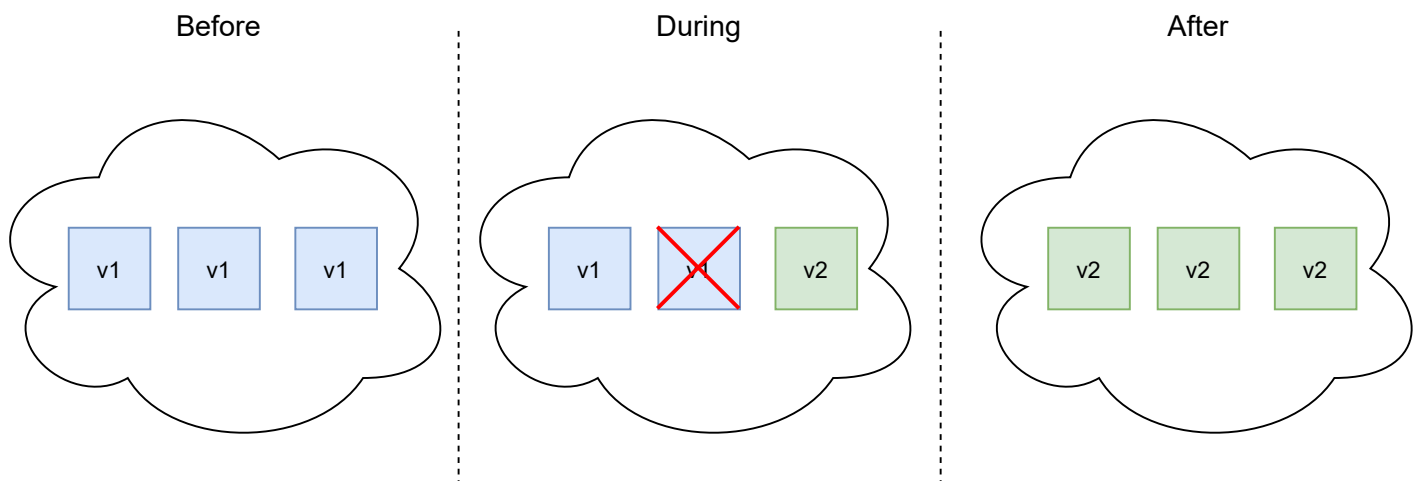
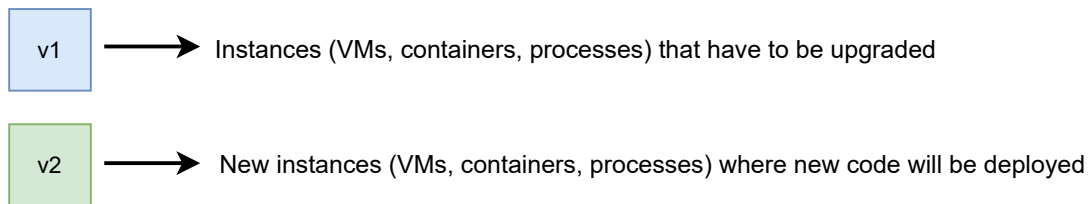


Solution for deploying the application

Rolling upgrade

Strategy which consist on adding new code into an existing deployment

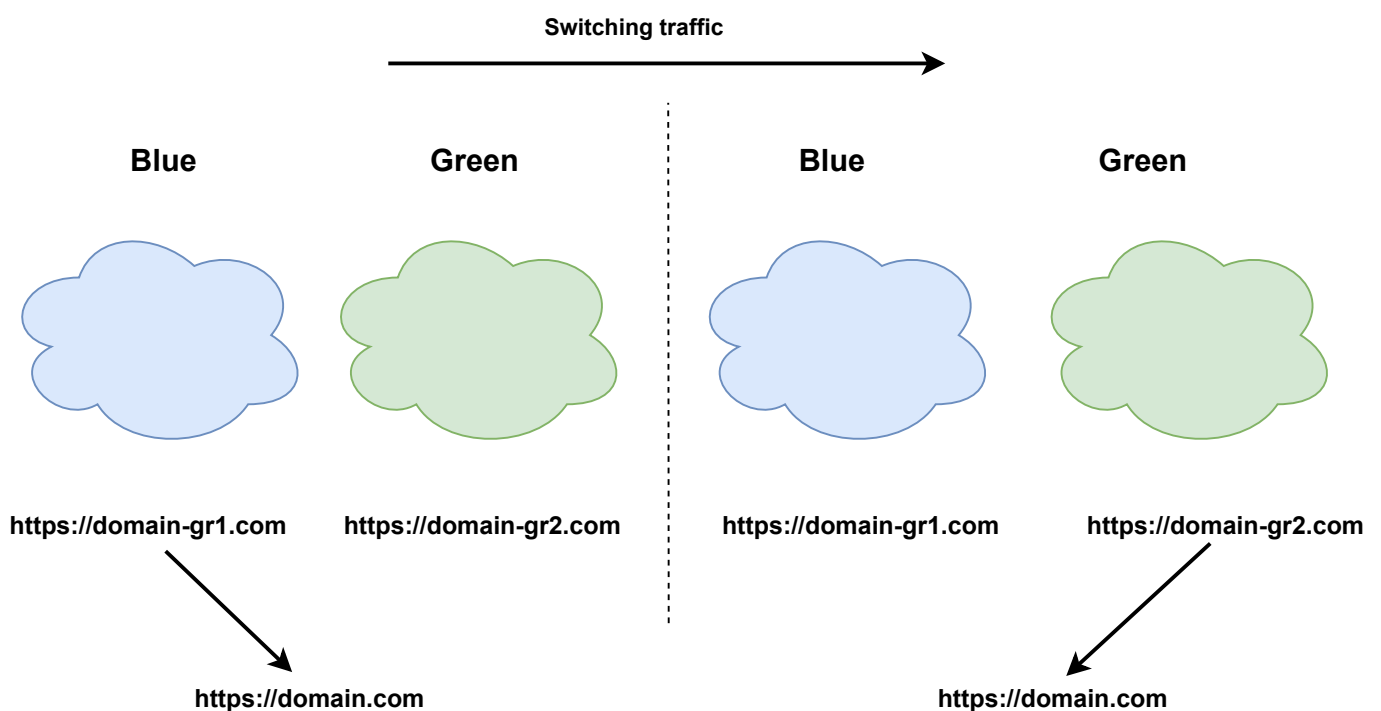


Blue-Green upgrade

Strategy that consists on deploying a new cluster (green), which is a copy of the production running cluster (blue), with the new code/version release. Once the green cluster is ready for production, the traffic is switched.

This is the safest strategy with minimal downtime, as the green cluster is fully isolated from the current production environment (blue).

Blue-Green upgrade strategy provides the ability of rolling back to blue deployment in case of any issue after the upgrade.



Canary upgrade

Canary deployment combines the two above methods. It involves first deploying a new code application in part of the production infrastructure for testing, as in blue-green deployment. But rather than having a fully cloned environment, this testing environment is just a small portion of the total infrastructure.

Once the update receives approval for release, canary deployment makes use of the phased approach of rolling deployment to expose a subset of users to the new software. The number of users routed to the new application then increases over time as the software receives confirmation of functionality. In this manner, the new and old versions exist simultaneously for a little while, but the new version progressively replaces the older one.

My recommendation is the Blue-Green upgrade strategy:

This technique can eliminate downtime due to app deployment. In addition, blue-green deployment reduces risk: if something unexpected happens with the new version on Green, it can be immediately rolled back to the last version by switching back to Blue