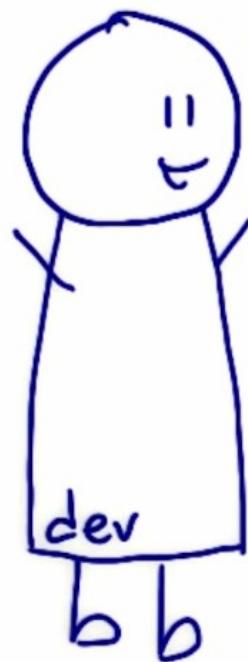


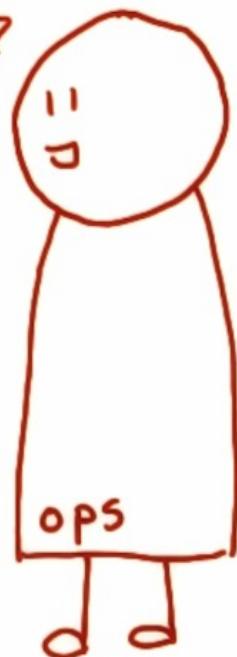
# The Purpose of DevOps

... is the purpose of your organization!

Let's do stuff!



What stuff?

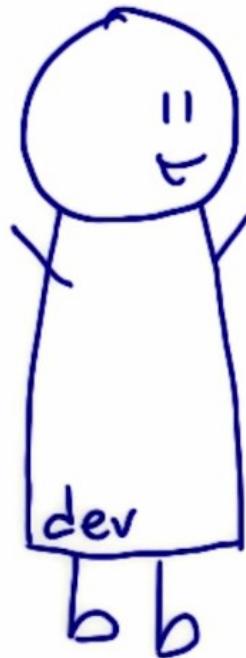


# The Purpose of DevOps

... is the purpose of your organization!

Let's do stuff!

Create value!



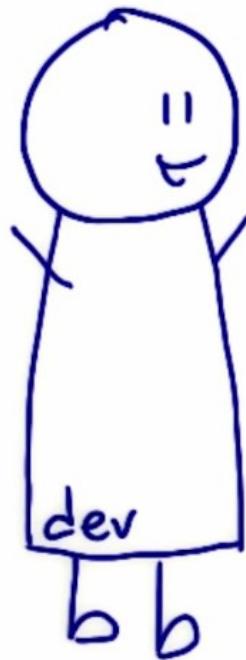
What stuff?



# The Purpose of DevOps

... is the purpose of your organization!

Let's do stuff!



What stuff?

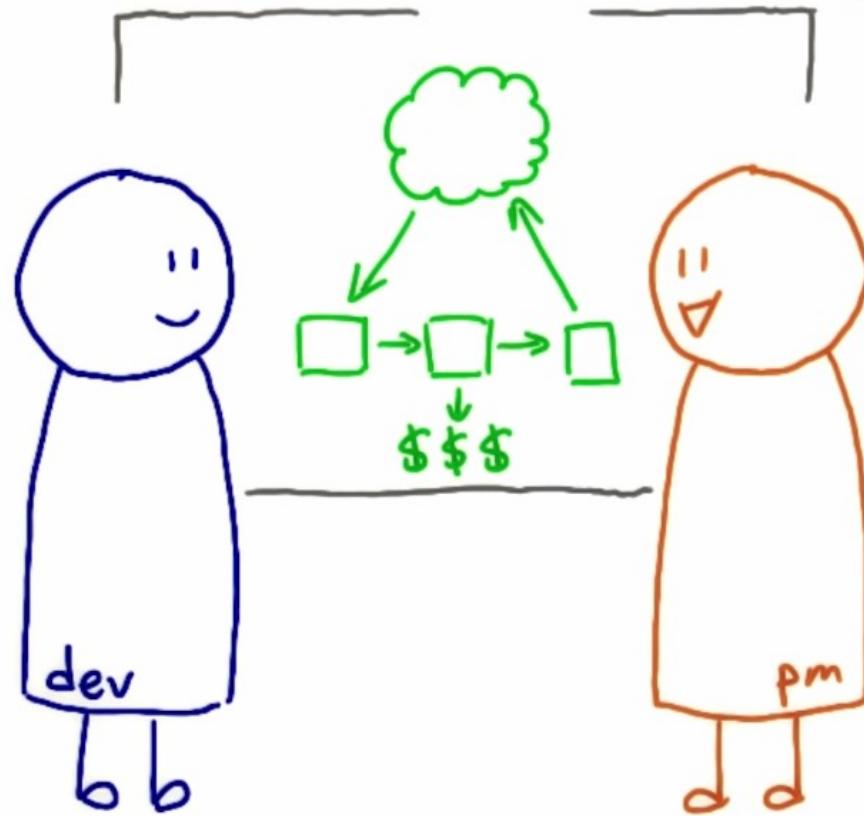


Throw a bunch of money away  
And waste time  
And make ourselves all miserable...



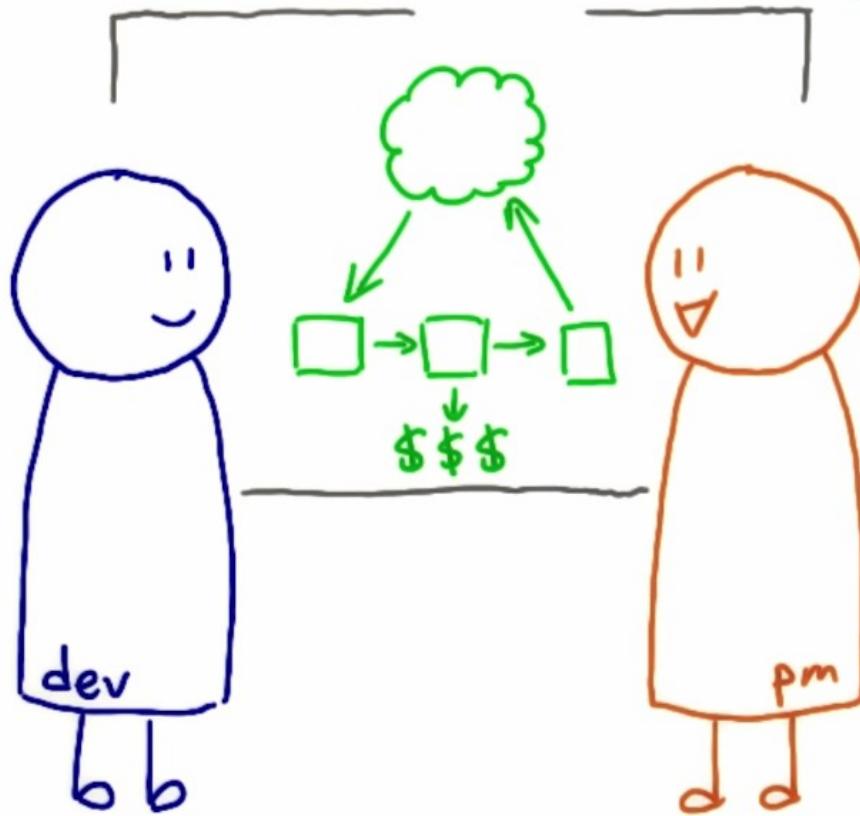
# The Purpose of DevOps

... is the purpose of your organization!



# The Purpose of DevOps

... is the purpose of your organization!



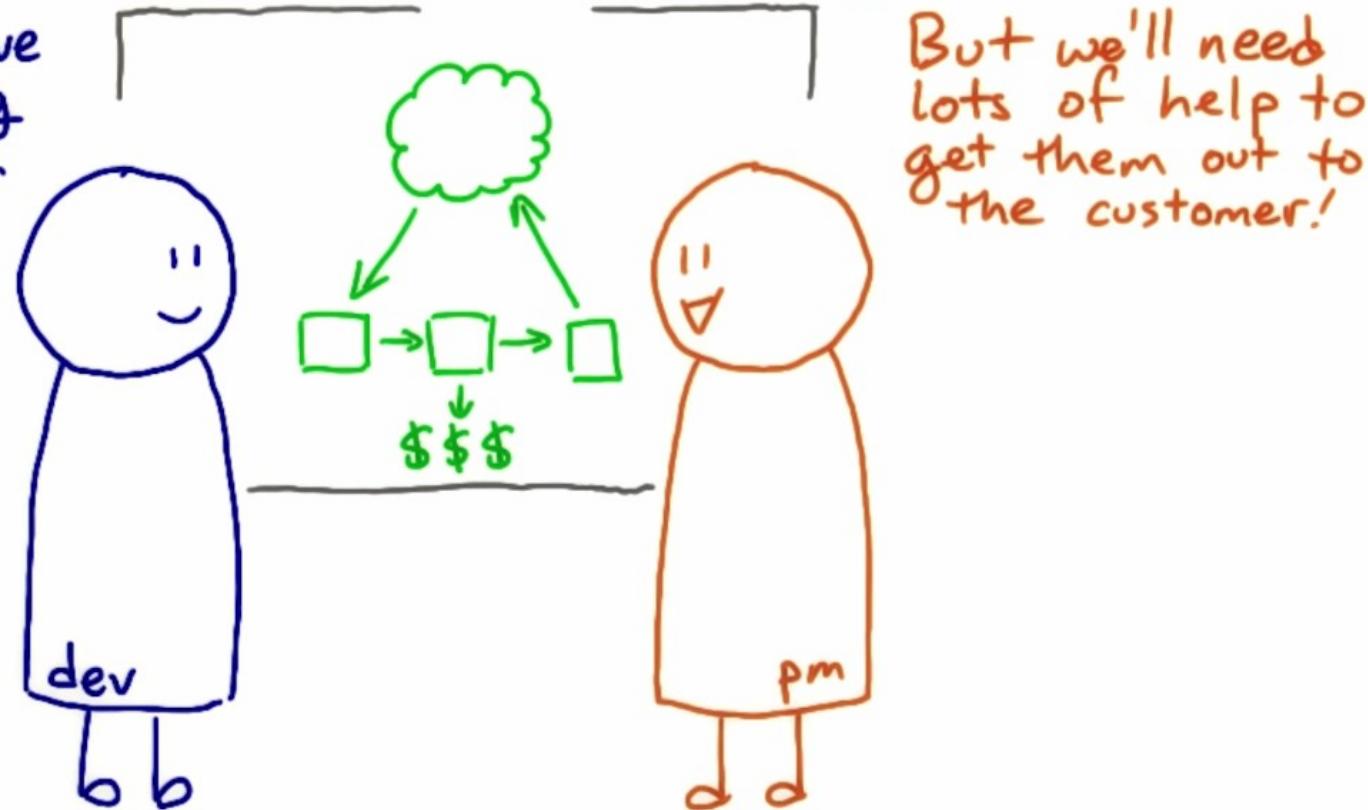
Hey, I don't  
pay for vaporware.  
Show me the value.



# The Purpose of DevOps

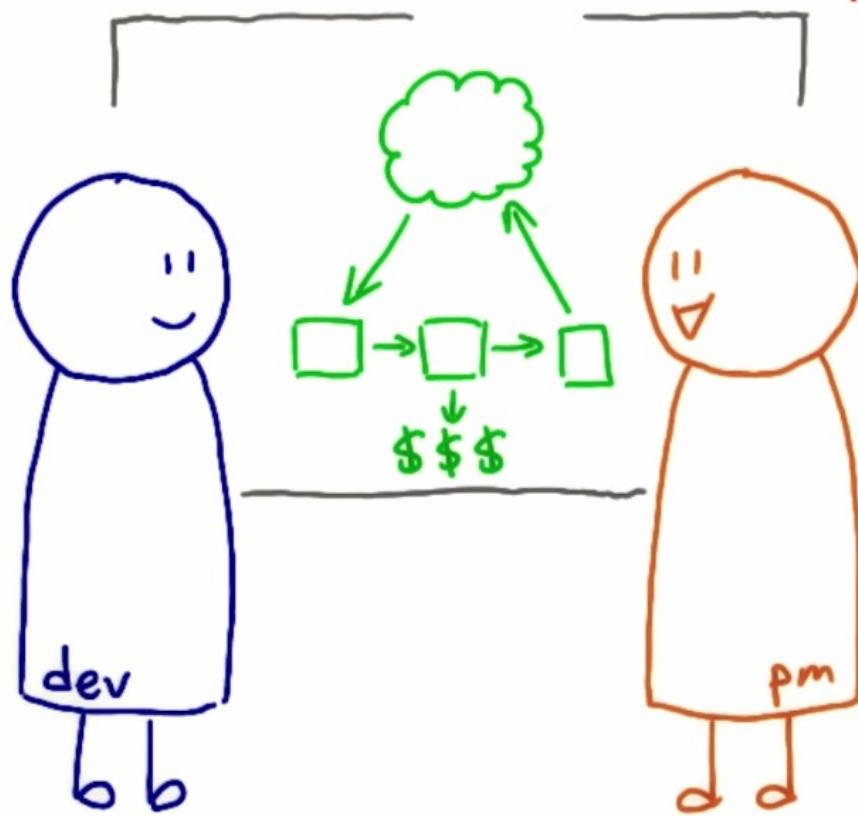
... is the purpose of your organization!

We have  
some big  
ideas...

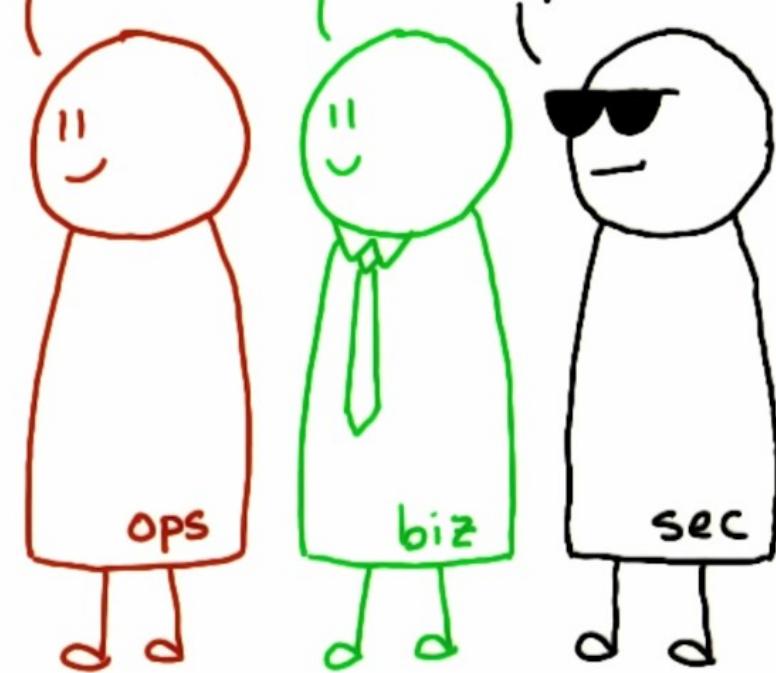


# The Purpose of DevOps

... is the purpose of your organization!



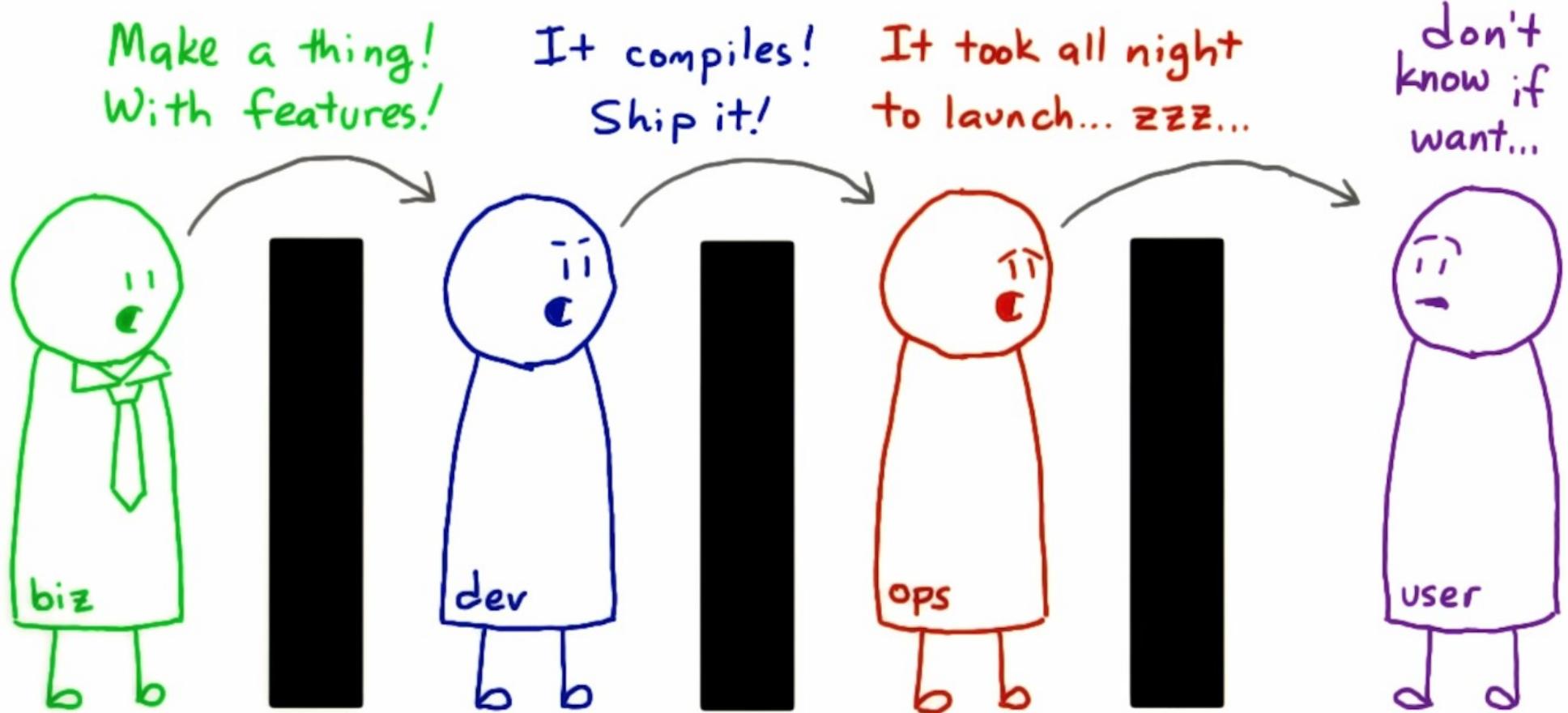
How can we help?



# The Purpose of DevOps

... but if we're not careful ...

# NOT The Purpose of DevOps



# ~~REALLY NOT~~ The Purpose of DevOps

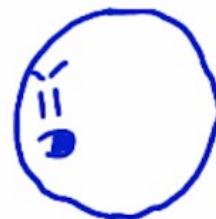
Hey, why does  
your site crash?



Lazy technical  
staff!



Unrealistic  
project deadlines!



Inadequate  
server budget!



Sloppy coding  
practices!

# ~~REALLY NOT~~ The Purpose of DevOps



Unhappy

Slow + ineffective at delivery

Can't learn from failures

Hard to make changes

Operational problems stack up

# The Purpose of DevOps

Hey wait... we both know stuff...

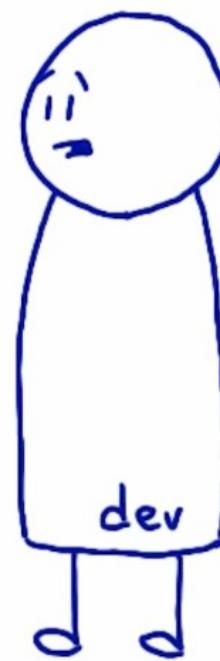
Yeah! How about we gang up  
and rule the world?

Or we could just make  
our services run better.

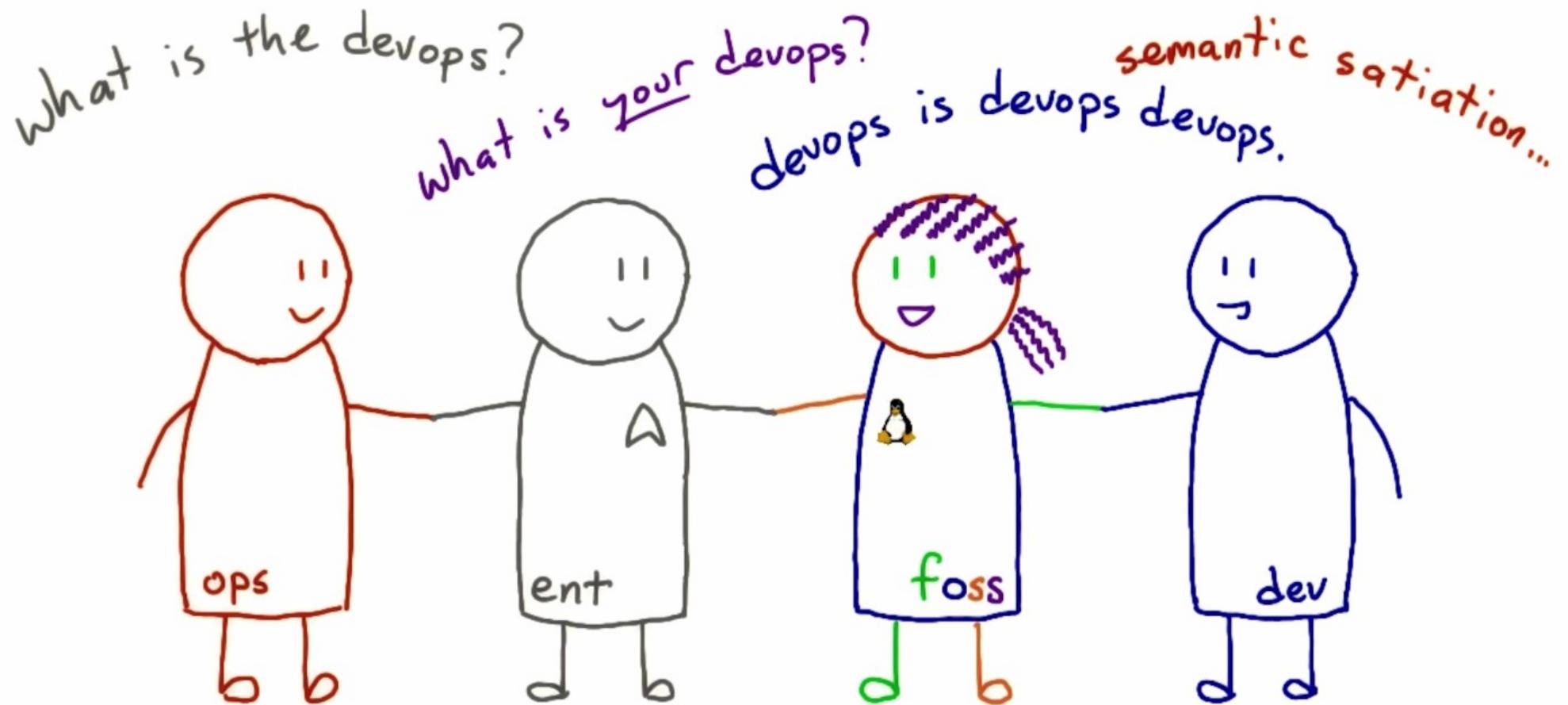
And launch more  
features?

Yeah.

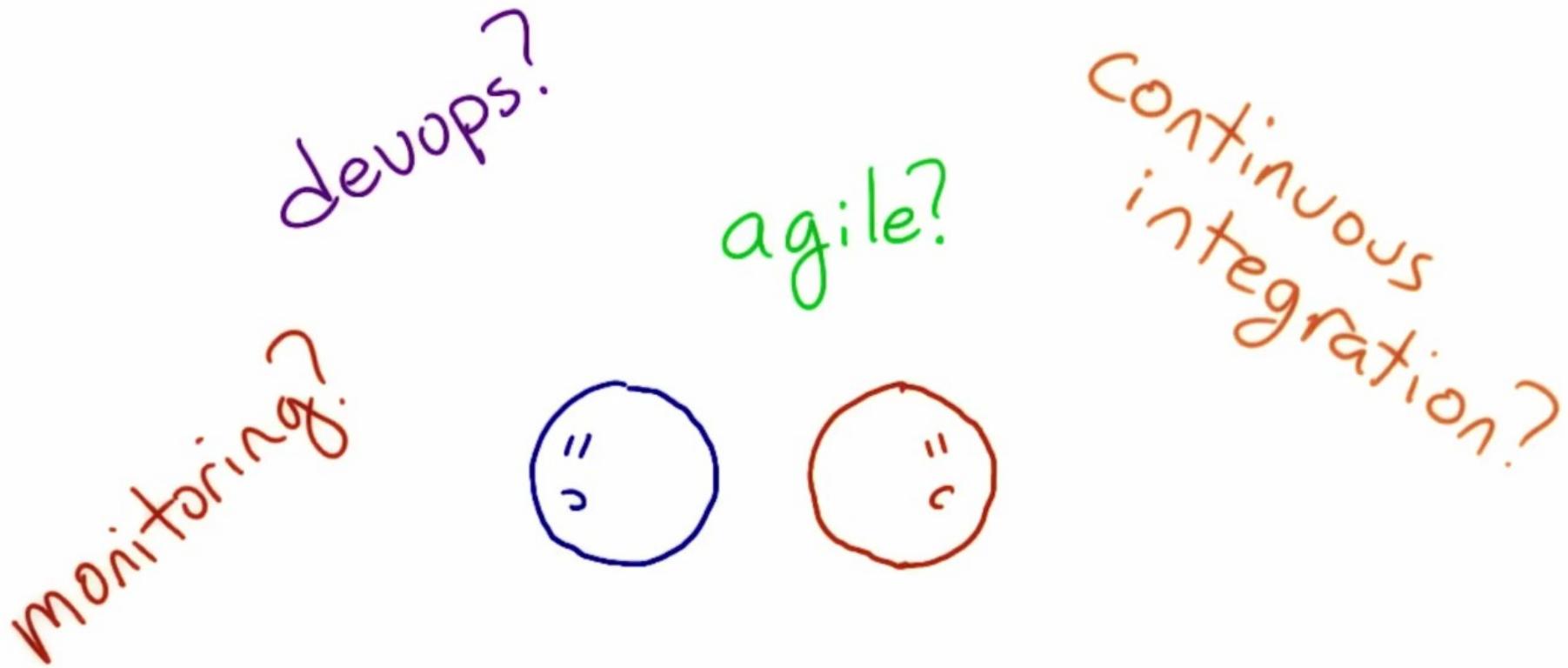
Okay!



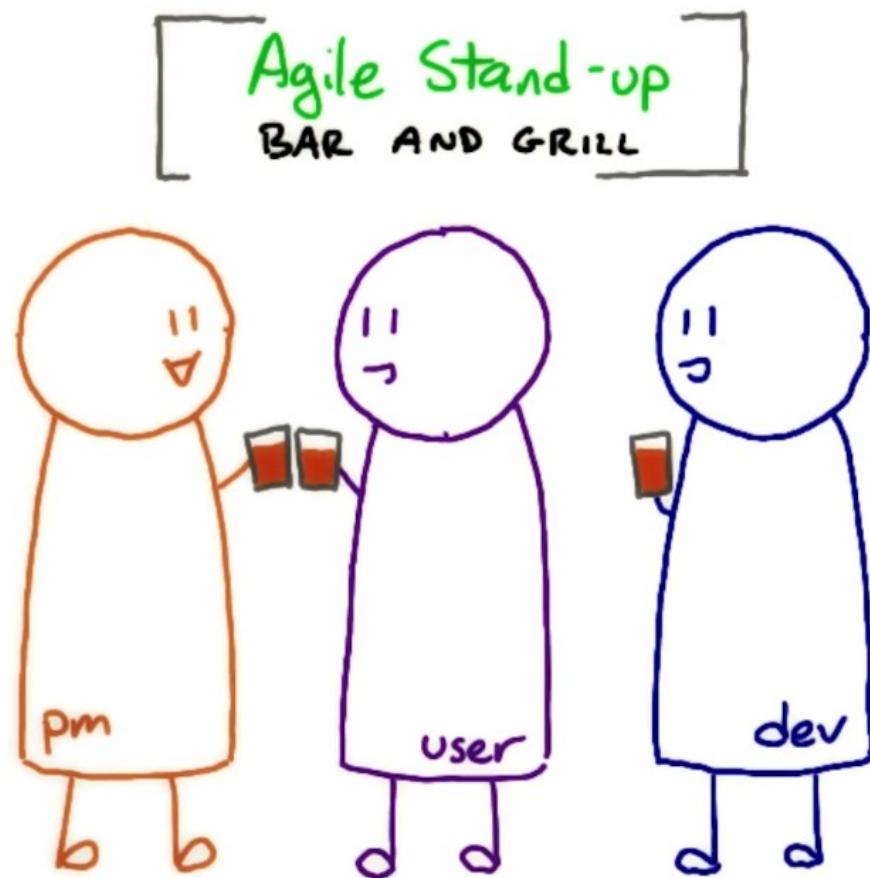
# The Purpose of DevOps



# Software Development Lifecycle



# Software Development Lifecycle



# Software Development Lifecycle

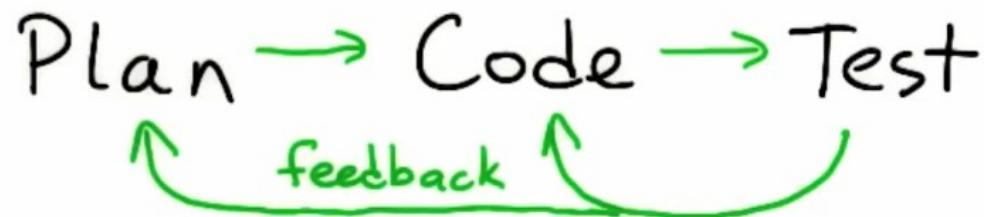
Agile Development

Plan → Code → Test



# Software Development Lifecycle

## Agile Development



# Software Development Lifecycle

## Agile Development

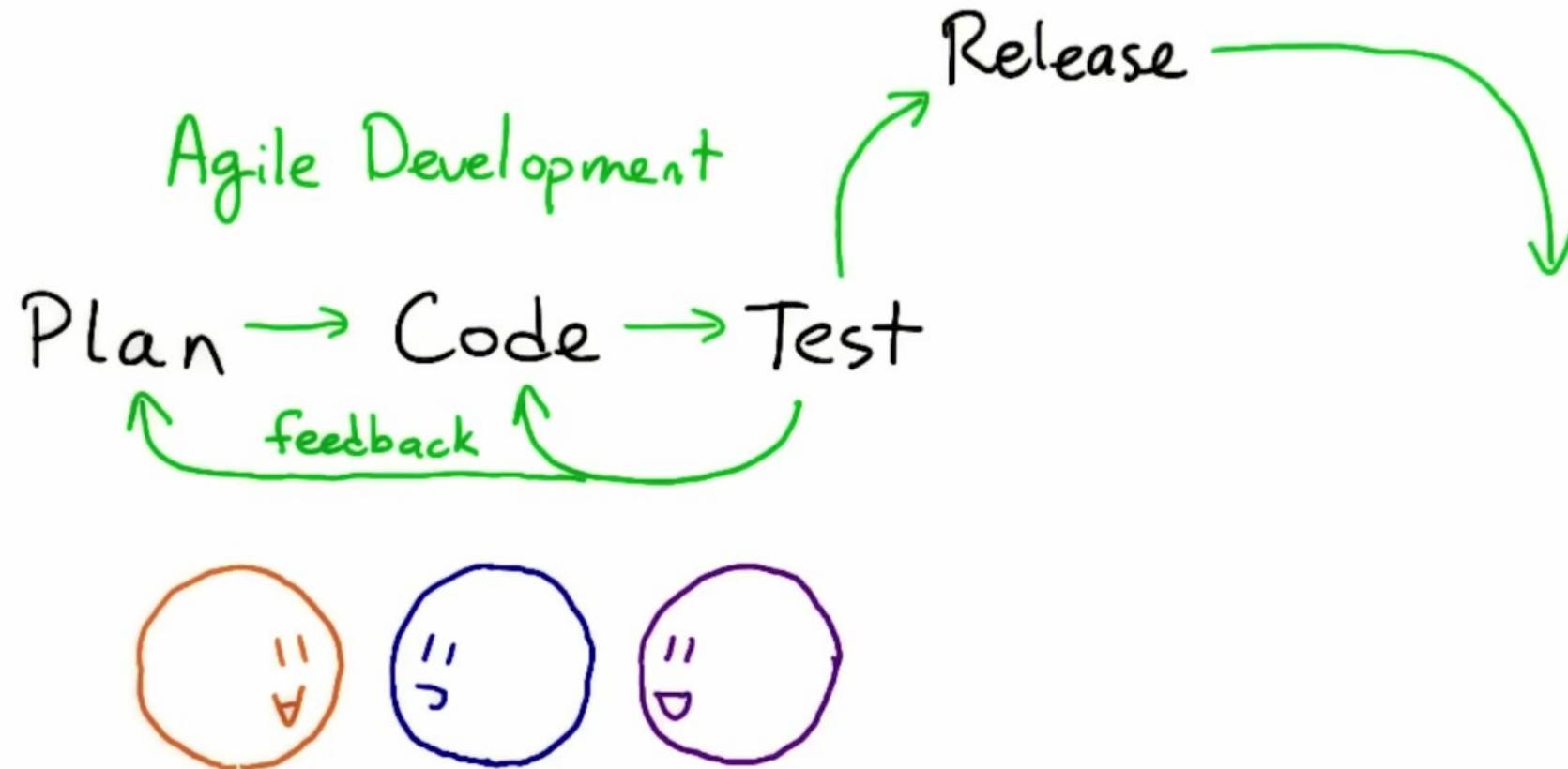
Plan → Code → Test



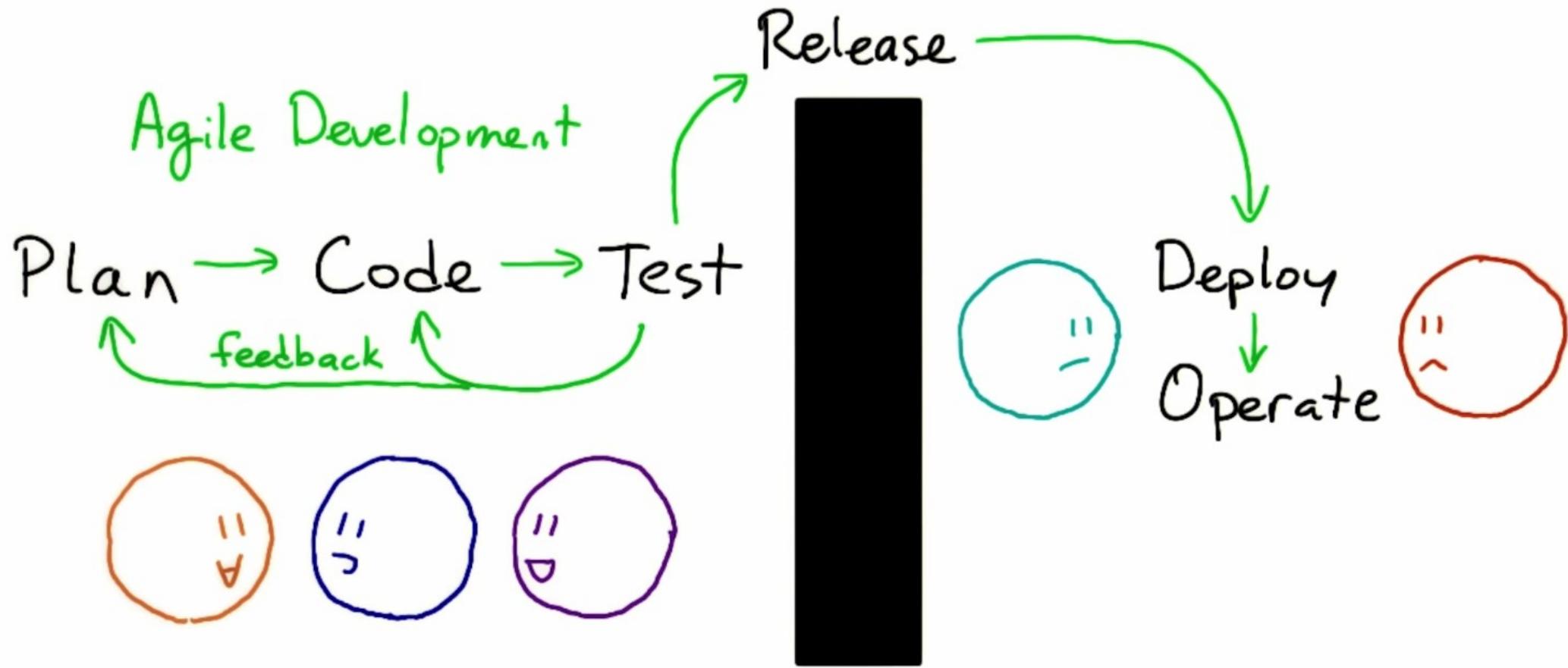
unit testing  
integration testing  
user acceptance testing



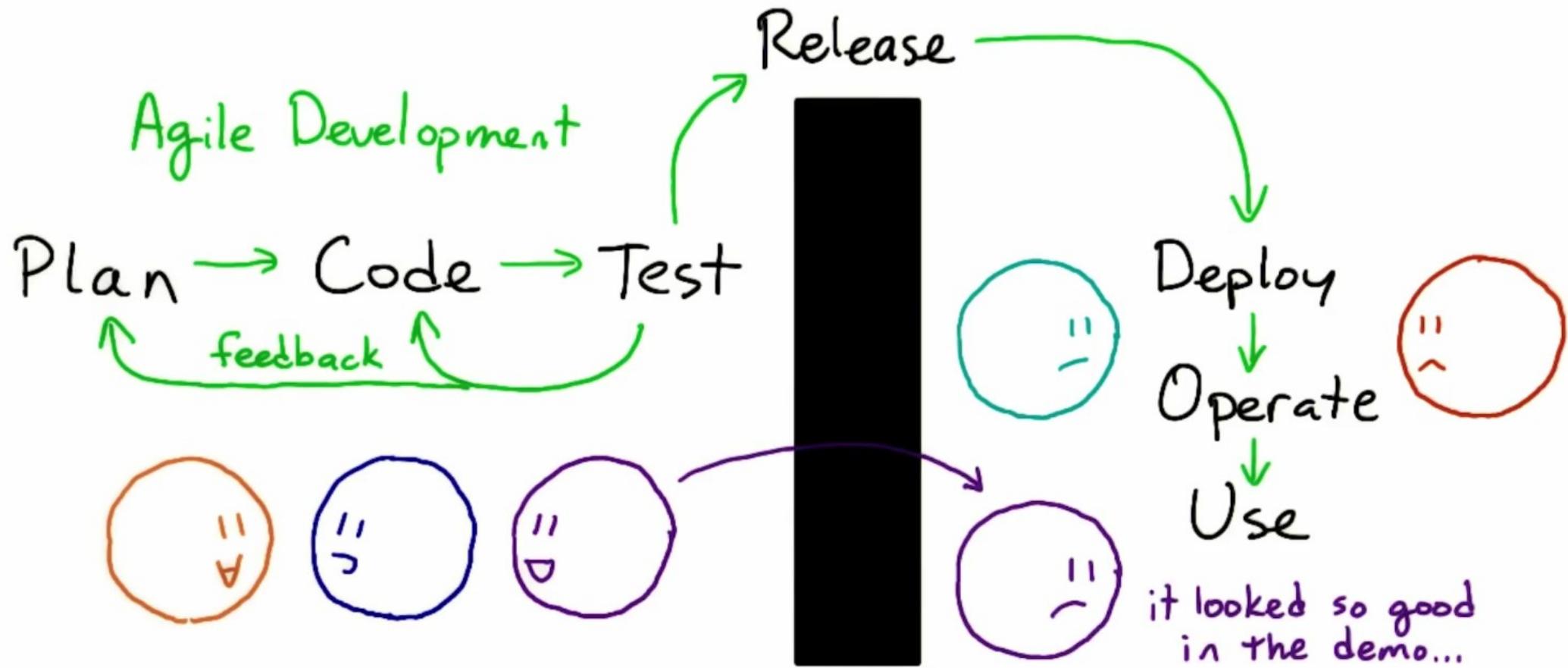
# Software Development Lifecycle



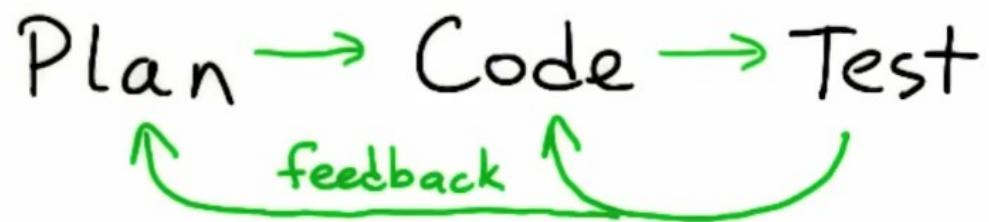
# Software Development Lifecycle



# Software Development Lifecycle

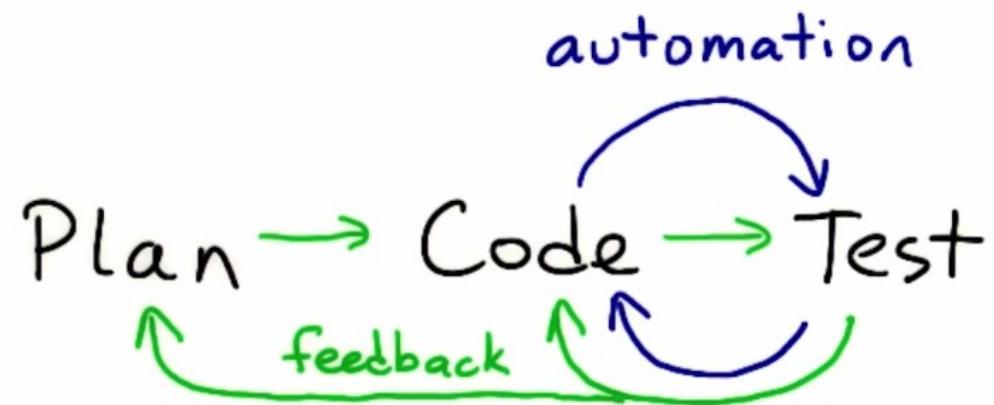


# Software Development Lifecycle



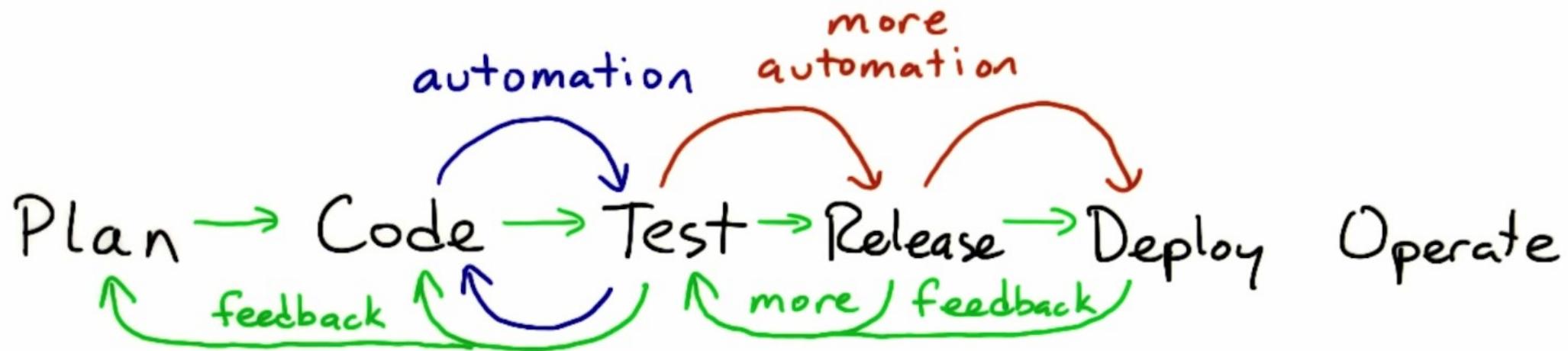
Agile Development

# Software Development Lifecycle



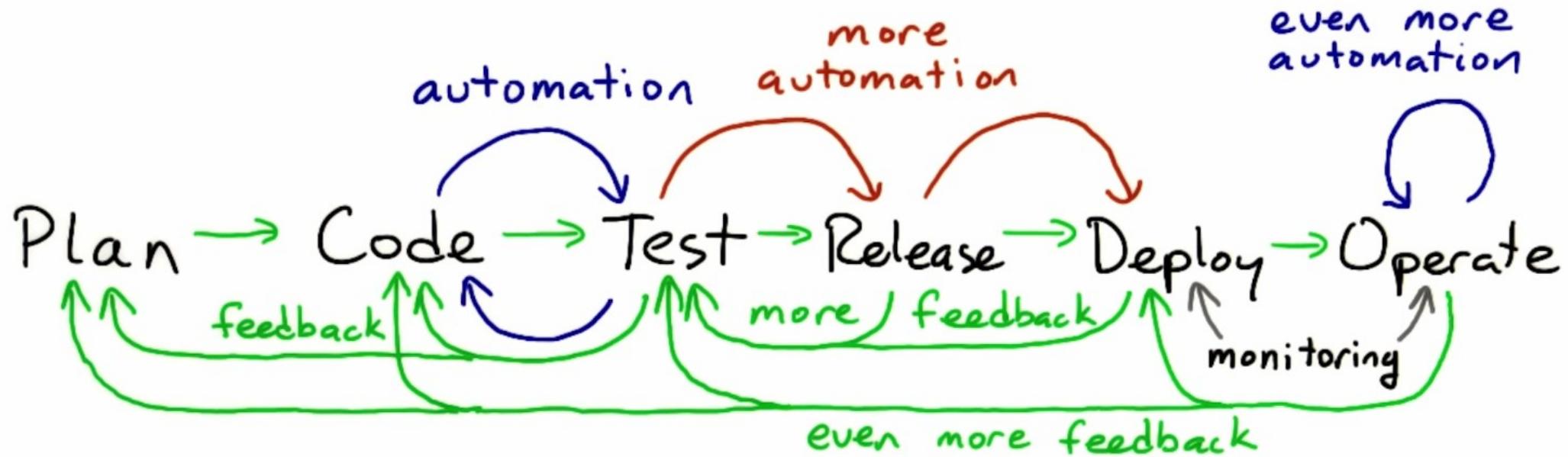
Agile Development Continuous Integration

# Software Development Lifecycle



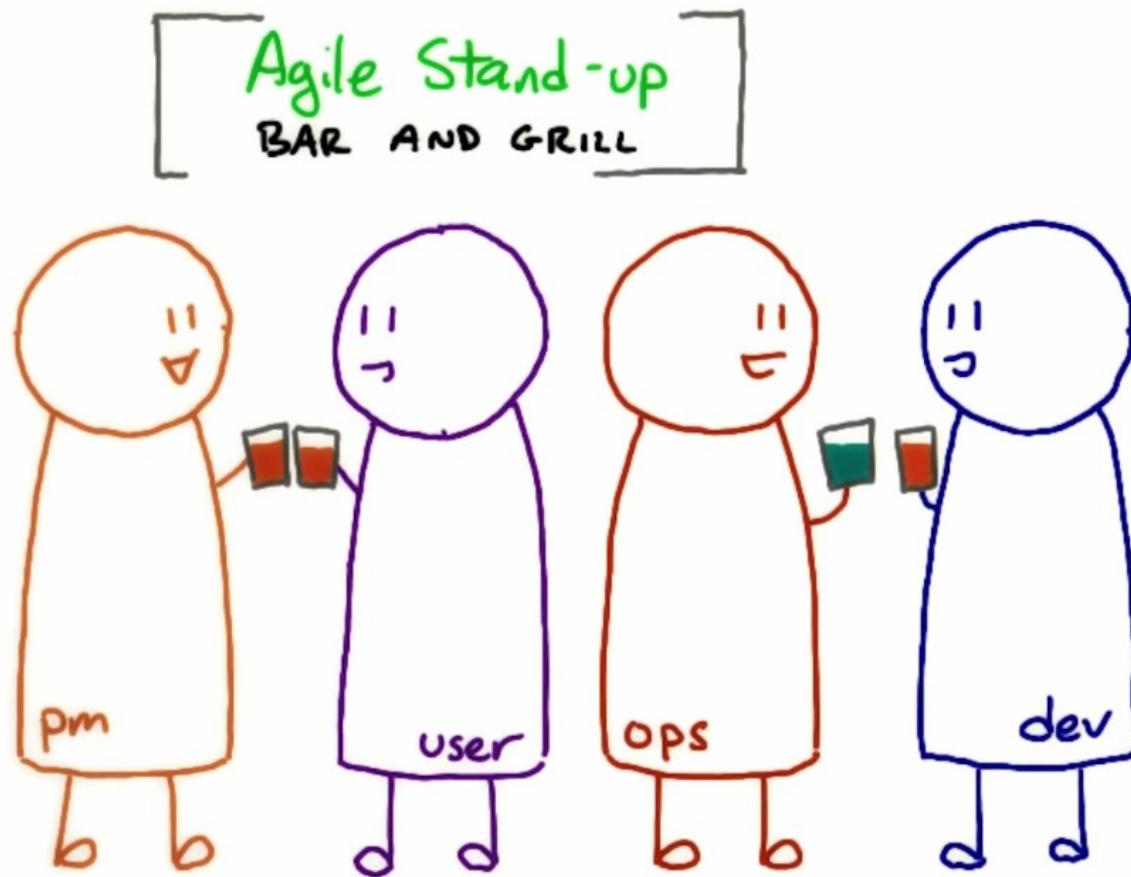
Agile Development Continuous Integration + Delivery

# Software Development Lifecycle

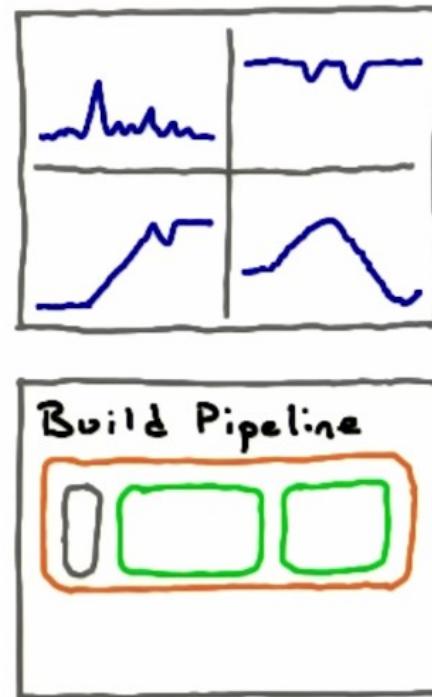
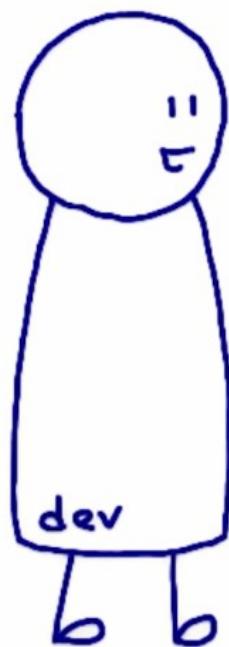


Agile Development Continuous Integration + Delivery DevOps?

# Software Development Lifecycle



# Course Project



## Organizational Problem: Noodle Search



#1 search site for pasta

"Noo" feature:

gluten free rice noodle search

# Organizational Problem: Noodle Search



#1 search site for pasta

"Noo" feature:  
gluten free rice noodle search



# Organizational Problem: Noodle Search

Let's launch more  
stuff and win  
more trophies!



## Organizational Problem: Noodle Search

Let's launch more  
stuff and win  
more trophies!



Some releases cause production problems.

Increased load leads to user-visible latency and timeouts. Users complain to support.

Ops responds by adding more server capacity.

## Organizational Problem: Noodle Search

What are the causes of Noodle Search's repeated production problems? Mark all that apply.

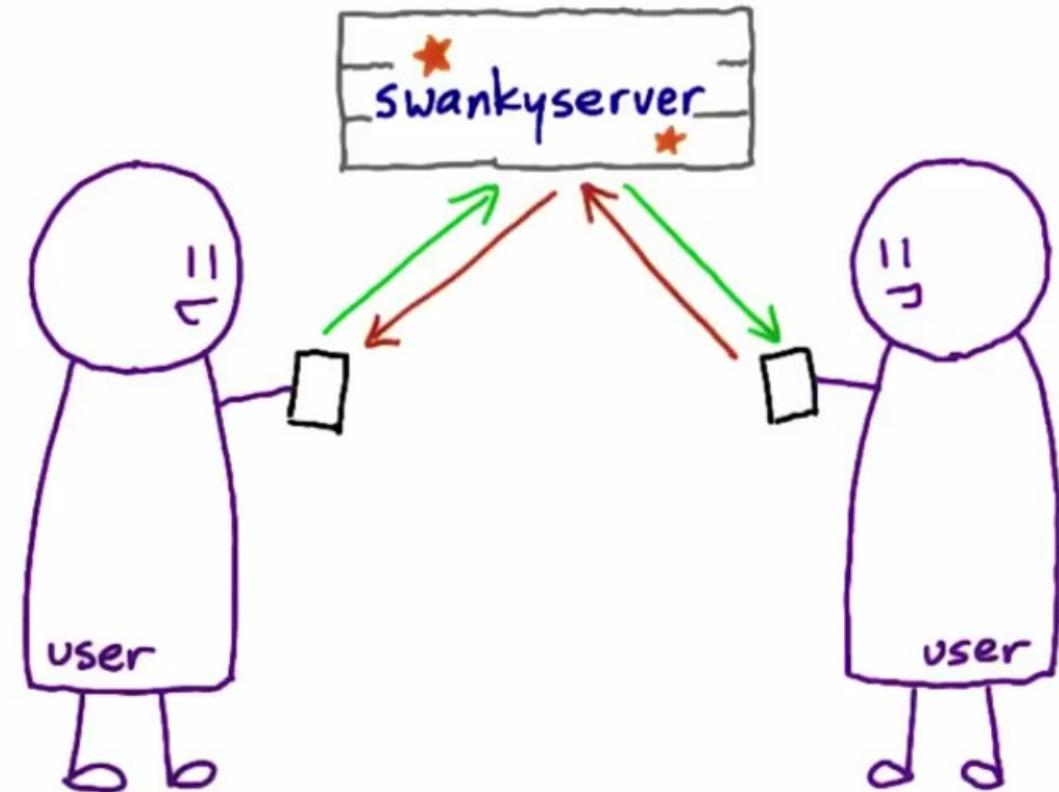
- The development environment doesn't match production.
- New (or "noo") releases aren't tested for performance.
- Ops should automate the release process.
- The business isn't anticipating its scaling needs.
- Developers are being too harsh to the operations staff.

## Organizational Problem: Noodle Search

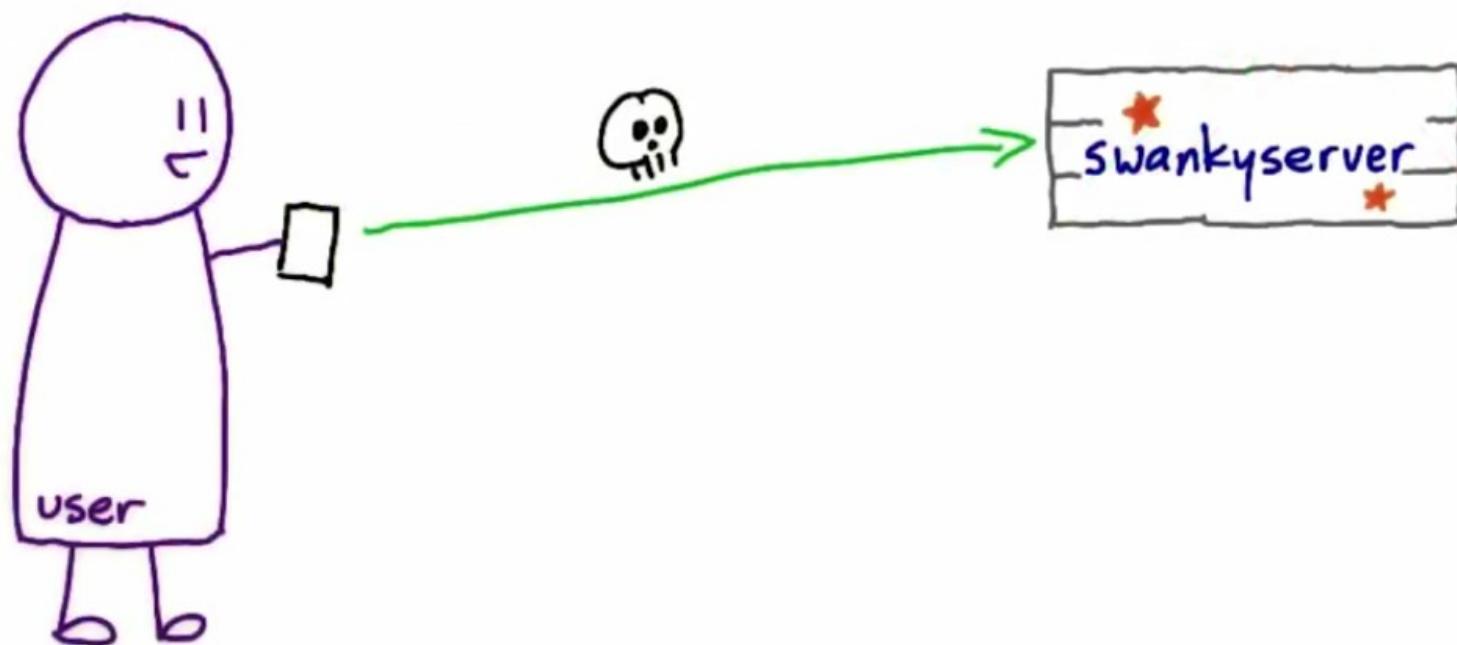
What are the causes of Noodle Search's repeated production problems? Mark all that apply.

- The development environment doesn't match production.
- New (or "noo") releases aren't tested for performance.
- Ops should automate the release process.
- The business isn't anticipating its scaling needs.
- Developers are being too harsh to the operations staff.

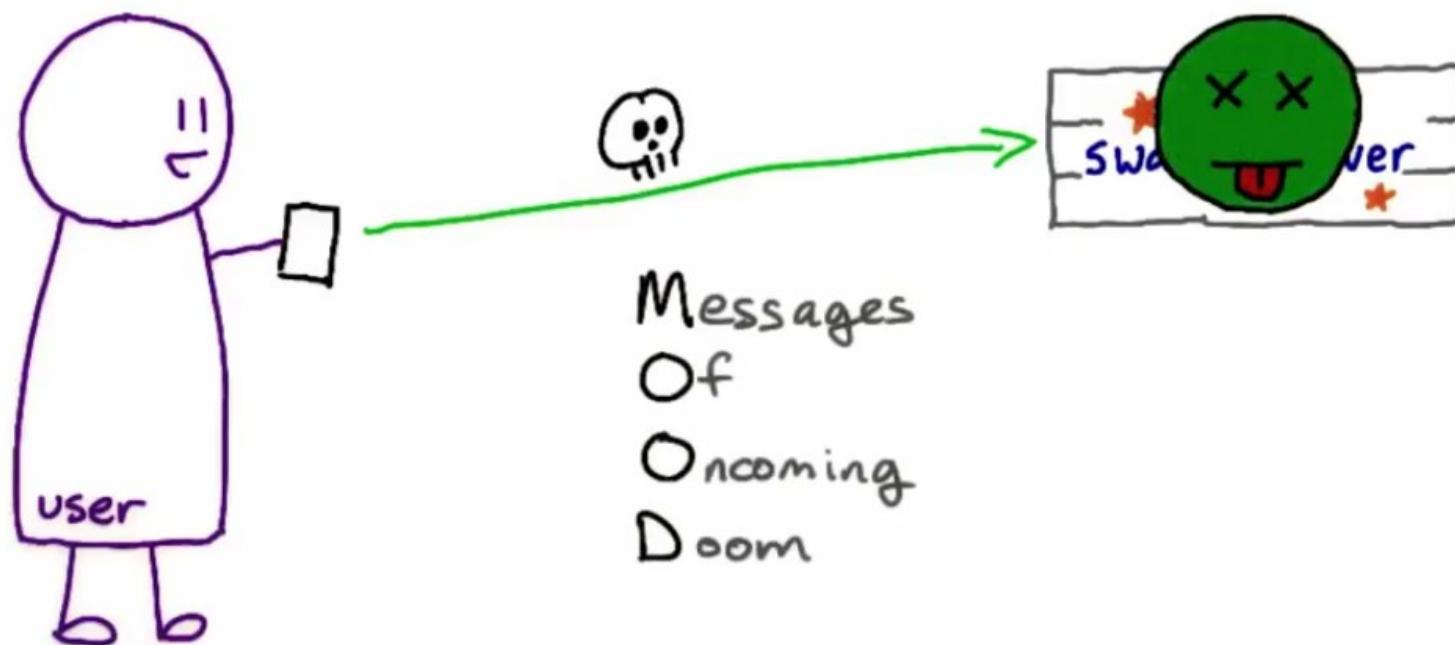
# Organizational Problem: Swanky Chat



# Organizational Problem: Swanky Chat

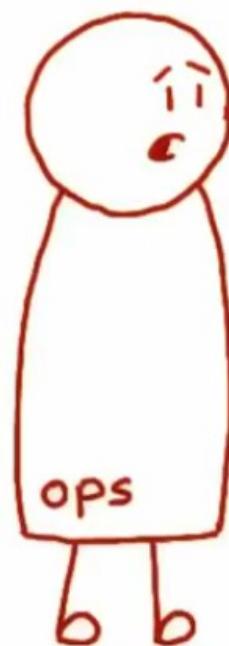


# Organizational Problem: Swanky Chat



# Organizational Problem: Swanky Chat

Hey, we could  
really use a  
fix for that  
crashing bug!

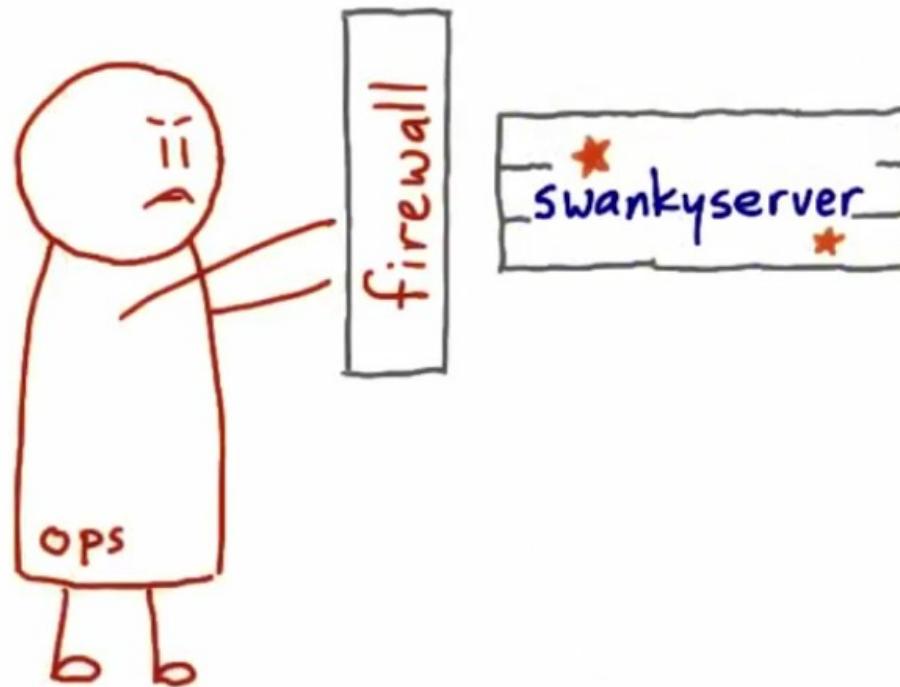


Sure thing...  
gotta finish  
writing this one  
new feature first.

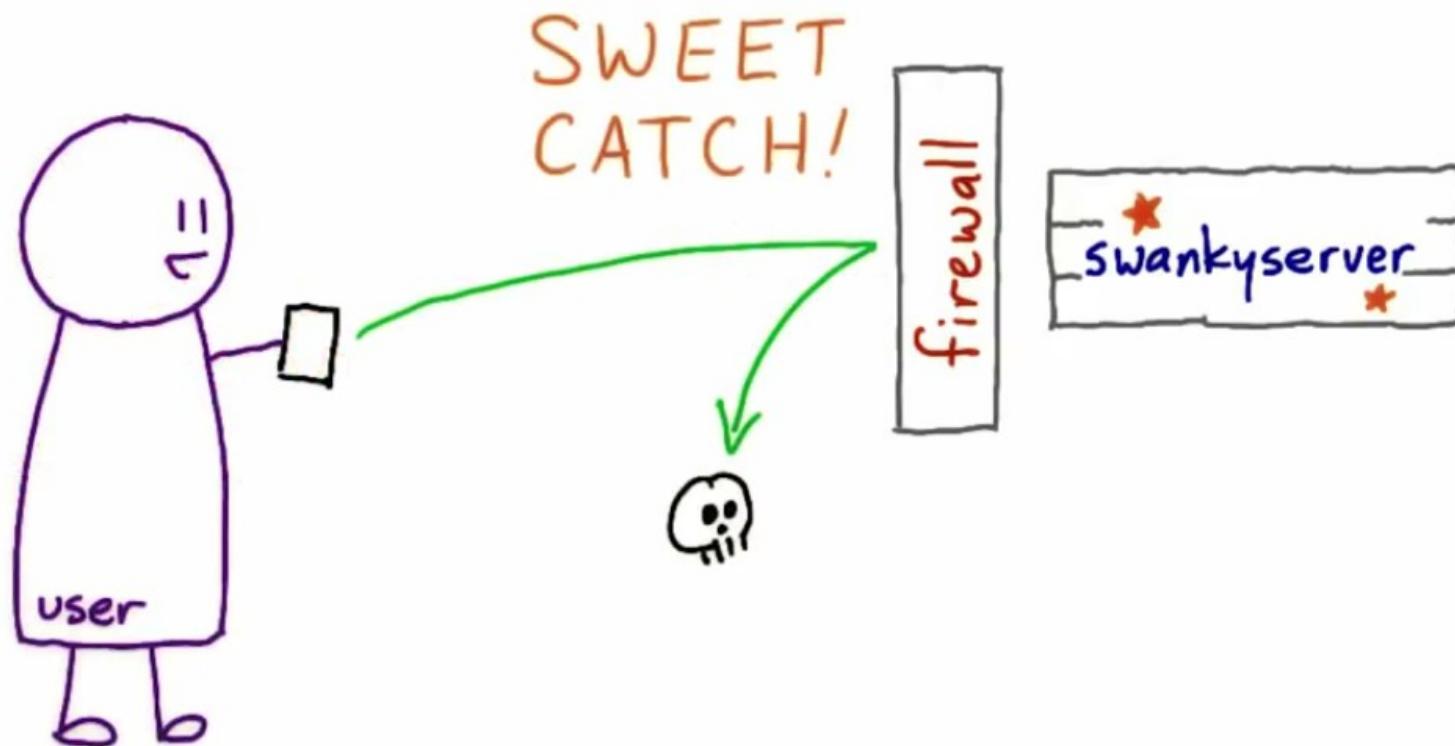


# Organizational Problem: Swanky Chat

freakin' devs...  
always leaving it  
up to ops to save  
the business from  
their bugs...



# Organizational Problem: Swanky Chat



# Organizational Problem: Swanky Chat

So can you  
fix it now?



ops



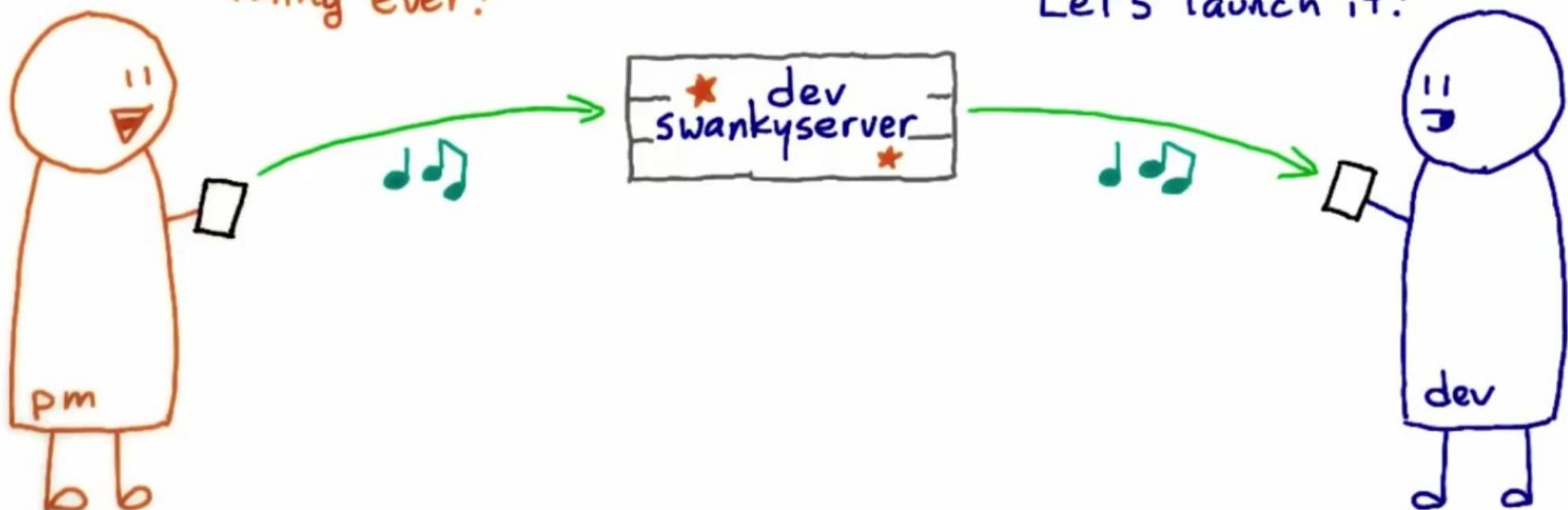
dev

Looks like you  
already did!

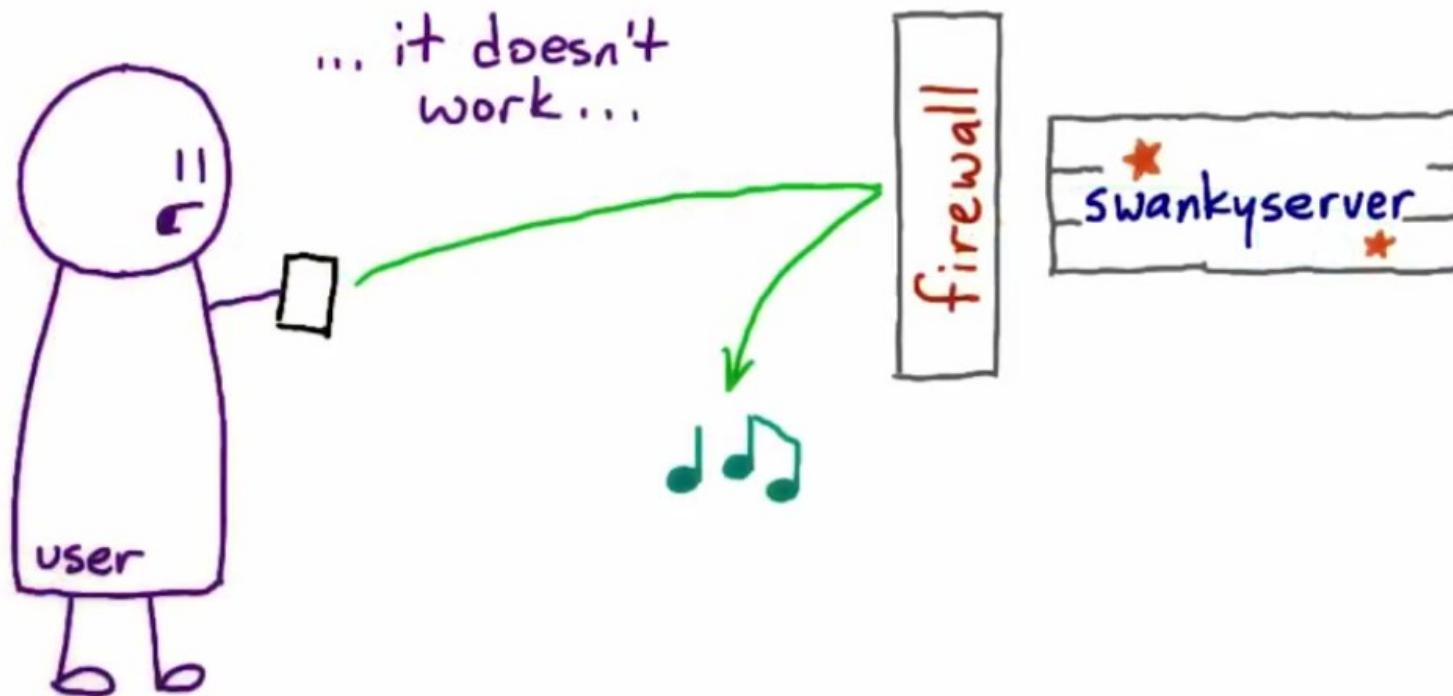
# Organizational Problem: Swanky Chat

This is the coolest  
thing ever!

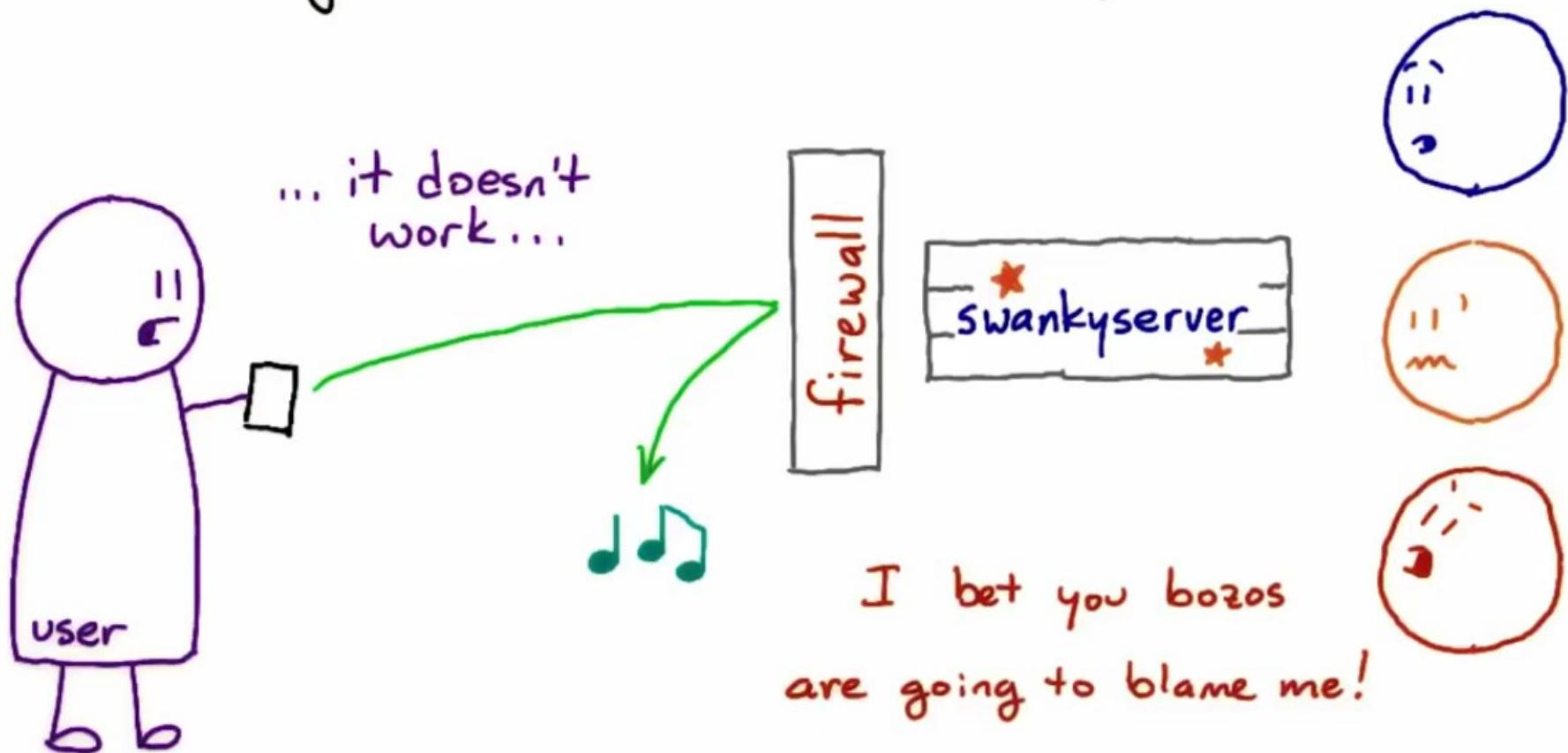
Thanks!  
Let's launch it!



# Organizational Problem: Swanky Chat



# Organizational Problem: Swanky Chat



## Organizational Problem: Swanky Chat

What are the causes of Swanky Chat's deployment problem?

Mark all that apply.

- The development and production environments don't match.
- New releases aren't tested for performance.
- A workaround relieved the pain and hid the severity of a bug.
- Firewalls are unreliable and should never be used.
- Operations should be fired and replaced with more developers.

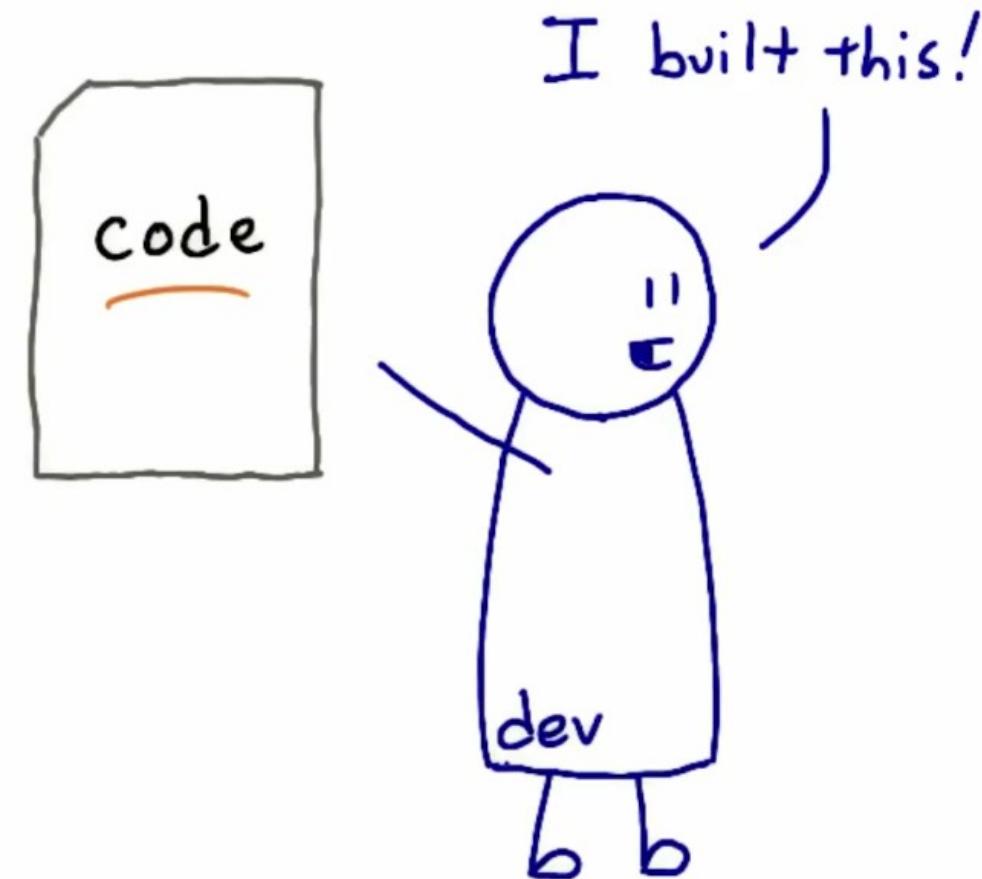
## Organizational Problem: Swanky Chat

What are the causes of Swanky Chat's deployment problem?

Mark all that apply.

- The development and production environments don't match.
- New releases aren't tested for performance.
- A workaround relieved the pain and hid the severity of a bug.
- Firewalls are unreliable and should never be used.
- Operations should be fired and replaced with more developers.

# Roles and Perspectives



Developer role:

- Write project code
- Implement feature requests
- Fix bugs
- Meet deadlines
- Launch stuff! Yay!

# Roles and Perspectives

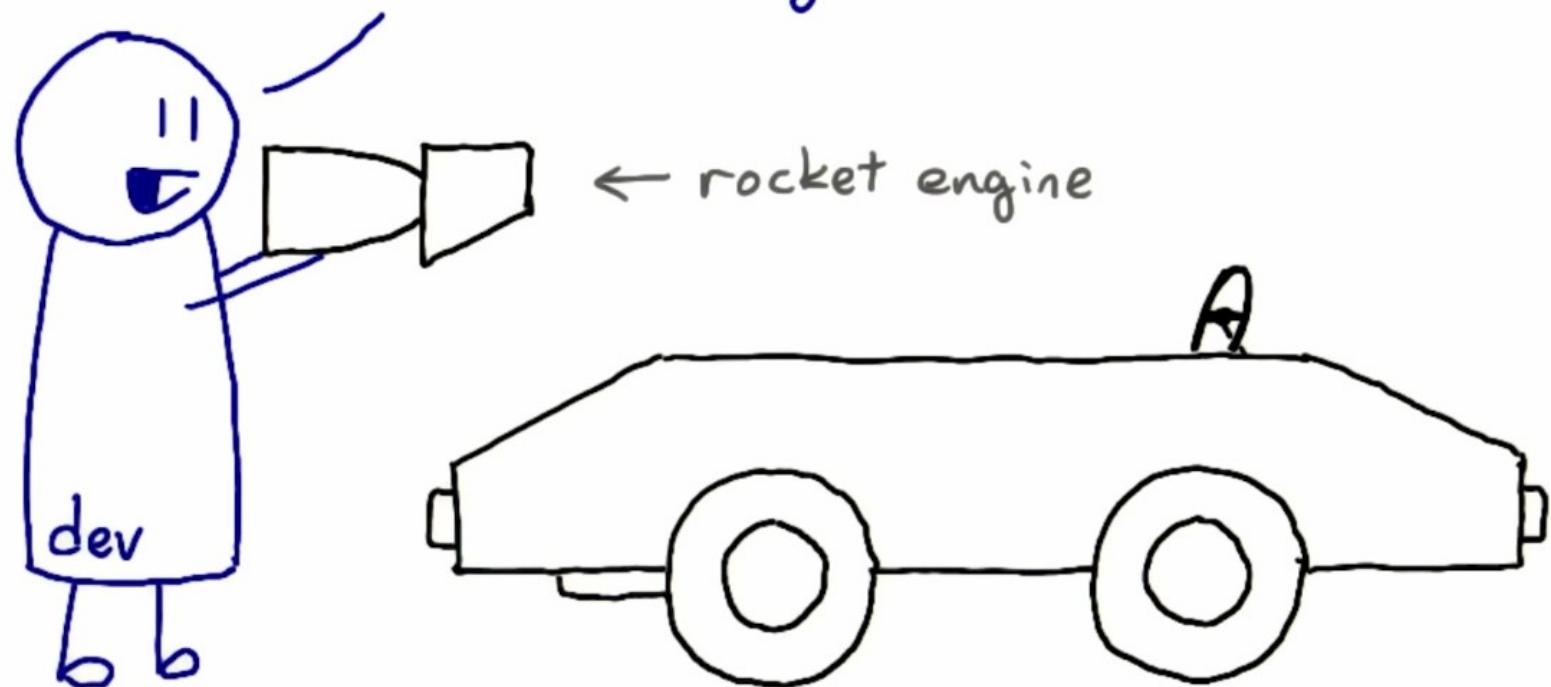


Operations role:

- Maintain services
- Deployment / scaling
- Monitoring
- Troubleshooting / diagnosis
- Incident management

# Roles and Perspectives

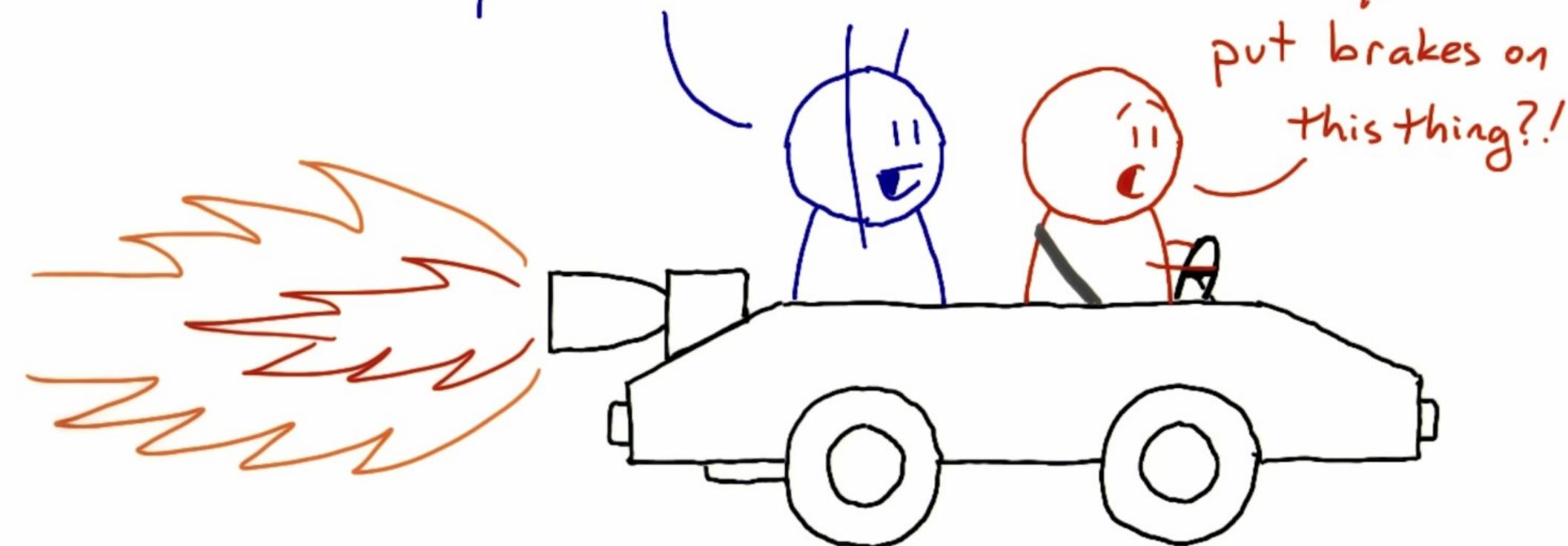
This is gonna be  
the coolest thing ever!



# Roles and Perspectives

Yay! Go faster!

Aaaah! Did you even  
put brakes on  
this thing?!

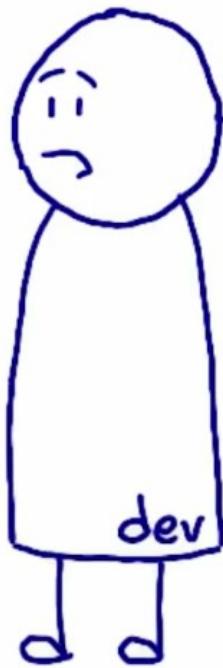


# Roles and Perspectives

I told the user  
we can solve the  
halting problem!

You know that's  
impossible, right?

So...  
next release?

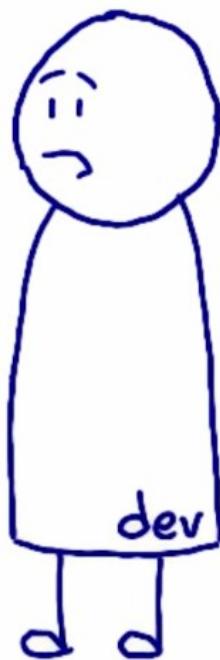


# Roles and Perspectives

I told the user  
we can solve the  
halting problem!

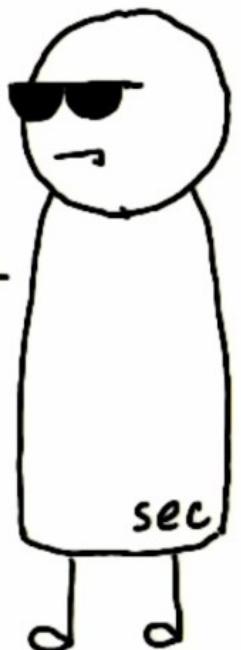
You know that's  
impossible, right?

So...  
next release?



We can't start a full release cycle today; we're already in the middle of one!

Well...  
I hope you don't mind being pwned by every passing script kiddie.

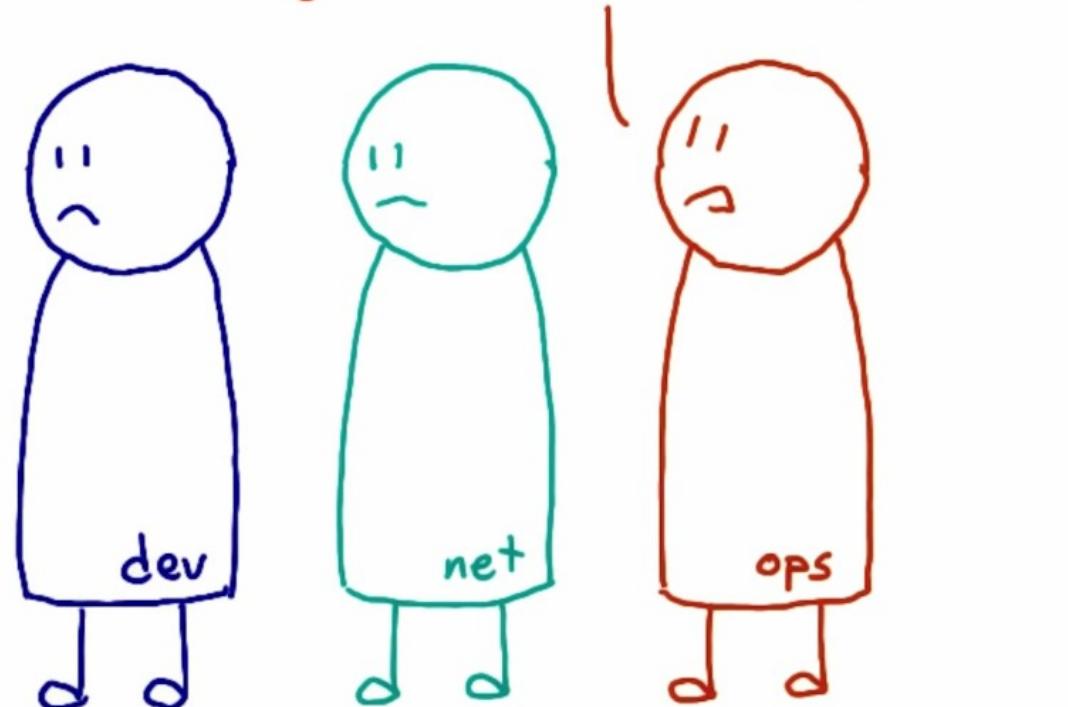


# Roles and Perspectives

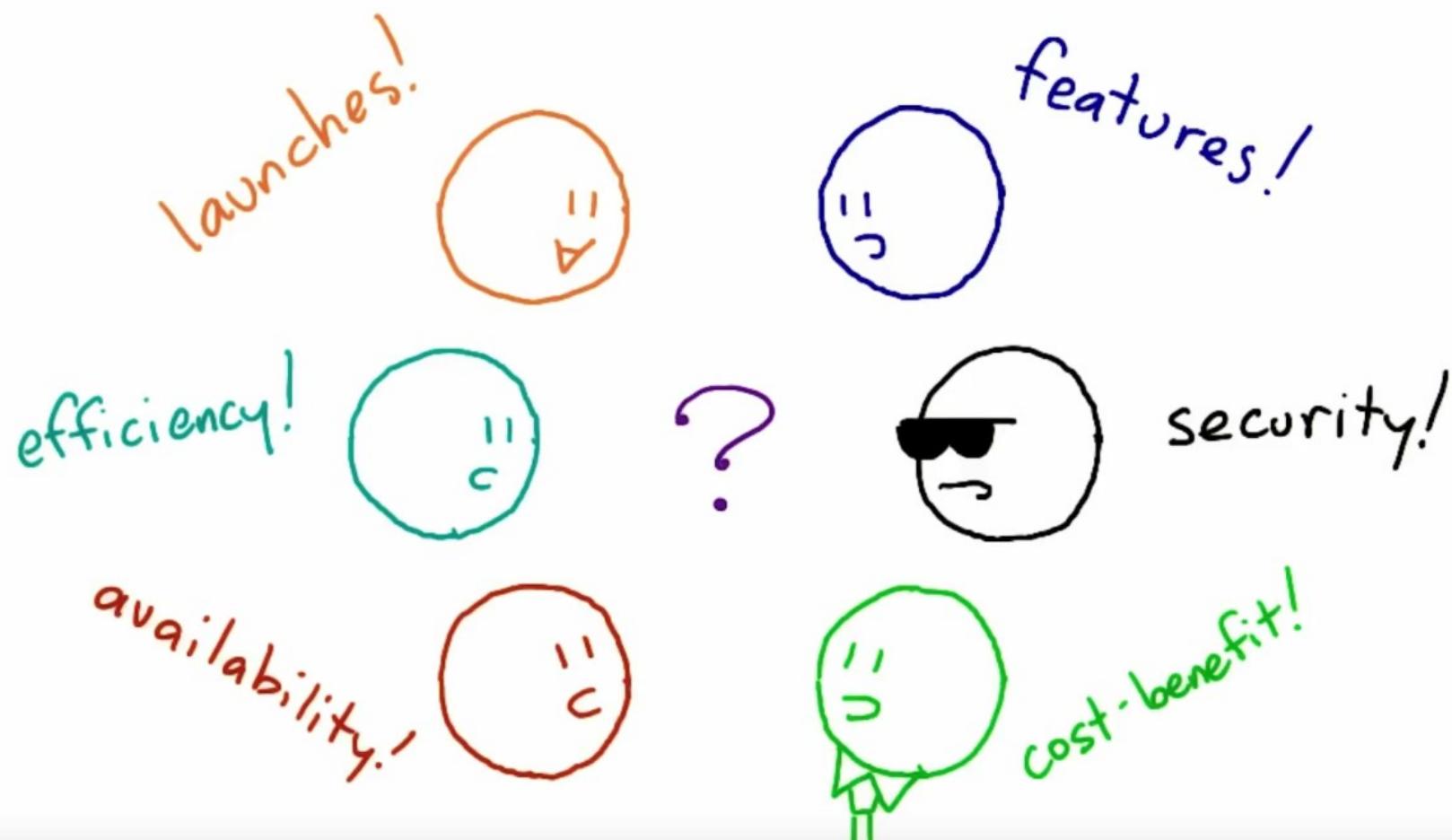
Revenue is down, so  
nobody can buy  
any new things.



... without some bigger servers,  
revenue's gonna be even more down.



# Roles and Perspectives



Organizational Problem: Wind and Shade

## WIND AND SHADE

Wildly popular distributed  
weather forecasting app

v1.0 (nameless)

v2.0 Serenity

v3.0 twinkly herbert

# Organizational Problem: Wind and Shade

## WIND AND SHADE

Wildly popular distributed  
weather forecasting app

v1.0 (nameless)

v2.0 Serenity

v3.0 twinkly herbert

Was there anything that went  
right with the 2.0 launch?



Well ...  
nobody died...



# Organizational Problem: Wind and Shade

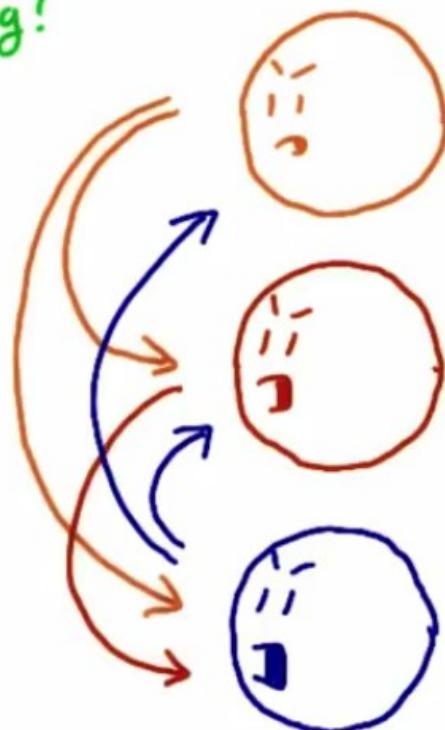


So what  
went wrong?

First Post-Launch Meeting:

We didn't get to test it!  
And the performance sucked!

BLAME



Those dev bozos built it on  
Windows when our production  
system is all Linux!

We had to build our own dev  
environment! We don't know Linux!  
Also, deadlines from hell!

## Organizational Problem: Wind and Shade

Hey, wait, we can't  
go on working like this.



Yeah, we should come  
up with an actual list  
of concerns for 3.0,  
based on the 2.0 launch.

Good idea. I'm sorry  
I called you a bozo.



Yay!  
I'll get the  
whiteboard!



## Organizational Problem: Wind and Shade

Which of these solutions would help solve Wind and Shade's launch trouble?

- Use agile development practices.
- Run production servers using devs' desktop OS.
- Involve Ops team earlier in development.
- Implement performance tests.
- Buy faster server hardware to make the app faster.
- Make smaller incremental launches instead of big ones.
- Require devs to only use technologies vetted at start of project.

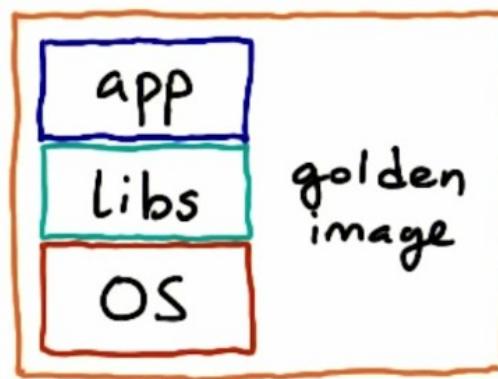
## Organizational Problem: Wind and Shade

Which of these solutions would help solve Wind and Shade's launch trouble?

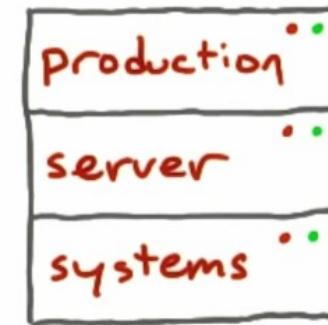
- Use agile development practices.
- Run production servers using devs' desktop OS.
- Involve Ops team earlier in development.
- Implement performance tests.
- Buy faster server hardware to make the app faster.
- Make smaller incremental launches instead of big ones.
- Require devs to only use technologies vetted at start of project.

# Solving the Environment Problem

Solution 1:



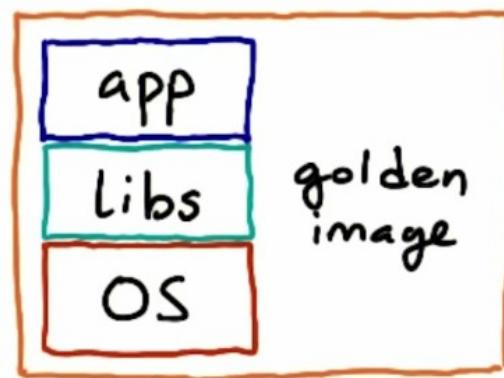
full system image  
containing all dependencies  
produced by build automation



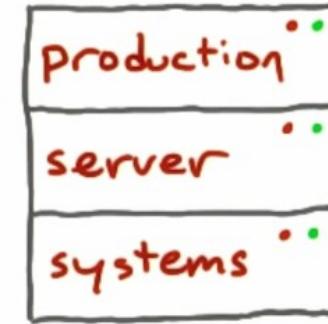
# Solving the Environment Problem

Solution 1:

testing  
(in VM)

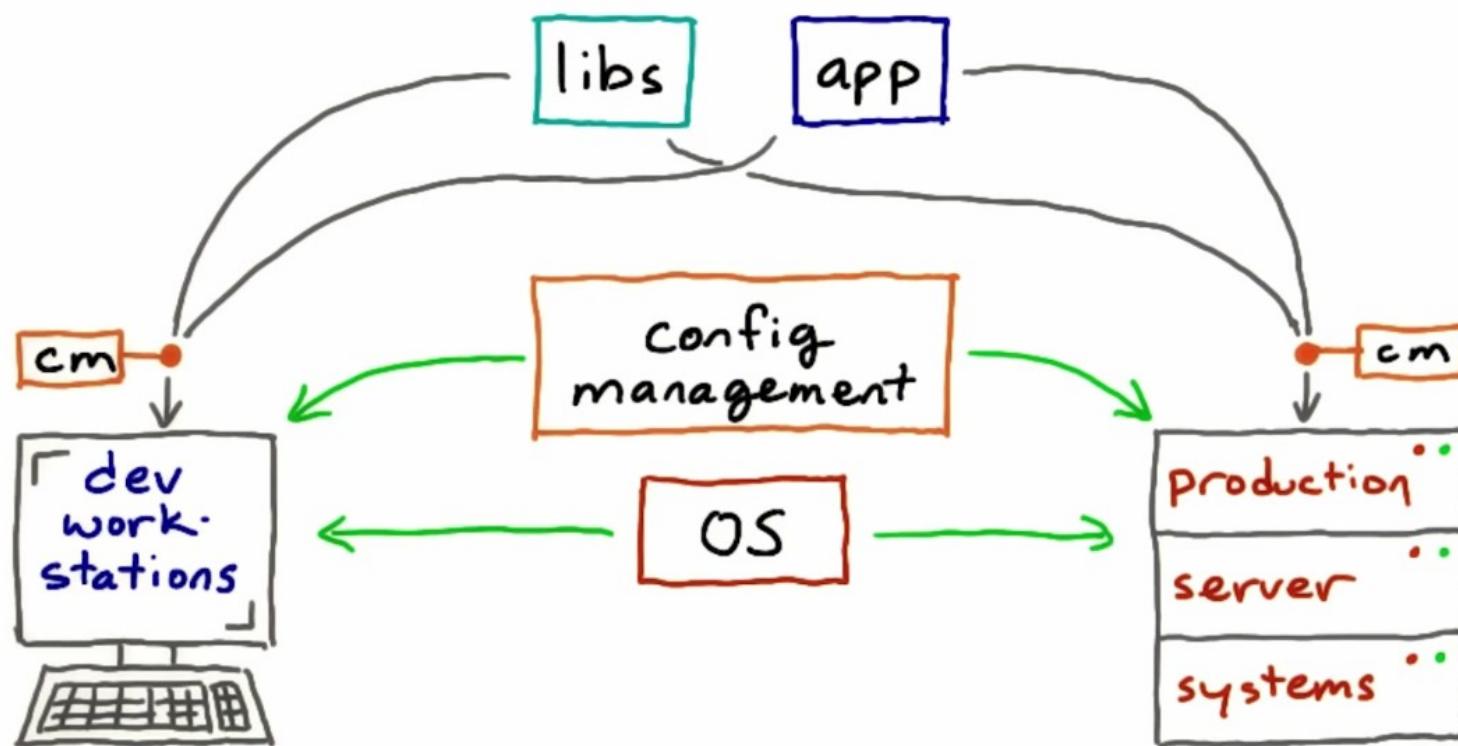


deployment



# Solving the Environment Problem

Solution 2:



# Solving the Environment Problem

golden image:

more work up front -

large install image must be  
regenerated for any change

much faster installation/boot

immutable image

configuration management:

lighter build process -

integration is done at  
install/initial boot time

slower startup process

easy to update

These approaches can be combined!

# Solving the Environment Problem

Either way, it's all checked in  
to our repo, right?

Yes! So we can easily  
regenerate the system.

Let's do a golden image, so  
my test VM will be just  
like production.

OK!

