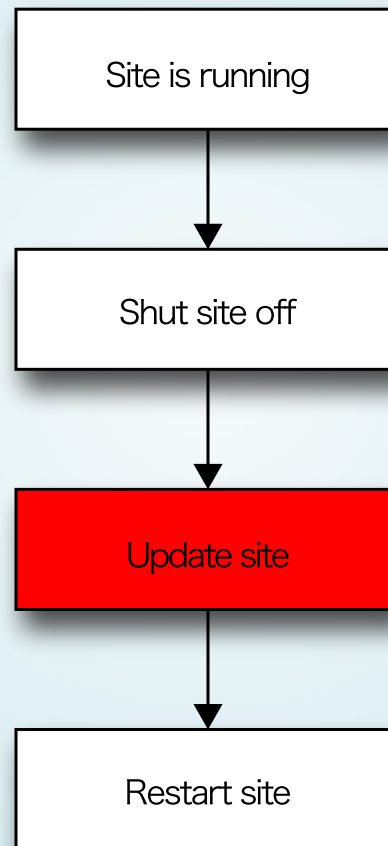


STEALTHY DJANGO DEPLOYMENTS

HOW TO UPDATE A DJANGO SITE WITHOUT ANYBODY
NOTICING YOU DID IT

Jeremy Bowman
jbowman@edx.org

TRADITIONAL DEPLOYMENT



WHY THE NEED FOR STEALTH?

- Minimize inconvenience to users
- Reduce site errors during deployment
- Deploy when other people are around to help
- Enable frequent deployments

MAIN POINTS TO CONSIDER

- In-progress requests
- Asynchronous tasks
- Package installation, upgrades, and removal
- Database migrations
- Static assets
- Configuration changes

ADD OR RELOAD WSGI WORKERS

- Wait until all the new code is in place
- Use WSGI reload instead of restart
- Old and new code must be able to run concurrently

RESTART ASYNCHRONOUS WORKERS

- Stop Celery, etc. before making any real changes
- Limit the duration of asynchronous tasks
- Allow some time for tasks to wrap up
- Terminate workers if shutdown takes too long
- Restart at end of deployment

INSTALL OR UPGRADE DEPENDENCIES

- Use wheels if at all possible
- Keep an “uninstall.txt” file
- Upload package files before installation

DATABASE MIGRATIONS

- Always backwards-compatible
- Sometimes requires multiple deployments
- Test on a production-scale database when possible
- Watch out for deadlock risks

WHAT IS A BACKWARDS-COMPATIBLE MIGRATION?

- Don't remove or dramatically change an in-use field
- Don't remove an in-use table
- Don't directly rename an in-use field
- New fields must initially be nullable

Automate these checks whenever possible!

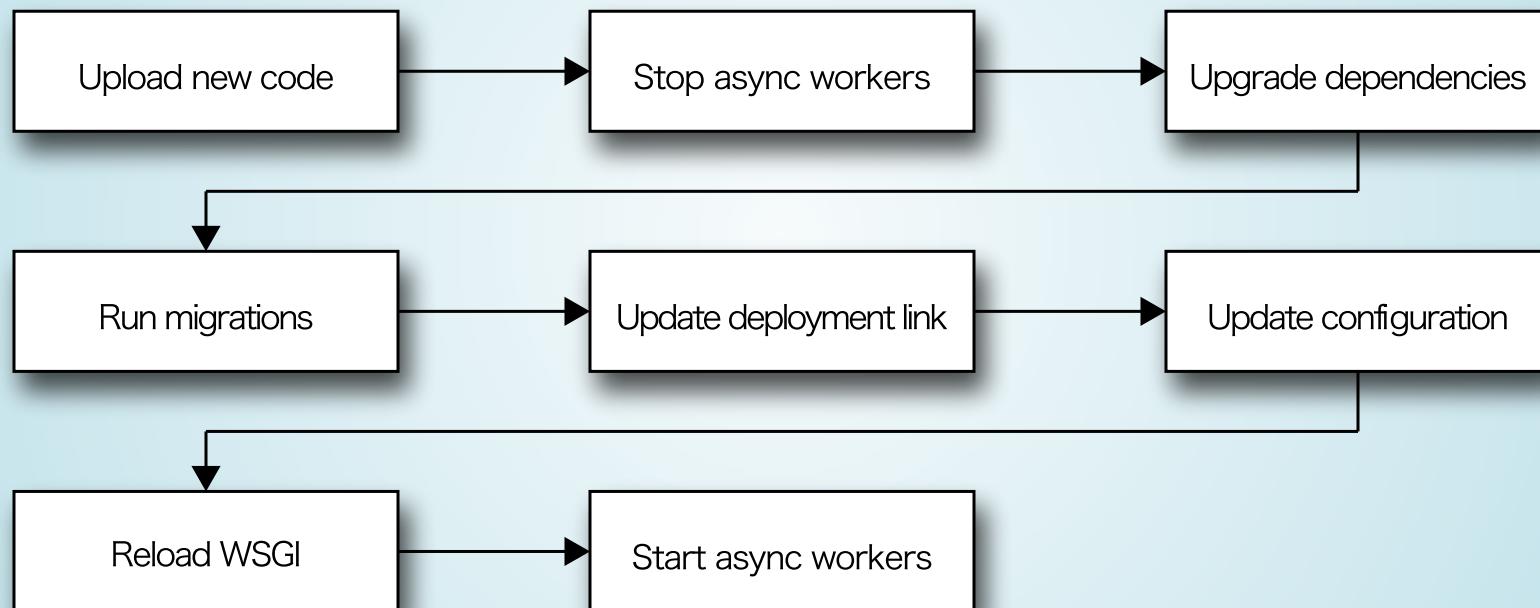
UNDERSTAND YOUR DATABASE

- PostgreSQL: Avoid new fields with default values
- MySQL: Schema changes are slow on any large table
- MySQL: No transactional schema changes

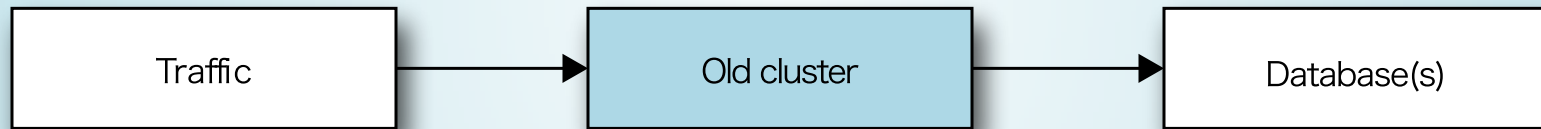
STATIC ASSETS

- Append hashes to filenames
- New files must be available before WSGI reloads
- Don't stop serving the old files

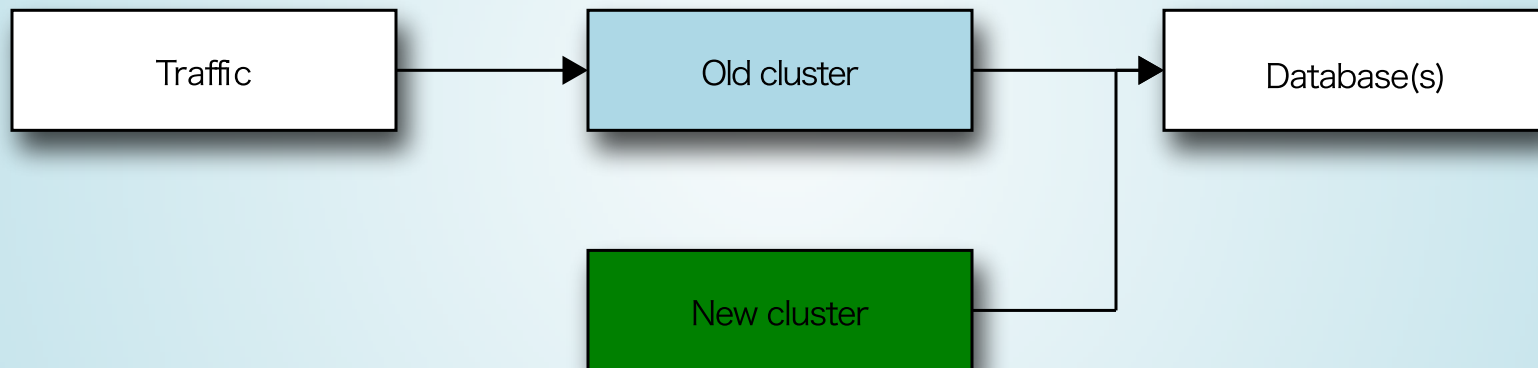
IN-PLACE STEALTHY DEPLOYMENT



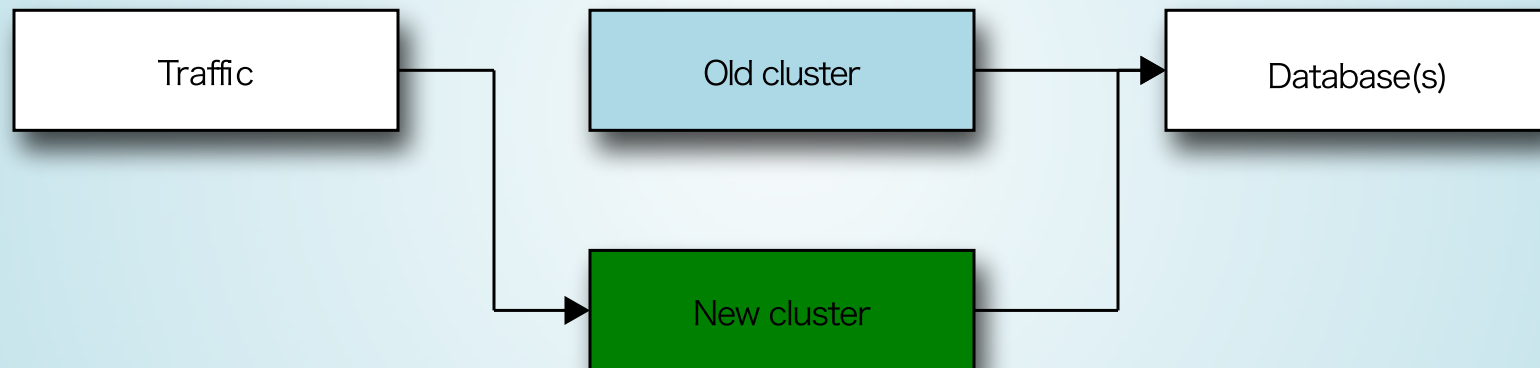
BLUE-GREEN DEPLOYMENT



BLUE-GREEN DEPLOYMENT



BLUE-GREEN DEPLOYMENT

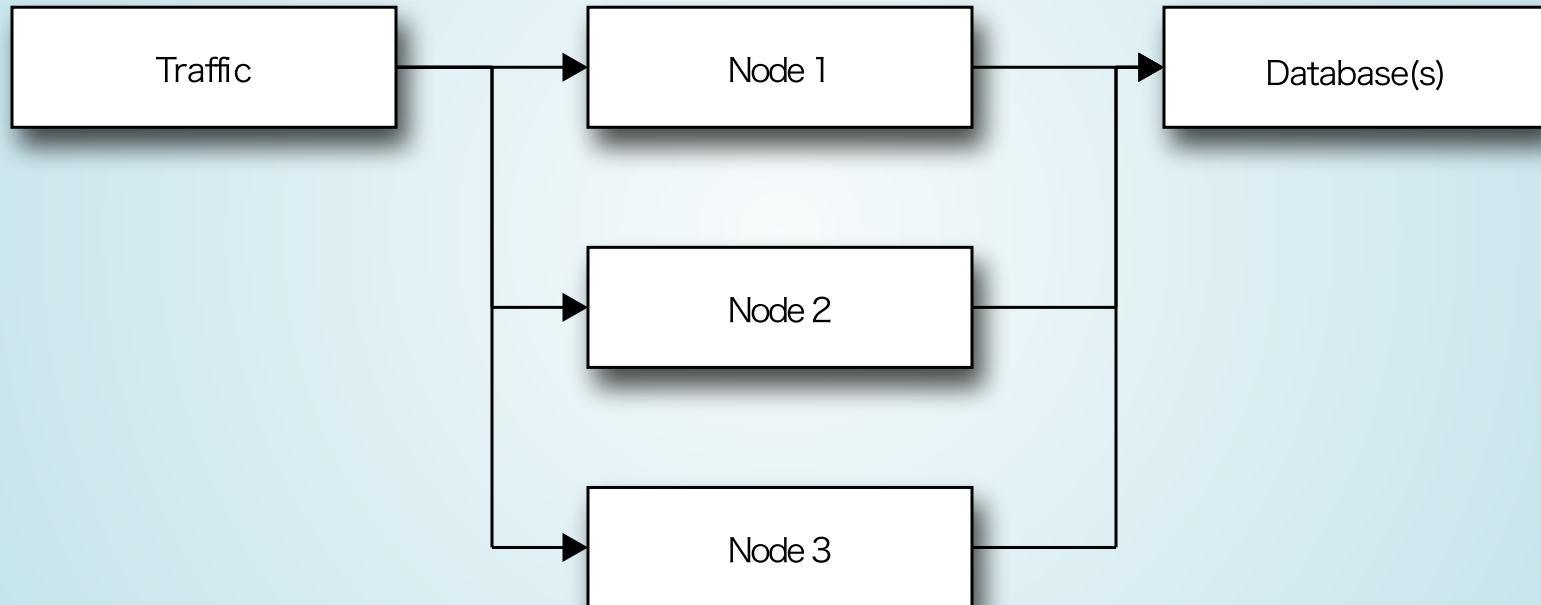


BLUE-GREEN DEPLOYMENT

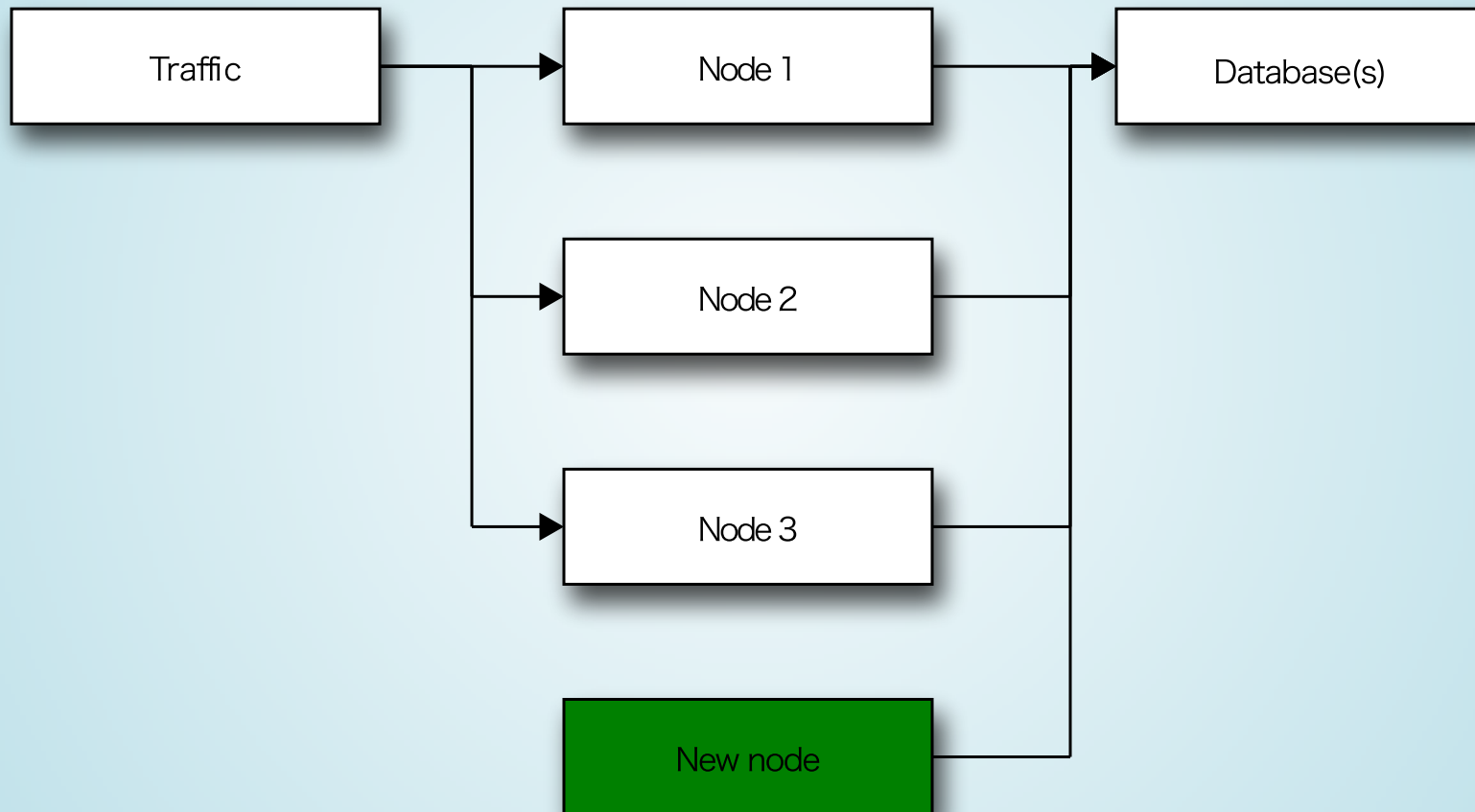


- Database migrations **must** be backwards compatible

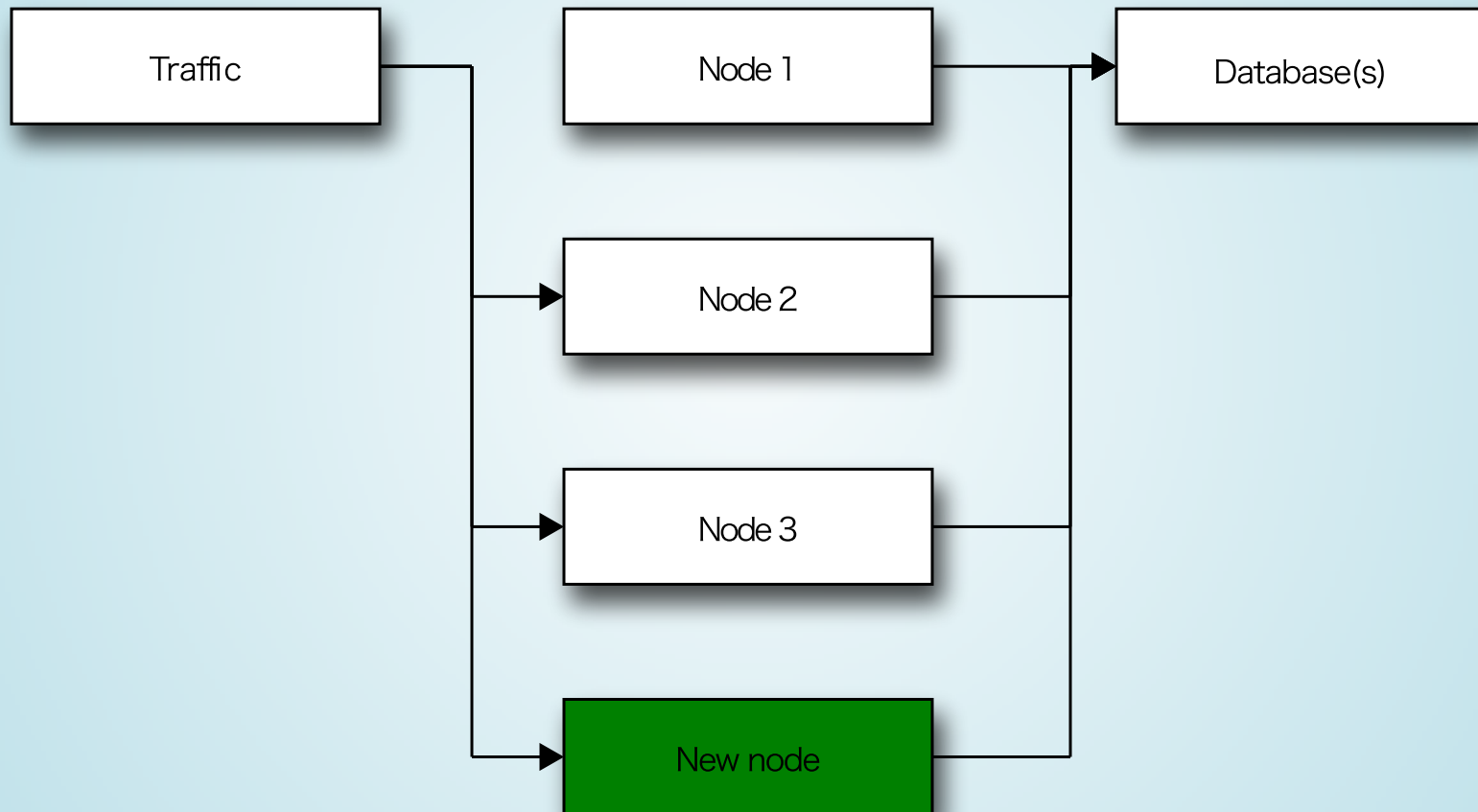
ROLLING DEPLOYMENT



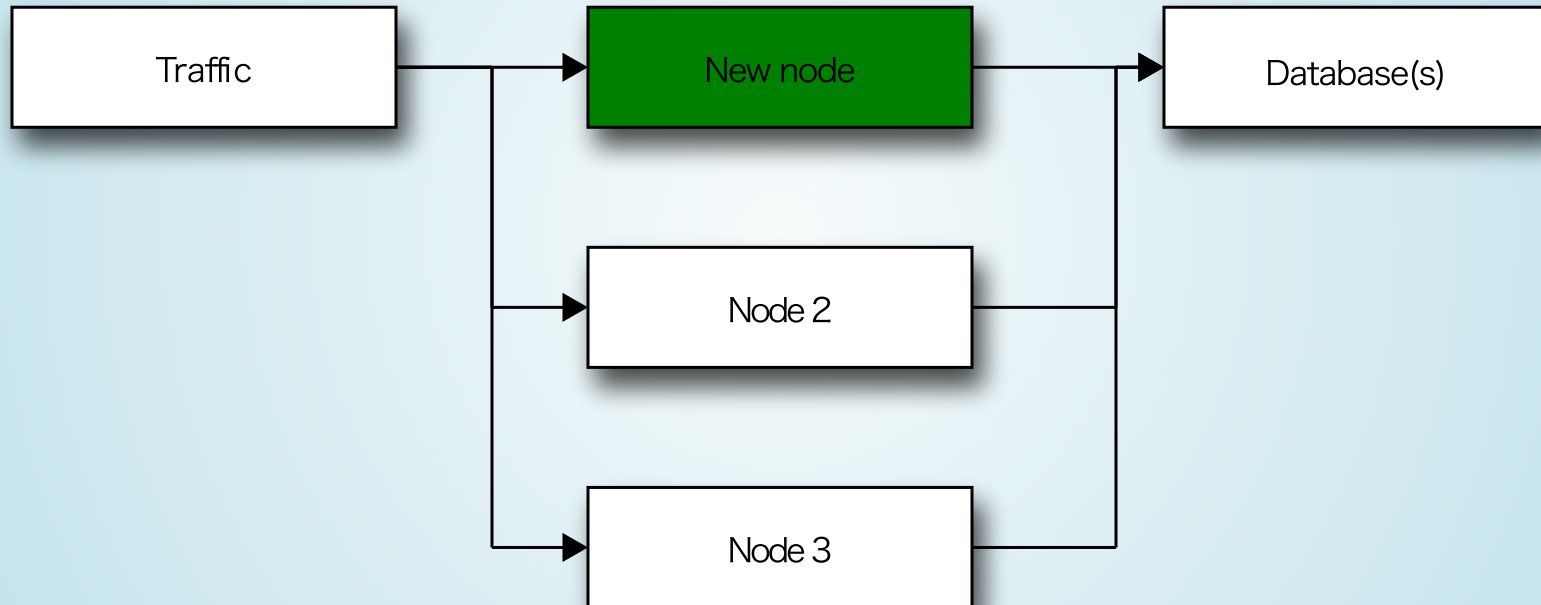
ROLLING DEPLOYMENT



ROLLING DEPLOYMENT



ROLLING DEPLOYMENT



OTHER TIPS

- Automate, automate, automate
- Test the deployment process in staging
- Maintain backwards compatibility with other services

THANK YOU!

Questions?

Jeremy Bowman
jbowman@edx.org