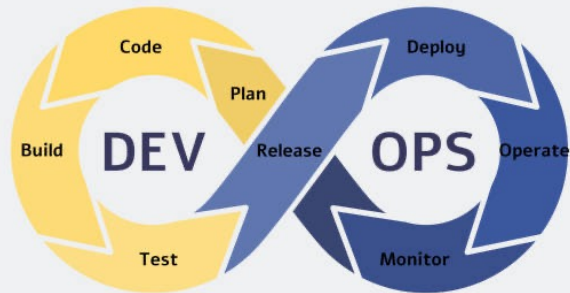


The background is a solid blue gradient. Overlaid on this are numerous thin, white, curved lines that flow from the left side towards the right, creating a sense of motion and depth. These lines vary in density and curvature, some forming gentle waves while others are more pronounced and steep.

GETTING STARTED WITH CI/CD

CONTINUOUS INTEGRATION/CONTINUOUS DEPLOYMENT(CI/CD)



Continuous Integration

*The practice of merging all developers' working copies to a shared mainline several times a day. It's the process of "**Making**".*

Continuous Deployment

A software engineering approach in which the value is delivered frequently through automated deployments.

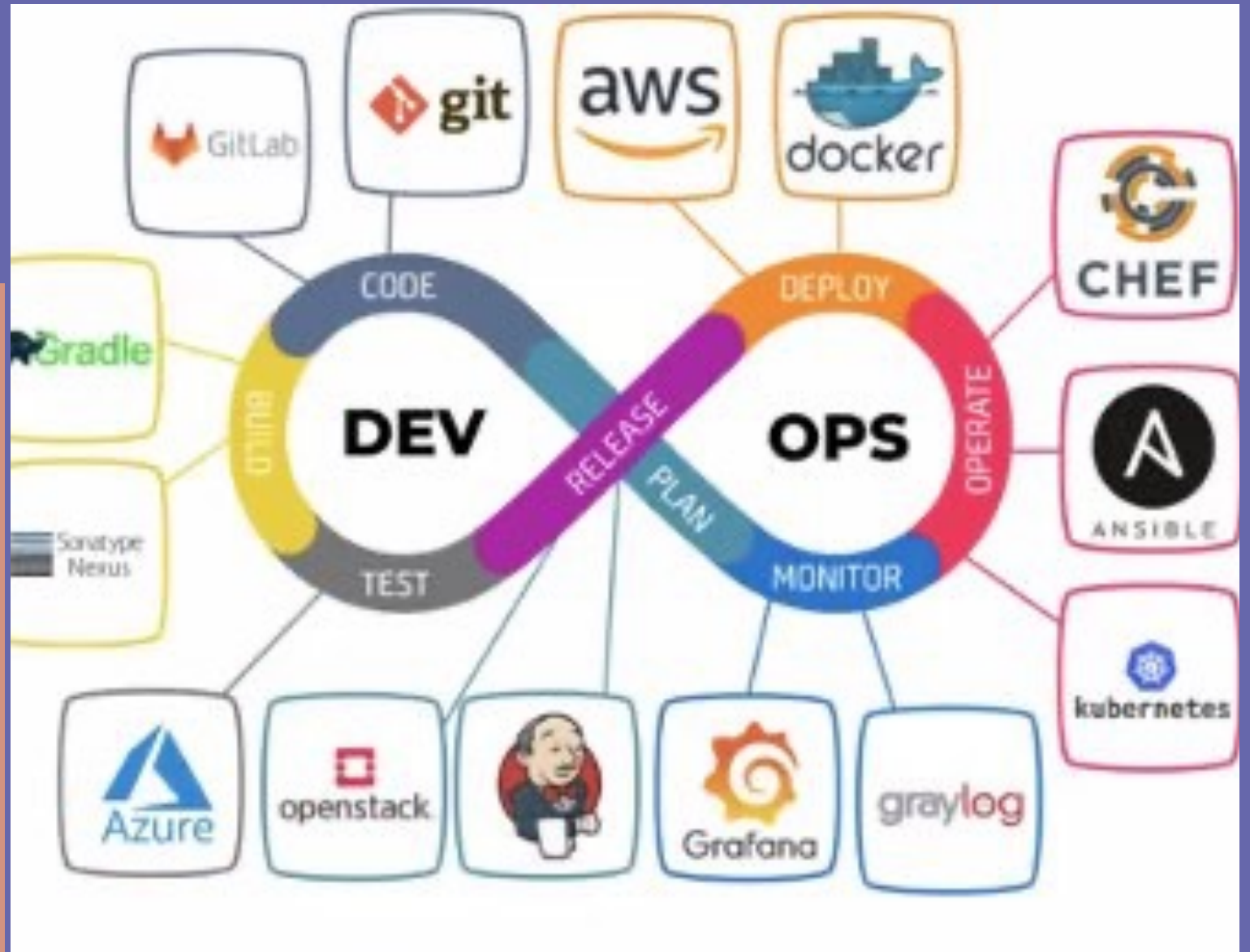


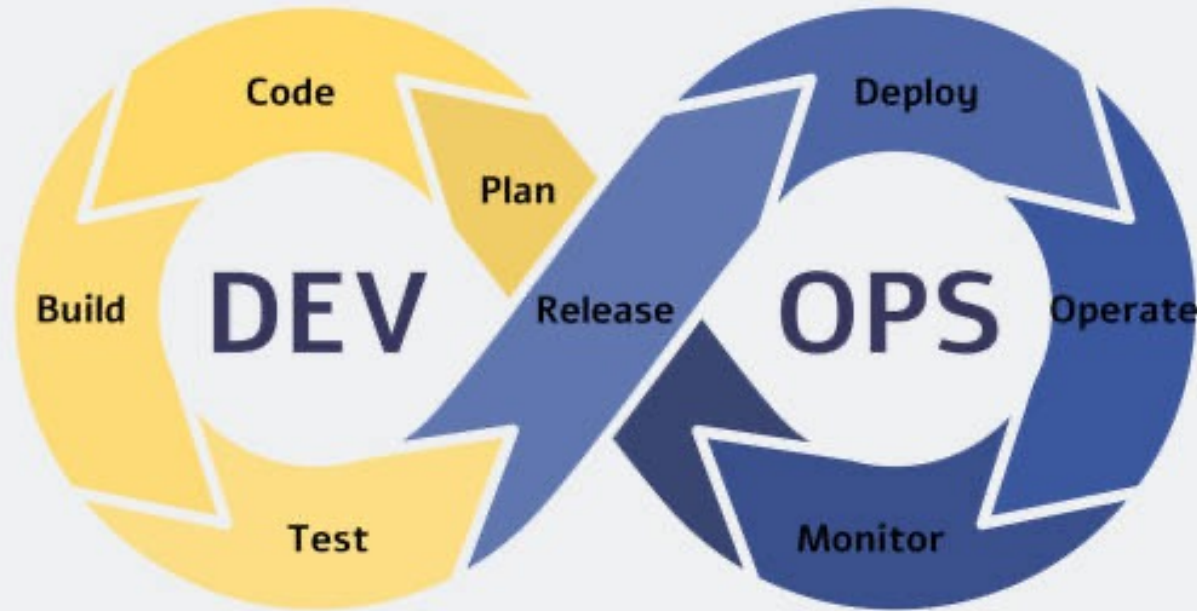
WHY CI/CD?

Value	Translation
Reduce Cost	Less developer time on issues from new developer code
Avoid Cost	Less bugs in production and less time in testing
Avoid Cost	Prevent embarrassing or costly security holes
Avoid Cost	Less human error, Faster deployments
Reduce Cost	Less infrastructure costs from unused resources
Increase Revenue	New value-generating features released more quickly
Increase Revenue	Less time to market
Protect Revenue	Reduced downtime from a deploy-related crash or major bug
Protect Revenue	Quick undo to return production to working state

CI/CD TOOLS

CI/CD tools typically support a marketplace of plugins. For example, Jenkins lists more than 1,800 plugins that support integration with third-party platforms, user interface, administration, source code management, and build management.





CONCLUSION

- To recap, *continuous integration* packages and tests software builds and alerts developers if their changes fail any unit tests. *Continuous delivery* is the automation that delivers applications, services, and other technology deployments to the runtime infrastructure and may execute additional tests.
- Developing a CI/CD pipeline is a standard practice for businesses that frequently improve applications and require a reliable delivery process.
- Getting started with CI/CD requires devops teams to collaborate on technologies, practices, and priorities. Teams need to develop consensus on the right approach for their business and technologies. Once a pipeline is in place, the team should follow CI/CD practices consistently.