831e680702b41>
- <https://github.build.ge.com/cicada/ui/pull/320/files?file-filters%5B%5D=.ts#diff-21e8168b4f159e2affb3fb1e50ccfb89>

That of course will give us a whole bunch of "missing export" errors because suddenly a bunch of things are missing.

Updating the Views

Updating the Deployments Page

Next, we will want to update the Deployments Page.

Of note here is that we actually require the list of tabs to be on the Deployments Page itself because the Deployments Page uses the Service Tabs to do some checks for proper case handling in the template, and for proper messaging. So, we attach that here:

ui/src/components/DeploymentsPage/DeploymentsPage.vue
Lines 114 to 117 in 1d8f9e8

```
114    // We have to deep-clone it because vue-router mutates the objects you pass to it.  
115    protected renderableTabs: IServiceTab[] = deepClone(  
116        deploymentsPageTabs.renderableTabs  
117    );
```

We then do all of the necessary derived data computations elsewhere in the component as needed, as well as take [the currently visible tabs](#) and [pass them down to the menu](#).

And, perhaps most important, we change the `<component>` to a `<router-view>`:

ui/src/components/DeploymentsPage/DeploymentsPage.vue
Lines 15 to 20 in 1d8f9e8

```
15    <router-view  
16        :deployment-status-subscription-id="deploymentStatusSubscriptionId"  
17        :current-in-progress-deployment-status="  
18            currentInProgressDeploymentStatus  
19        ">  
20    />
```

Updating the Deployments Menu

This component slims down a bit, now that we don't have to worry about rederiving all that tab data: we just get what tabs should be rendered from the parent and we're done with it.

There's a slight update to `selectInitialTab()` in light of the change to using the Router instead of the Store:

ui/src/components/DeploymentsPage/DeploymentsMenu.vue
Lines 140 to 147 in 03a850f

```
140    protected selectInitialTab() {  
141        const firstTab = this.visibleTabs[0];  
142        if (firstTab !== undefined) {  
143            // Use .replace() so that the user can actually use the back button of their browser  
144            // to back out of the Deployments page and go back to the Services page.  
145            this.$router.replace(firstTab.route);  
146        }  
147    }
```

And then an added watch to ensure the user always gets redirected when ever they land on the index page:

ui/src/components/DeploymentsPage/DeploymentsMenu.vue
Lines 149 to 158 in 03a850f

```
149    @Watch('$route')  
150    protected handleChangeRoute(next: Route): void {  
151        if (this.visibleTabs.length === 0) {  
152            return;  
153        }  
154  
155        if (next.name === 'deployments') {  
156            this.selectInitialTab();  
157        }  
158    }
```

That handles cases such as navigation via breadcrumb or other such cases where the user somehow ends up on the Deployments Page Index without actually leaving the deployments page.

Finally, we want to update the `<v-tabs>` components to use routes rather than a click handler:

ui/src/components/DeploymentsPage/DeploymentsMenu.vue

Lines 48 to 57 in 03a850f

```
48     <v-tabs>
49       <v-tab
50         v-for="tab in visibleTabs"
51         :key="tab.serviceFeature"
52         :to="tab.route"
53         ripple
54       >
55         {{ tab.title }}
56     </v-tab>
57 </v-tabs>
```

Note that we no longer need to track `<v-tabs>` state. This is because `<v-tabs>` has internal state handling when being used with router routes.

Where To Go From Here

Breadcrumb

I didn't update the breadcrumb in this PR because there's not much to say there. You won't be able to directly copy/paste code for the breadcrumb from your PR and expect it to work with this code, but you can reference the code from your PR as a starting point.

The one thing you will want to do, though, is change this line:

```
diff --git i/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue w/src/components/Comm
index 2a0d6c9f..c039f3d3 100644
--- i/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue
+++ w/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue
@@ -112,8 +112,8 @@ import IGetServiceSummaryResponse from '@/interfaces/requests/IGetServiceSummary
   name: 'MainNavigationBreadcrumb',
 })
export default class MainNavigationBreadcrumb extends Vue {
-  get currentRouteName(): string {
-    return getCurrentRoute(this.$store);
+  get currentRouteName(): string | void {
+    return this.$route.name;
  }

  get isEachThingLoadedForCurrentRoute(): { [key: string]: boolean } {
```

You want that computed prop to return `this.$route.name` instead. Of course, that's going to require the computed prop's return type be changed to `string | void`, but that is to be expected and is more correct, if more annoying.

- joe 212750376 referenced this pull request 7 days ago
- created routes and urls, breadcrumb suffix, service changed to servic... #303

Closed
- 0 of 2 tasks complete
- 212750376 added some commits 7 days ago
- joe Merge remote-tracking branch 'origin/dev' into feature/breadcrumb_tra... f1bb133
- joe store/Deployments: Delete empty export 3a8a6af


joe

212750376 commented 7 days ago

Author Member


There were a lot of changes in `dev`. I've merged the latest and pushed it up to this branch. Things should now be better matched.

- 212750376 added some commits 6 days ago
- joe Merge branch 'dev' into feature/breadcrumb_trace_in_services-joe-recc... 96d7c5d


- 


joe

Merge branch 'dev' into feature/breadcrumb_trace_in_services-joe-recc... ...


✓ b4065be
- 


503133850


assigned 212750376 5 days ago
- 





Breadcrumb for service tabs added

✓ 52b64f7
- 



503152573 added `wip` and removed `do not merge yet` labels 2 days ago
- 



503152573 changed the title ~~EXAMPLE Feature: Update Deployments Page Tabs to Use Router~~
Update Deployments Page Tabs to Use Router 2 days ago
- 

joe

212750376 assigned 503152573 and unassigned 212750376 2 days ago

joe

212750376 commented 2 days ago

AuthorMember

NOTE: If this seems a bit circuitous, then you're right. As I said, the breadcrumb logic really needs to be ripped out and replaced.

So, we're running into an issue with the breadcrumb: when the user is on a route which is not the "index" route, the breadcrumb changes the Product, Component, and Environment to "(Loading (whatever)...)".

Okay, to start, how are those names computed?

Well, how about `selectedEnvironmentName` to start with? This is computed in the Computed Prop `get selectedComponentName()` on line 321:

ui/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue

Lines 333 to 343 in 52b64f7

```
333     get selectedEnvironmentName() {
334       if (!this.isSelectedEnvironmentLoaded || !this.selectedEnvironment) {
335         return '(Loading Environment...)';
336       }
337
338       if (!this.selectedEnvironment.name) {
339         return '(Unnamed Environment)';
340       }
341
342       return this.selectedEnvironment.name;
343     }
```

- We can see easily there that [line 335](#) is what yields the value `'(Loading Environment...)'`, so now we need to check what conditions lead to that.
- Looking at [line 334](#) above it we see that there are two things which can influence this: `this.isSelectedEnvironmentLoaded` and `this.selectedEnvironment`.

Alright, `this.isSelectedEnvironmentLoaded` is defined on line 294:

ui/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue

Lines 294 to 300 in 52b64f7

```
294     get isSelectedEnvironmentLoaded() {
295       const idInRoute = String(this.currentRouteParams.environmentId);
296       const idInStore =
297         this.selectedEnvironment && String(this.selectedEnvironment.id);
298
299       return idInRoute === idInStore;
300     }
```

while `this.selectedEnvironment` is defined on line 219:

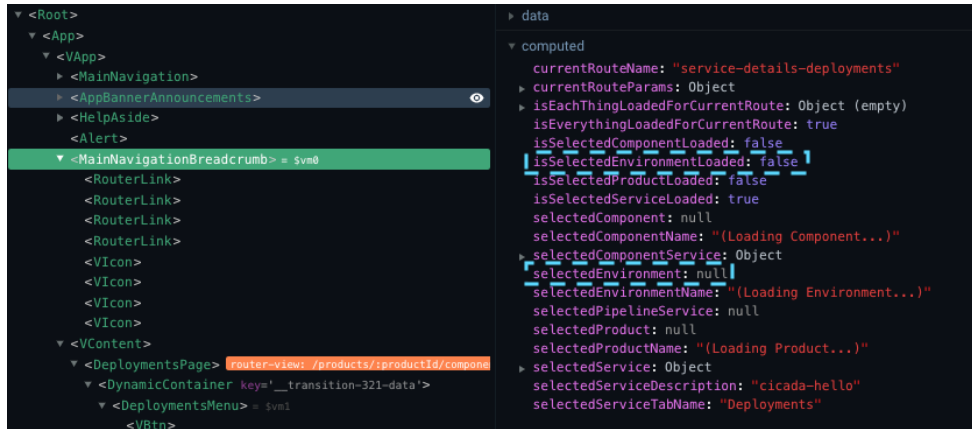
ui/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue

Lines 219 to 221 in 52b64f7

```
219     get selectedEnvironment() {
220       return getSelectedEnvironment(this.$store);
```

221 }

So, that's two things to check. Let's now take a look at those two computed props' current values in the Vue Dev Tools:



Hm. Both of those are not values that we want.

If we look at `get isSelectedEnvironmentLoaded()` we see that it depends on two other things:

- The Environment ID from the Route Params
- `this.selectedEnvironment`

One of those, the Environment ID from the Route Params, we're taking as a given. The other, `this.selectedEnvironment`, is something we still need to look at.

As seen from before, that's computed using `getSelectedEnvironment(this.$store);`, a getter-wrapper that wraps around `environments.getters.selectedEnvironment`:

[ui/src/store/Environments/environments.ts](#)

Lines 67 to 86 in 52b64f7

```
67     selectedEnvironment(
68       state: IEnvironmentsState,
69       stateGetters: any,
70       rootState: IRootState
71     ): IEnvironmentSummary | null {
72       // NOTE: Symptom of incorrect store organization: requested data should be managed
73       const rootStateCast = (rootState as any) as AppStoreState;
74       const { environmentId } = routerModule.getters.currentRouteParams(
75         rootStateCast.router
76       );
77       const { environments: environmentsCollection } = state.selectedComponent;
```

Oof, that's a bit of a doozy.

The short of it is:

- Get Environment ID from Current Route
- Get Environments of Selected Component from `state.selectedComponent`
- If both of those things are useful values, find the Environment in the Selected Component's Environments with the same ID as the Environment ID in the Current Route

What's happening when we navigate directly to a tab, then? Well, we know the the Environment ID is present in the route since our route is defined to have it and we're at a valid route (or at least we're assuming it's still valid) so that leaves the other part of that computation, the Environments of the Selected Component. (`state.selectedComponent.environments`)

We can see that by looking at that computation, there are 3 main cases which may lead to `null` being returned:

- `state.selectedComponent.environments` being `falsey` (which shouldn't happen because its type is defined to be an array of some sort).
- `environmentId` being `falsey`, which shouldn't happen in this particular case because we're on a defined route which has that ID in the path.
- `state.selectedComponent.environments` not containing an environment whose ID matches the Environment ID from the Route. This has two possible sub-causes:
 - i. `state.selectedComponent.environments` has environments, but none of them match by ID.
 - ii. `state.selectedComponent.environments` has no environments in it.

Alright, what populates that array? Two identical mutations (nice...):

[ui/src/store/Environments/environments.ts](#)
Lines 88 to 93 in 52b64f7

```
88   SET_SELECTED_COMPONENT_ENVIRONMENTS(  
89     state: IEnvironmentsState,  
90     payload: any  
91   ) {  
92     state.selectedComponent.environments = payload;  
93   },
```

and

[ui/src/store/Environments/environments.ts](#)
Lines 100 to 105 in 52b64f7

```
100  SET_COMPONENT_ENVIRONMENTS(  
101    state: IEnvironmentsState,  
102    payload: IEnvironmentSummary[]  
103  ) {  
104    state.selectedComponent.environments = payload;  
105  },
```

But only one of these, via the wrapper named `commitSetSelectedComponentEnvironments`, is actually used in any relevant Actions:

[ui/src/store/Environments/environments.ts](#)
Lines 108 to 123 in 52b64f7

```
108  async fetchComponentEnvironments(  
109    environmentsContext: IEnvironmentsContext,  
110    componentId: number  
111  ): Promise<void> {  
112    try {  
113      const environmentDetails = (await axios.get(  
114        `/api/cicada/components/${componentId}/summary`  
115      )).data.environments;  
116      commitSetSelectedComponentEnvironments(  
117        environmentsContext,  
118        environmentDetails  
119      );
```

That gets exported via the wrapper `fetchComponentEnvironments`. So what all uses that wrapper? Well, a number of things, but one of them is `MainNavigationBreadcrumb.vue` itself, so let's check that first. Looks like this dispatcher is called on line 405, in the watch handler named `protected handleChangeCurrentRouteName()`:

[ui/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue](#)
Lines 399 to 407 in 52b64f7

```
399    if (  
400      'isSelectedEnvironmentLoaded' in this.isEachThingLoadedForCurrentRoute &&  
401      !this.isSelectedEnvironmentLoaded  
402    ) {  
403      const componentId = Number(this.currentRouteParams.componentId);  
404      if (Number.isFinite(componentId)) {  
405        fetchComponentEnvironments(this.$store, componentId);  
406      }  
407    }
```

That itself seems to check `this.isEachThingLoadedForCurrentRoute`, specifically `'isSelectedEnvironmentLoaded'` in `this.isEachThingLoadedForCurrentRoute`. That is, `this.isEachThingLoadedForCurrentRoute` is controlling if some request is dispatched.

So, now we need to look at `this.isEachThingLoadedForCurrentRoute`:

`ui/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue`
Lines 153 to 196 in 52b64f7

```
153     get isEachThingLoadedForCurrentRoute(): { [key: string]: boolean } {
154       switch (this.currentRouteName) {
155         case 'products':
156           return {};
157
158         case 'components':
159           return {
160             isSelectedProductLoaded: this.isSelectedProductLoaded,
161           };
162
163         case 'environments':
164           return {
```

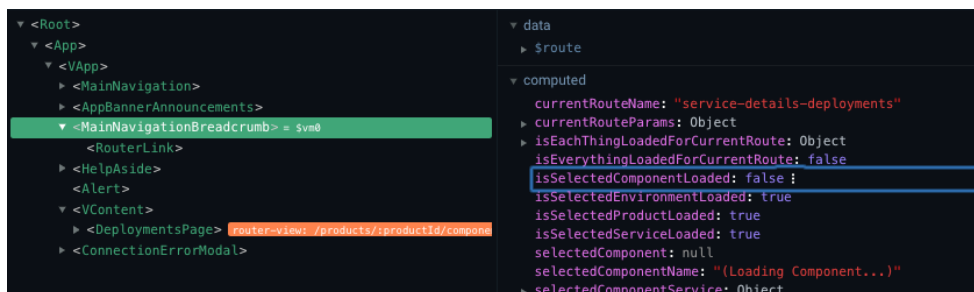
Reading through that, we see there's a case for `'deployments'` but not for any of the tab subroutes. Well, seems like a good place to put them, then. Let's try that and see what happens.

NOTE: I had to change the `v-if` because it could try to call `.match()` on `undefined` under certain conditions.

```
diff --git i/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue w/src/components/Comm
index 7552f567..21ec9f00 100644
--- i/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue
+++ w/src/components/Common/MainNavigation/MainNavigationBreadcrumb.vue
@@ -61,7 +61,7 @@
     </template>
   </span>
   <span
-    v-if="currentRouteName.match(/service-details-/)"
+    v-if="hasNameInMatchedRoutes('deployments')"
     class="breadcrumb-link-group"
   >
     <router-link to="/" class="breadcrumb-link">
@@ -174,6 +174,10 @@ export default class MainNavigationBreadcrumb extends Vue {
   };




   case 'deployments':
+  + case 'service-details-deployments':
+  + case 'service-details-logs':
+  + case 'service-details-config':
+  + case 'service-details-events':
   return {
     isSelectedProductLoaded: this.isSelectedProductLoaded,
     isSelectedComponentLoaded: this.isSelectedComponentLoaded,
```



That does cause it to load `this.selectedEnvironment`, which is good. However, we're still missing `this.selectedComponent`, and in fact we can see that `this.isSelectedComponentLoaded` is still `false`:





Which I guess means we need to conditionally request *that*, too. Ugh.

The above change is fine to start with, but before going any further than that it would be best to merge the latest `dev` to this branch. The Breadcrumb has been getting touched by a number of other branches recently, so we need to make sure we have the latest changes to avoid any more merge conflicts. There's already one in `DeploymentsMenu.vue` and that's bad enough as is.

-  503152573 added some commits 23 hours ago
-  Merge branch 'dev' into feature/breadcrumb_trace_in_services-joe-recc... ...

77b3337
-  fix: bookmarking service tabs

✓ cb11330
-  Revert "fix: bookmarking service tabs" ...

✓ 6faeb83