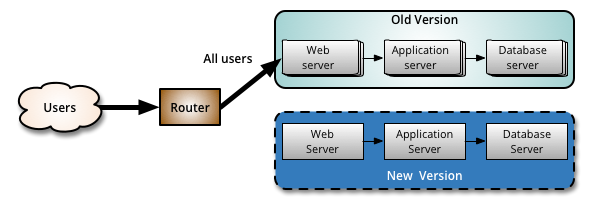
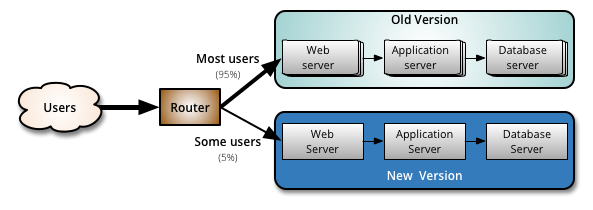
**Canary release** is a technique to reduce the risk of introducing a new software version in production by slowly rolling out the change to a small subset of users before rolling it out to the entire infrastructure and making it available to everybody.

Similar to a BlueGreenDeployment, you start by deploying the new version of your software to a subset of your infrastructure, to which no users are routed.

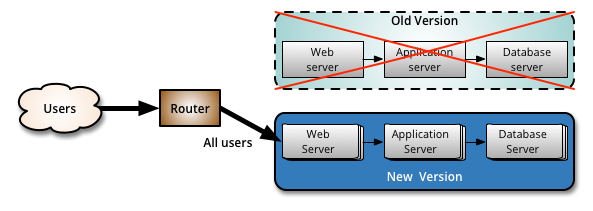


When you are happy with the new version, you can start routing a few selected users to it. There are different strategies to choose which users will see the new version: a simple strategy is to use a random sample; some companies choose to release the new version to their internal users and employees before releasing to the world; another more sophisticated approach is to choose users based on their profile and other demographics.



As you gain more confidence in the new version, you can start releasing it to more servers in your infrastructure and routing more users to it. A good practice to rollout the new version is to repurpose your existing infrastructure using PhoenixServers or to provision new infrastructure and decommission the old one using ImmutableServers.

Canary release is an application of ParalellChange, the migrate phase lasts until all the users have been routed to the new version. At that point, you can decommission the old infrastructure. If you find any problems with the new version, the rollback strategy is simply to reroute users back to the old version until you have fixed the problem.



A benefit of using canary releases is the ability to do capacity testing of the new version in a production environment with a safe rollback strategy if issues are found. By slowly ramping up the load, you can monitor and capture metrics about how the new version impacts the production environment. This is an alternative approach to creating an entirely separate capacity testing environment, because the environment will be as production-like as it can be.