# Challenges with managing Infra

- Problems should be quickly detected and all systems should be consistently configured and up to date & IT staff should spend less time on routine drudgery and focus more on improving systems to meet changing needs of customers
- Cloud and automation have made things worse - ease of provisioning new infra leads to an ever growing portfolio of systems and it takes an ever-increasing amount of time to manage and keep things from failing
- Adopting cloud and automation tools immediately lowers barriers for making changes to infrastructure
- Managing changes in a way that improves consistency and reliability doesn't come out of the box with the software.

# What is Infra as Code

- Approach to infra automation based on practices from software development
- Emphasizes on consistent and repeatable routines for provisioning and changing systems and their config
- Changes are made to definitions and rolled out to systems through unattended processes that include thorough testing and validation
- This includes applying VCS tools, automated testing libraries
- This also opens door to incorporate dev practices such as TDD, CI and CD

## Goals

1. Anyone can propose a change and it is upto reviewers to review and apply it to prod
2. Make infa changes fairly quickly without waiting on other teams to do it for us
3. Scaling up and scaling down is made as easy as raising a PR
4. Immutable server deployments

## Challenges

1. Tracking 100s and 100s of servers and managing them applying the latest security patches to all . Pushing the latest changes in code to all servers (server sprawl)
2. Configuration drift - when something is fixed on a live server, but is not set in the code base - it is better not to manage these locally
3. We don't want servers to unique, we want it to be just like any other server ( snowflake servers )

## Principles

1. Systems => easily reproduced - they are disposable :Software should continue to run
2. Handling changes gracefully which makes it easier to make improvements and fixes to running infra
3. Cattle not pets - treat server like cattle and not pets. they should be identical, replaceable. Stateful (DB) servers are pets, stateless servers are cattle
4. Systems consistent everytime bring it up
5. Processes are Repeatable
6. Design is always changing