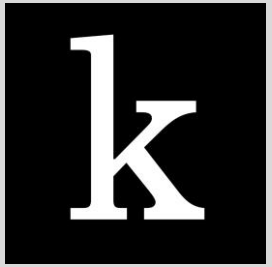
Is Your Software Going to Fail?

Obligatory "Who is this speaker?" slide

- My name is Michael Gregory
- Site Reliability Engineer at OverDrive
- But started my career as a developer and this presentation is intended for developers
- Part-time graduate student getting a theology degree
- Enjoy running and biking. Trying to figure out the swimming part for a triathlon this summer.



Is Your Software Going to Fail?

YES!

How will your software fail?

- Code
- Platform
- Dependencies
- The Network
- Human Error

What do I mean by fail?

- Failure is any way in which your application does not behave as expected.

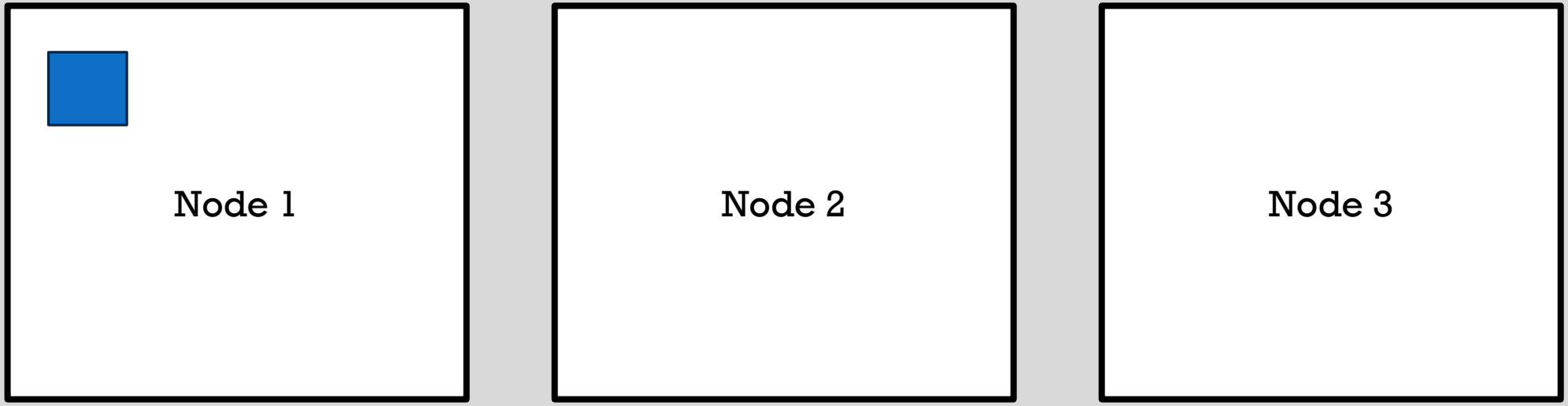
Code

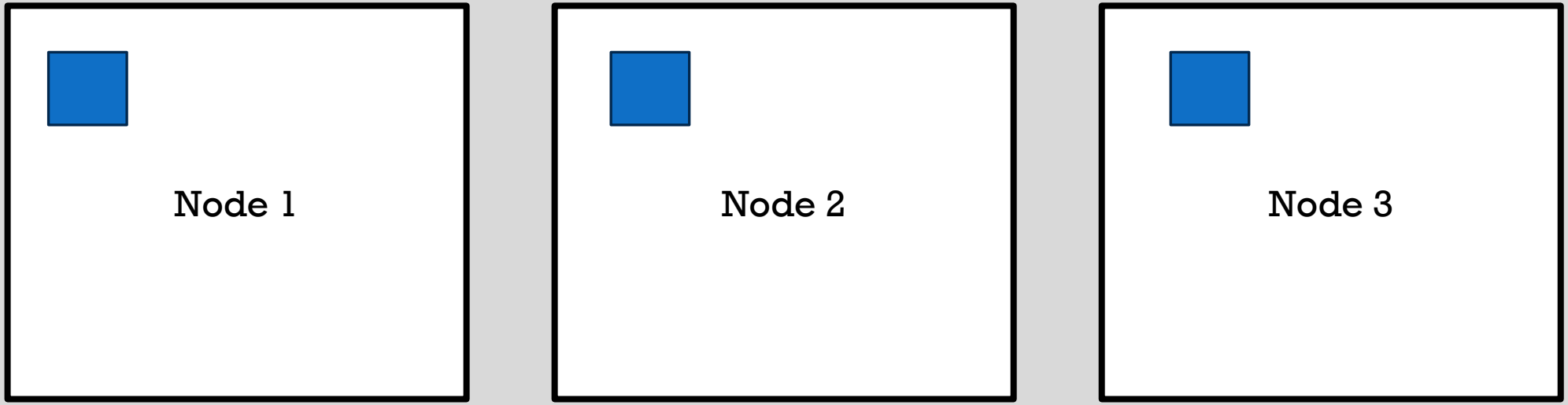
- How does it fail?
  - Bugs
  - Bad specifications

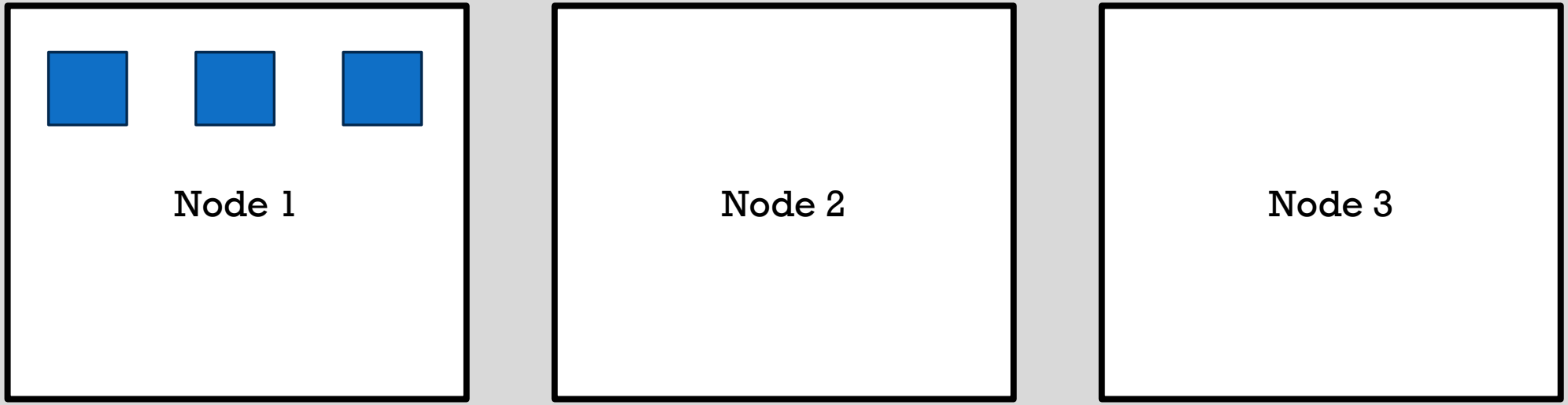
- How can we prevent these types of failures?
  - Pull Requests and code reviews
  - Automated Testing
  - Manual Testing
  - Ask questions
  - Short cycles

Platform

- How does it fail?
  - Depends on the platform
    - Are you running on a real server?
    - A virtual machine?
    - In the cloud? Which cloud?
    - Do you use containers? How are they deployed?
- How can we prevent these types of failures?
  - Know your platform (or at least work **closely** with someone who does)
  - What are its advantages?
  - What are its disadvantages?

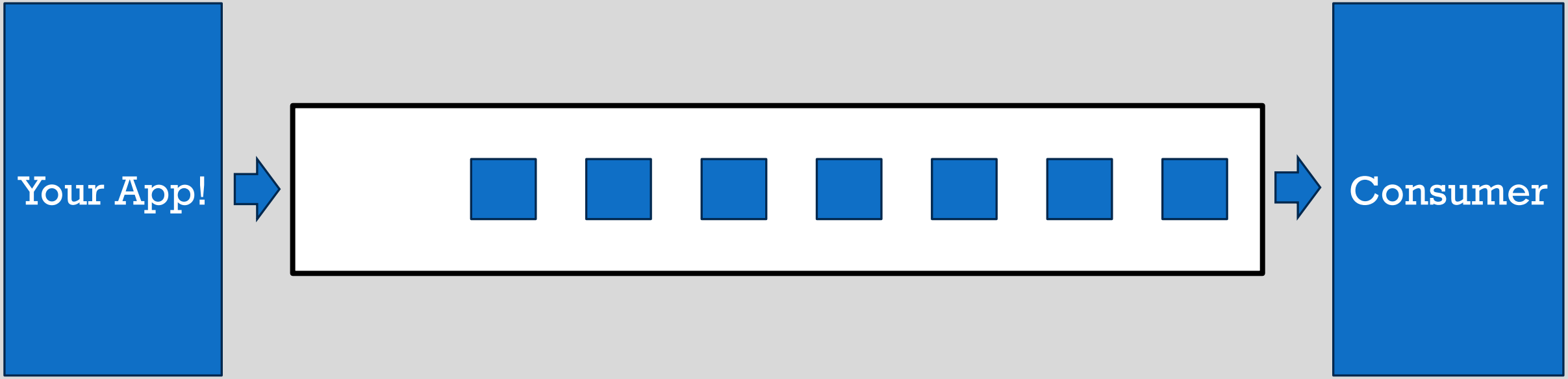

Node 1
Node 2
Node 3


Node 1
Node 2
Node 3


Node 1
Node 2
Node 3

Dependencies

- How does it fail?
  - It depends! (the favorite words of all developers)
  - Bugs in dependent code libraries
  - External dependencies could have outages
  - Dependencies have their own failure conditions
- How can we prevent these types of failures?
  - Choose your libraries carefully
  - Keep libraries up to date
  - Know your dependencies, especially how they fail
  - Can your application degrade gracefully?
  - Is anything else dependent on your application?


Your App!
Consumer

The Network

~~The Network~~ The Outside

- How does it fail?
  - ???
- How can we prevent these types of failures?
  - We can't
  - Have a plan for detecting and handling incidents
  - Be careful about assigning blame outside too quickly

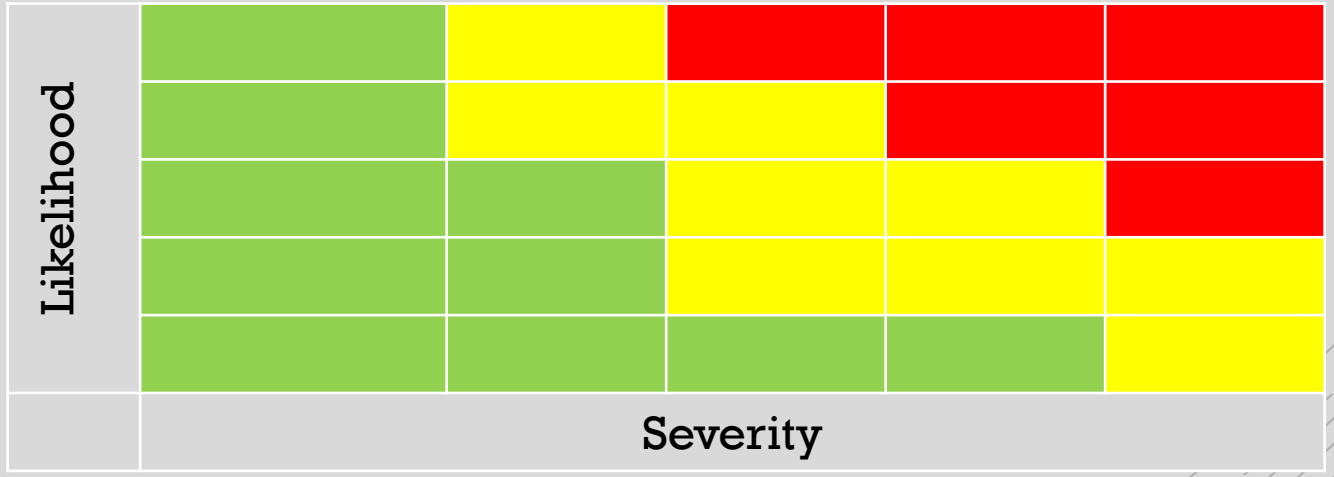
Human Error

- How does it fail?
  - Skipping steps
  - Undocumented config change
  - Deploying manual builds
- How can we prevent these types of failures?
  - Automate everything you can
  - Build/deploy pipelines you like
  - Store configuration in source control
  - Have an escape hatch for emergencies


uh... everything's perfectly all right now
We're fine. We're all fine here now, thank you.
quickmeme.com

Final Thoughts

- Stop making failures personal
- Accept that failures happen
- Prepare for failures when you can
- Break all the things in dev!
- Start small
- The answer for a particular failure case may be to do nothing or just to set up an alert



Thank You

Ask questions in the Discord channel today or after