

# From Puppetry to Pull Requests

Write better pull requests with less effort  
by adaptively reusing puppetry principles.

Puppetry -> Writing (fiction) -> Pull Requests

when you attend  
a talk about puppetry

and the speaker doesn't have any puppets



# Focus

# Focus indicates thought

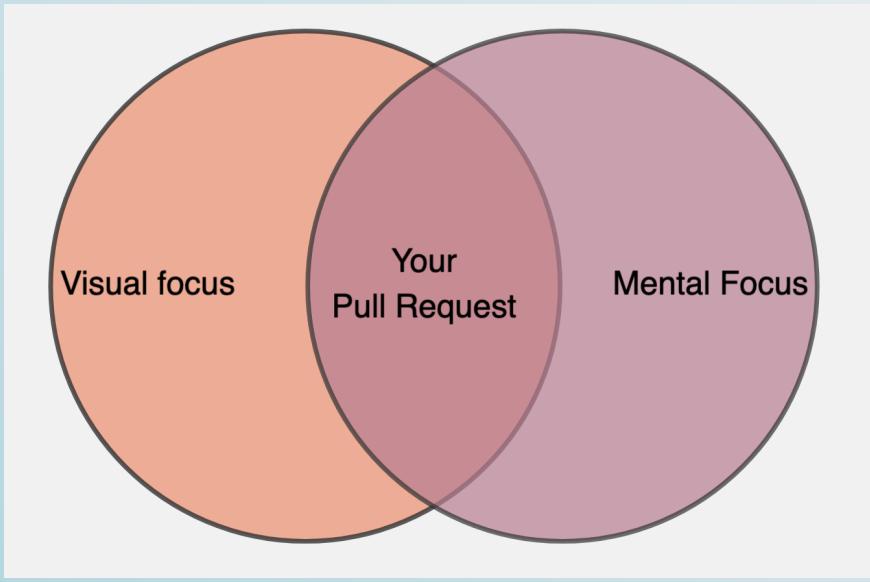


# One thing at a time





# Mixed use



**Dearest Reviewer,**

Credit: <https://github.com/sbeckeriv>

# This is how things are.

CN-2106: Simplify paid chat ui #25747

Merged percyhanna merged 5 commits into master from CN-2106-simplify-paid-chat-ui on Mar 10

Conversation 1 Commits 5 Checks 0 Files changed 2

ZempTime commented on Mar 10 · edited ...

Description

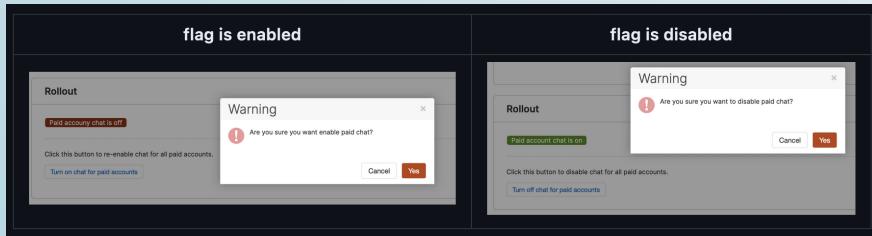
CS have been using this a lot more recently and keeps getting confused about what the current state is and what is going to happen when they click it

A screenshot of a GitHub pull request page. The title is "CN-2106: Simplify paid chat ui #25747". A purple button indicates it is "Merged". Below the title, it says "percyhanna merged 5 commits into master from CN-2106-simplify-paid-chat-ui on Mar 10". There are navigation links for "Conversation 1", "Commits 5", "Checks 0", and "Files changed 2". A comment from "ZempTime" is shown, dated Mar 10, with an edit history indicator. The comment text reads: "CS have been using this a lot more recently and keeps getting confused about what the current state is and what is going to happen when they click it".

# This is how we'd like them to be.

Rephrase the UI to remove double negatives and make it clear what the result is.

# Once merged, here's how things will be different.



## p.s.

The feature flag "disables." Feature flag on = chat off, feature flag off = chat on. Reverse button colors and use affirmative messaging to better reflect this.

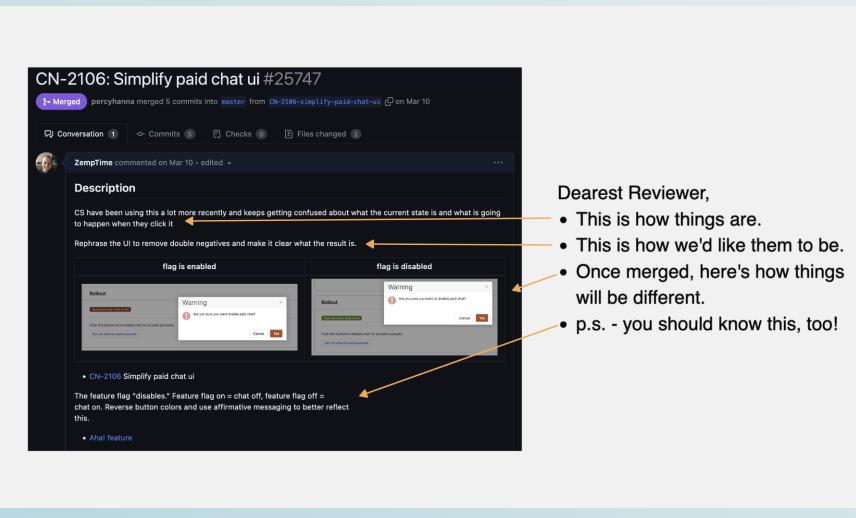
# Dearest Reviewer,

This is how things are.

This is how we'd like them to be.

Once merged, here's how things will be different.

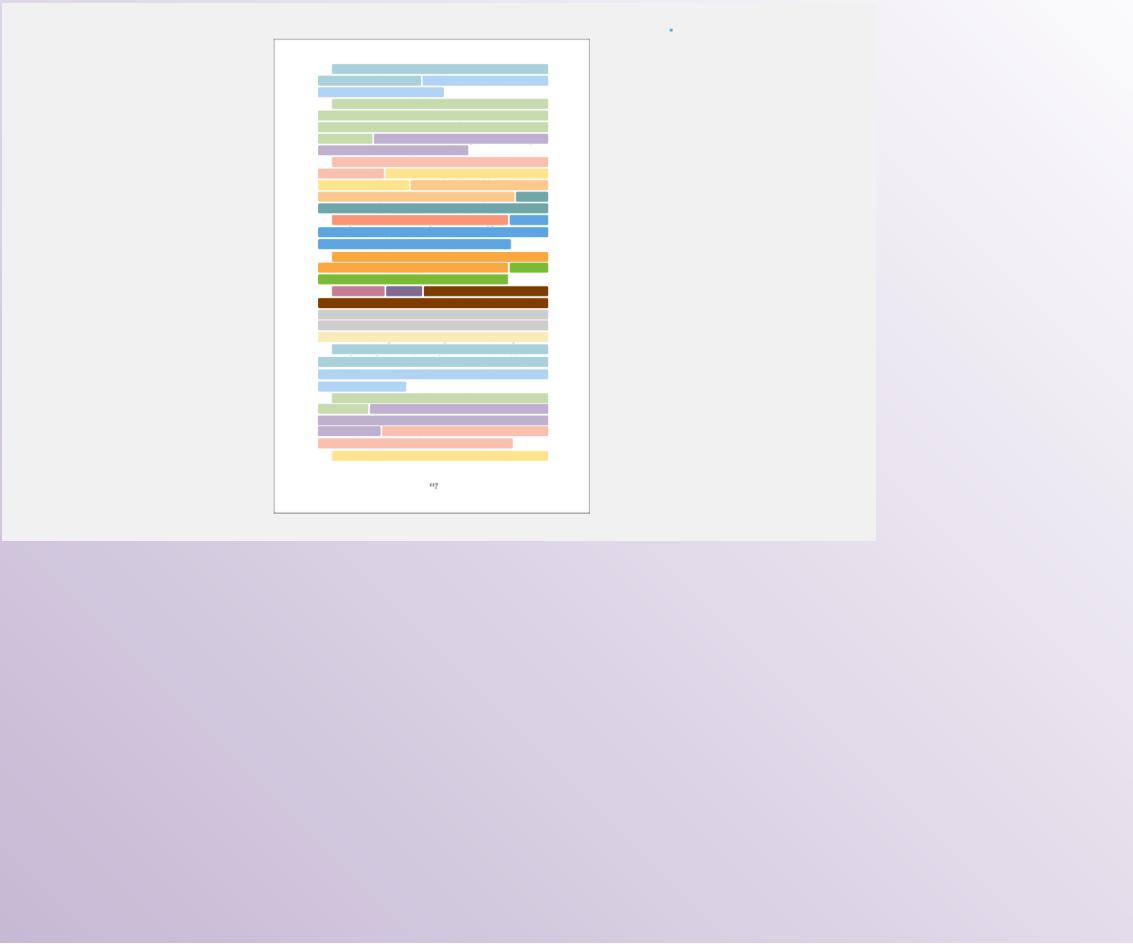
p.s. - you should know this, too!



+55,073 -1,463 ■■■■■

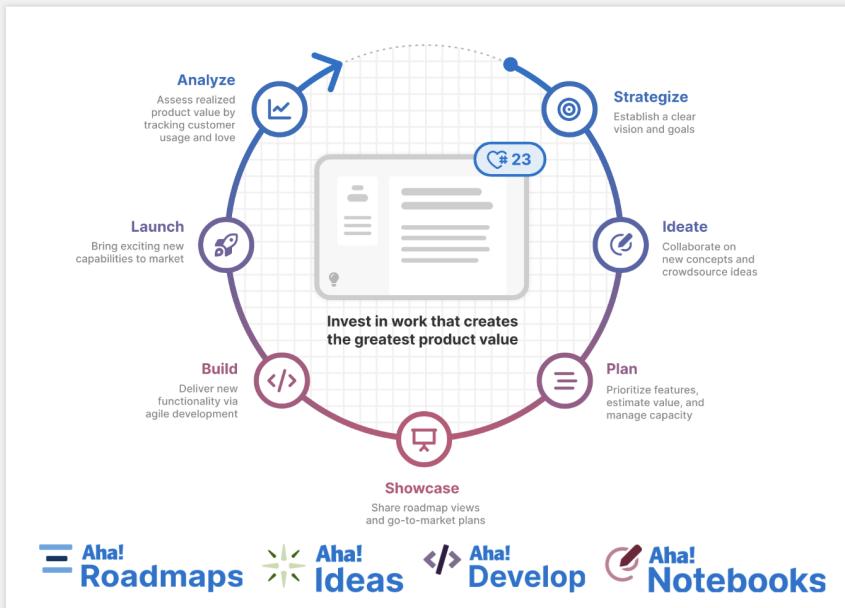
Breath





+55,073 -1,463 ■■■■■

# Aha!



# Start large scale, move finer scale

DEVOPS-1800: Reduce size of all webhook payloads by 50%

We enable parameter wrapping on all JSON requests:

[aha-app/config/initializers/wrap\\_parameters.rb](#)

Lines 7 to 9 in [9c0430d](#)

```
7   ActiveSupport.on_load(:action_controller) do
8     wrap_parameters format: [:json]
9   end
```

# Start large scale, move finer scale

According to the docs for `ActionController::ParamsWrapper` :

You can send parameters like this:

```
{"name": "Konata"}
```

And it will be wrapped into a nested hash with the key name matching your parameter name. If you're posting to UsersController, your new params hash will look like this:

```
{"name" => "Konata", "user" => {"name" => "Konata"}}
```

# Start large scale, move finer scale

For `Api::V1::WebhooksController`, each request contains a potentially large JSON object. With parameter wrapping enabled, the entire webhook payload is duplicated in a nested param named `webhook`. So we end up with a `params` structure something like:

```
{  
  webhookEvent: "jira:issue_updated",  
  timestamp: 123456789,  
  issue: {  
    id: 42,  
    key: "ABC-123",  
    fields: [  
      # Potentially huge data structure  
    ]  
  },  
  webhook: {  
    webhookEvent: "jira:issue_updated",  
    timestamp: 123456789,  
    issue: {  
      id: 42,  
      key: "ABC-123",  
      fields: [  
        # Another copy of the same potentially huge data structure  
      ]  
    }  
  }  
}
```

# Start large scale, move finer scale

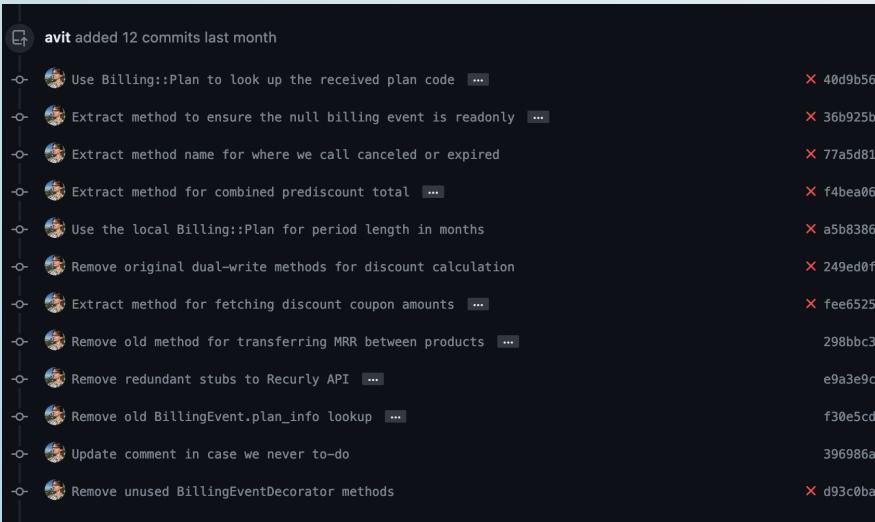
The nested `webhook` hash would then be passed along to Kafka and Resque as part of the payload, even though nothing depends on it. This created performance overhead at every step of webhook processing for JSON serialization, payload compression, network I/O, Redis memory, Kafka disk, etc.

# Start large scale, move finer scale

As far as I can tell, this has always been the case. The `wrap_parameters` initializer was introduced in the very first commit [75aea13](#) and has not changed since.

I have no idea what other code might be implicitly depending on parameter wrapping, but I am confident that we do not need it for webhooks. I disabled parameter wrapping for Jira Connect webhooks, too.

# Breathe evenly (by commit)



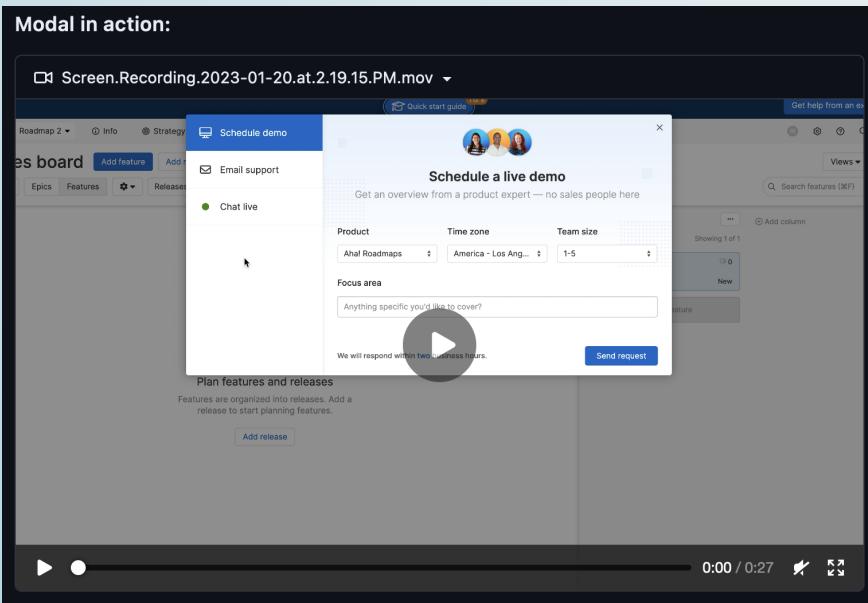
avit added 12 commits last month

- Use Billing::Plan to look up the received plan code ... 40d9b56
- Extract method to ensure the null billing event is readonly ... 36b925b
- Extract method name for where we call canceled or expired 77a5d81
- Extract method for combined prediscount total ... f4bea06
- Use the local Billing::Plan for period length in months a5b8386
- Remove original dual-write methods for discount calculation 249ed0f
- Extract method for fetching discount coupon amounts ... fee6525
- Remove old method for transferring MRR between products ... 298bbc3
- Remove redundant stubs to Recurly API ... e9a3e9c
- Remove old BillingEvent.plan\_info lookup ... f30e5cd
- Update comment in case we never to-do ... 396986a
- Remove unused BillingEventDecorator methods ... d93c0ba

# Breathe in videos, pictures, and tables

GR-102: Redesign demo signup form in-app #25000

# Breathe in videos, pictures, and tables



# Breathe in videos, pictures, and tables

**Modal variations:**

	roadmaps	ideas	develop
demo			
email			
chat			
chat disabled			

Breath

# Muscle

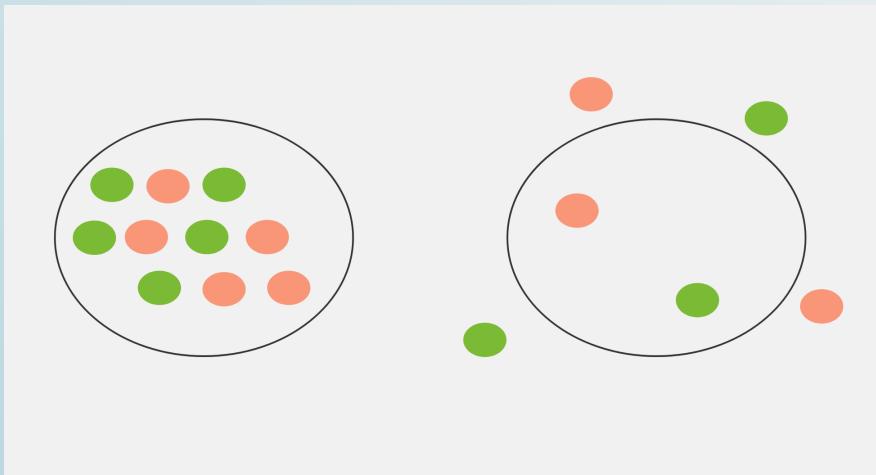


Marching in close formation vs explosive fireballs?

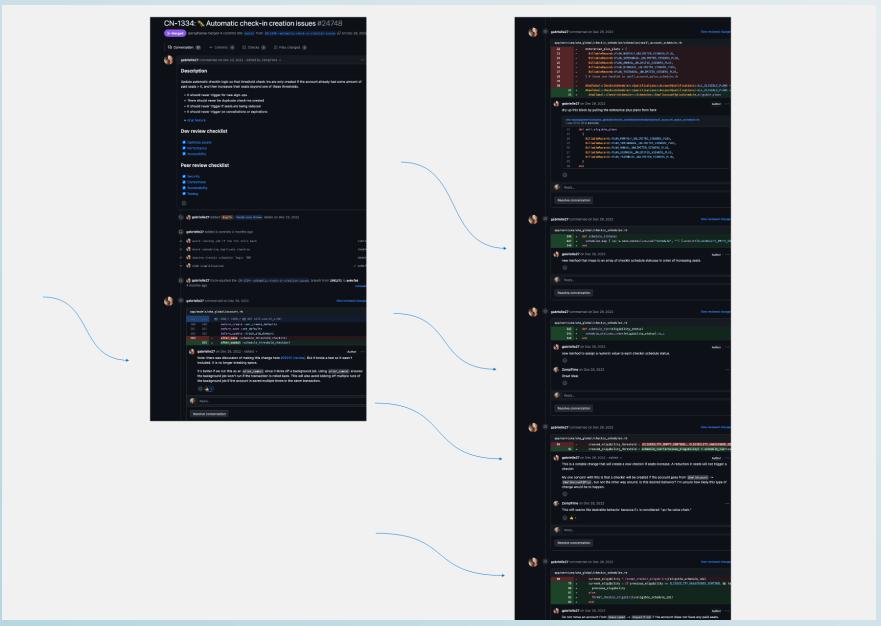
Right... 😳



# Coherence



# Details in comments



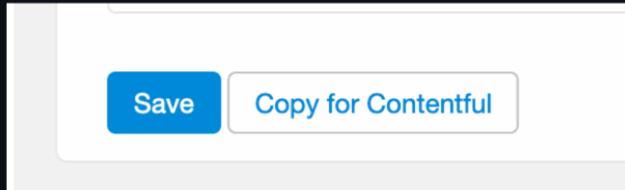
# <details></details>

```
<details>
  <summary> Title </summary>

  contents
</details>
```

Need to copy content from admin into contentful. Adds a button to simplify the process, providing some clipboard output which can be pasted in the json props area of templates like [this one](#)

To be used with [aha-app/aha-gatsby#1185](#)



▼ Sample of copied output

```
{
  "id": "7213013120322134616",
  "name": "SWOT analysis",
  "slug": "swot",
  "categories": [
    "Research"
  ],
  "summary": "{\"object\":\"value\",\"document\":{\"object\":\"document\",\"data\":{},\"nodes\":[{\"object\":\"value\"}]}}",
  "details": "{\"object\":\"value\",\"document\":{\"object\":\"document\",\"data\":{},\"nodes\":[{\"object\":\"value\"}]}}",
  "bestPractices": "{\"object\":\"value\",\"document\":{\"object\":\"document\",\"data\":{},\"nodes\":[{\"object\":\"value\"}]}}"
}
```

# Muscle

# Meaningful movement

# Movement



# Jar-Jar Binks



# Tragedy



# Darth Jar Jar

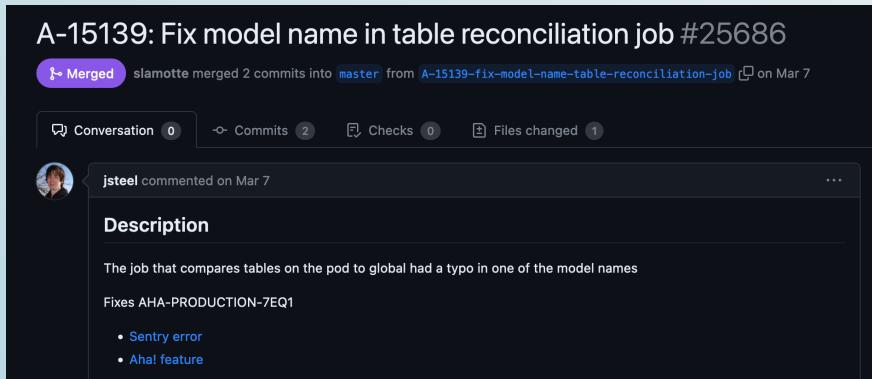


# Meaningful movement

...in pull requests

# Meaningful movement

## By omission



# Meaningful movement

By inclusion

DEVOPS-1805: Handle ADO webhooks without an iteration path #25834

Merged · avit merged 1 commit into `master` from `DEVOPS-1805-handle-ado-webhooks-without-iteration-path` 3 weeks ago

Conversation 0 · Commits 1 · Checks 0 · Files changed 2 · ...

maeve commented 3 weeks ago · edited

### Description

In #25834, we introduced logic to partition ADO webhooks based on the iteration path. In Azure DevOps Services, the iteration is a required field for all work items and webhooks always include an iteration path.

However, it appears there are two accounts that receive webhook payloads without an iteration path ([example log](#)). They may be integrating with an older version of Azure DevOps Server. We have temporarily disabled Kafka for webhooks for these accounts, so they are not experiencing a disruption while we work on the fix.

We have to accommodate many different webhook payload shapes, which may change over time. The producer logic should gracefully handle missing attributes in the payload and fall back to using the integration ID as the message key. If a webhook payload is ultimately not processable, the remote identifier consumer will detect that and log it appropriately (including user-friendly integration messages).

- [Sentry error](#)
- [Aha! feature](#)

Puppetry -> Writing (fiction) -> Pull Requests

# Puppetry

Principle	Puppetry	Writing	Pull Requests
<b>Focus</b>	Thought	One thing at a time	Dearest reviewer
<b>Breath</b>	Emotion	Sentence length	Matching info structures
<b>Muscle</b>	Trust	Believability	Detailing with <details> or comments
<b>Meaningful Movement</b>	Efficiency	Knowledge beyond action	Presence or omission of info

# Special Thanks

Mary Robinette



Focus  
Breath  
Muscle  
Meaningful Movement

feedback? questions? compliments? insults? threats?

@ZempTime

