**What Are Deployment Patterns?**

Deployment patterns are automated methods of introducing new application features to your users. Your ability to cut downtime depends on the deployment style you use. Some patterns also let you roll out extra functionality. Doing this allows you to test new features with a small group of users before making them available to everyone.

We have different options for deployment patterns

1. **Canary releases**

A canary release is a method of spotting possible issues before they affect all consumers. Before making a new feature available to everyone, the plan is only to show it to a select group of users. We monitor what transpires after the feature is available in a canary release. If there are issues with the release, we fix them. We transfer the canary release to the actual production environment once its stability has been established.

Canary releases represent one of the main enablers of continuous deployments.

**2. Blue/green deployments**

We have run two similar environments simultaneously, lowering risk and downtime. These surroundings are referred to as blue and green. Only one of the environments is active at any given moment. A router or load balancer that aids in traffic control is used in a blue-green implementation. The blue/green deployment also provides a quick means of performing a rollback. We switch the router back to the blue environment if anything goes wrong in the green environment.

Another variant of blue/green deployments is **red/black deployments**. The red version is live in production. We deploy the black version to one or more servers. When a Black version is operational, you switch the router to move all traffic to it. If there is an error, you revert it. What is the difference between blue/green deployments? While in blue/green deployments, both versions may temporarily get requests simultaneously, in red/black, only one version gets traffic at any time.

Red/black deployments is a newer term than blue/green, used by different companies nowadays, and could be used as a synonym.

**3. Feature toggles**

Here, we can turn a switch on/off with feature toggles at runtime. We may roll out new software without exposing our users to any other brand-new or modified functionality. When we build new functionality, we can use feature toggles to enable continuous deployments by splitting releases from deployments.

**4. A/B testing**

Two versions of an app are compared using A/B testing to see which one performs better. An experiment is like A/B testing. In A/B testing, we randomly present users with two or more page versions. Then, we use statistical analysis to determine which variant is more effective in achieving our objectives.

**5. Dark launches**

In a “dark launch,” we introduce a new feature to a select group of users rather than the general public. These users must be aware that they are helping us test the functionality. We need to point out the new functionality to them. It is nicknamed a “dark launch” for this reason. Users are introduced to the program to get feedback and test its effectiveness.