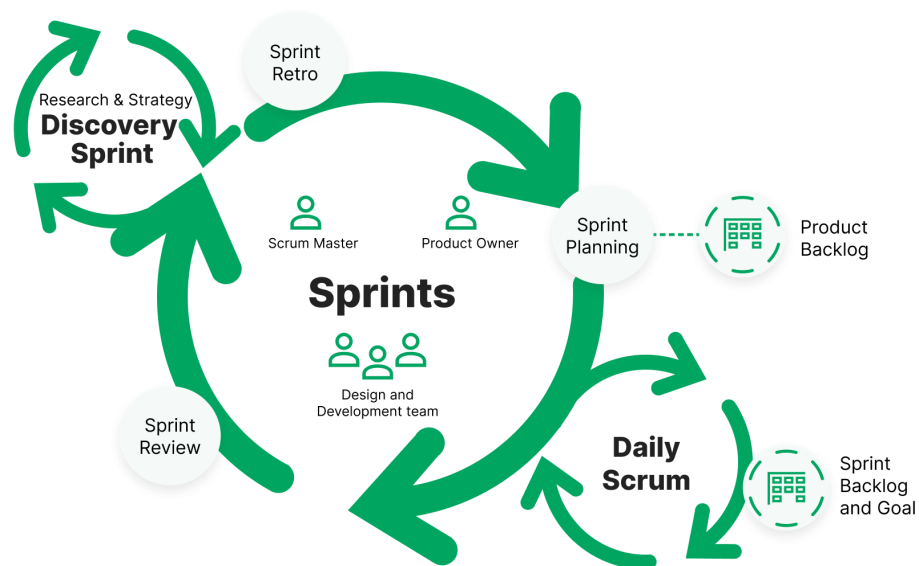




Agile and DevOps

▼ What is Agile?



- **Iterative and Incremental Development:** Software is developed in small, workable increments.
- **Customer-Centric:** Constant feedback from customers to refine requirements.
- **Frequent Delivery:** Rapid releases of smaller, functional product versions.
- **Adaptability:** Agile responds to change quickly.

▼ Agile Principles

- **Individuals and Interactions:** Over processes and tools.
- **Working Software:** Over comprehensive documentation.
- **Customer Collaboration:** Over contract negotiation.
- **Responding to Change:** Over following a plan.
- **Quote:** "The highest priority is to satisfy the customer through early and continuous delivery of valuable software."



Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it. These are our values and principles.

<https://agilemanifesto.org/>

▼ Agile Frameworks

• Agile Methodologies and Frameworks

- **Scrum:** Divides work into sprints (2-4 weeks) with regular standups and reviews.
- **Kanban:** Focuses on visualising workflow and limiting work-in-progress (WIP).
- **XP (Extreme Programming):** Emphasizes technical excellence and frequent releases.
- **Lean Development:** Focuses on minimizing waste and maximizing value.

What is Agile? | Atlassian

Learn agile software development, agile methodologies and industry best practices from beginner tutorials to advanced topics.

<https://www.atlassian.com/agile>



What is Agile Software Development?

Agile gives organizations the ability to quickly create and respond to change in today's disruptive marketplace. Learn more at AgileAlliance.org.

<https://www.agilealliance.org/agile101/>



Agile Project Management

Offered by Google. This is the fifth course in the ... Enroll for free.

https://www.coursera.org/learn/agile-project-management?trk_ref=article
ProductCard



Scrum Vs. Kanban: How to decide when to use Scrum and when to use Kanban

Many teams and organizations have leveraged a combination of practices from both scrum and kanban, sometimes to work to their advantage, sometimes not so much.

<https://resources.scrumalliance.org/Article/scrum-vs-kanban>



▼ What is DevOps?

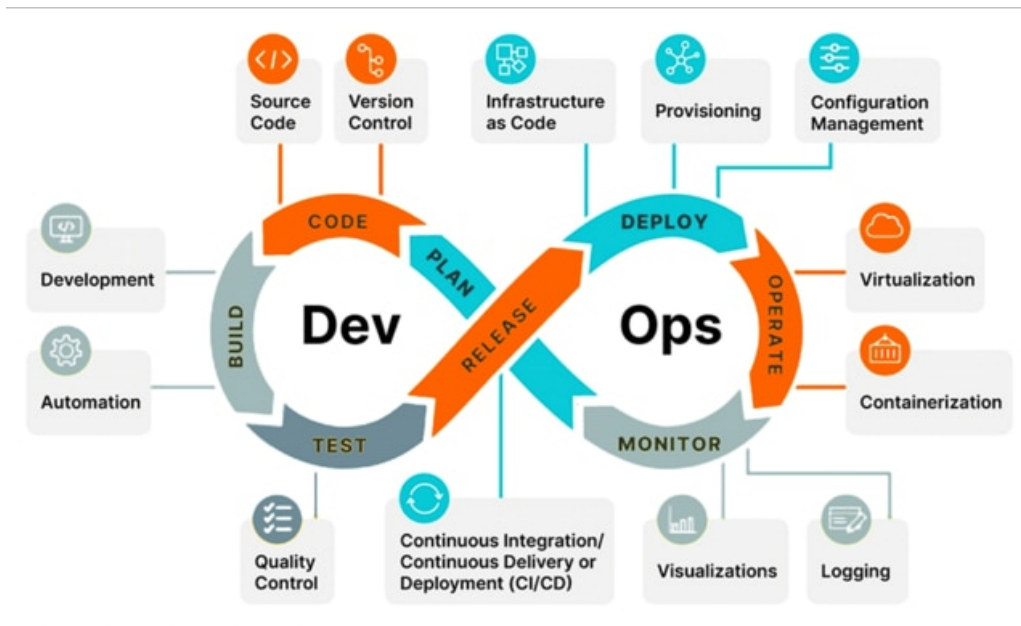


- **DevOps = Development + Operations:** A culture and set of practices that integrates development (Dev) and operations (Ops).
- **Collaboration and Automation:** Between developers and IT operations for faster delivery of high-quality software.
- **Continuous Integration/Continuous Delivery (CI/CD):** Automates code testing and deployment.

▼ DevOps Core Practices

- **CI/CD Pipelines:** Automating the building, testing, and deployment of code.

- **Infrastructure as Code (IaC):** Managing infrastructure through code (e.g., Terraform, Ansible).
- **Monitoring and Logging:** Ensures system reliability through real-time tracking and analysis.
- **Collaboration and Communication:** Cross-functional teams sharing ownership of development and operations tasks.



▼ Key Differences between Agile and DevOps

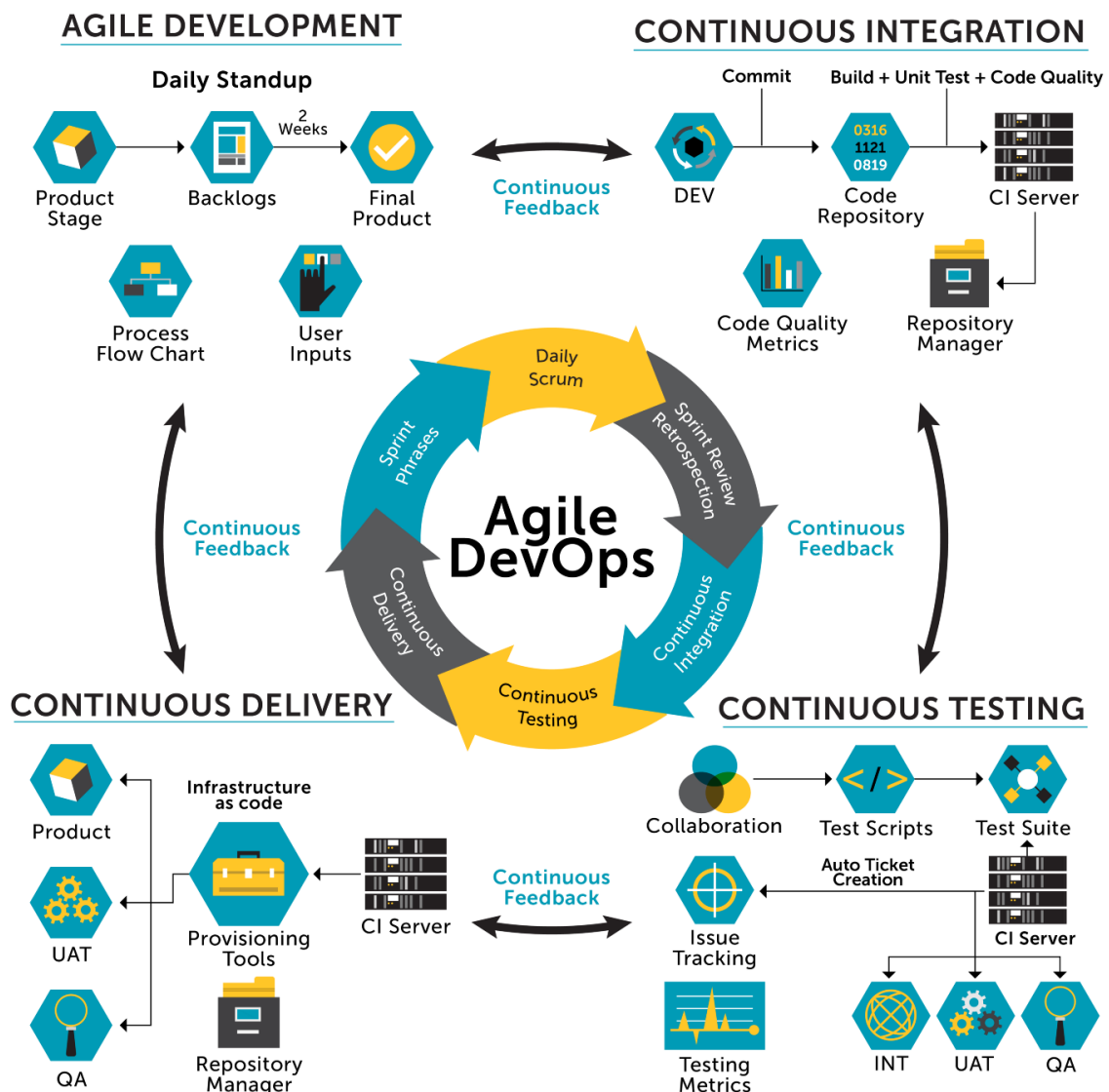
	DevOps	Agile
Focus	Collaboration between dev and ops teams	Frequent customer feedback
Iterations	Rapid feedback loops	Iterative cycles
Scope	Large-scale projects	Smaller, incremental changes
Tools	Automation tools (e.g. Jenkins)	Task management software (e.g. Jira)
Frameworks	Kanban, DevOps lifecycle	Scrum, XP

- **Agile:** Focuses on iterative development and customer feedback.
- **DevOps:** Focuses on automating delivery, collaboration, and integration between dev and ops teams.
- **Agile:** Has a shorter feedback loop.
- **DevOps:** Integrates the entire process for faster releases.

▼ Why DevOps Complements Agile

- Agile improves development velocity, but DevOps extends the concept to deployment and maintenance.
- Both are customer-focused, but DevOps ensures rapid and reliable deployment in addition to development.
- DevOps fills gaps Agile doesn't cover, like operations, infrastructure, and automation.

▼ Agile + DevOps in Action



- **Agile:** Helps development teams iterate and adapt to changing requirements.
- **DevOps:** Bridges the gap between developers and IT operations.
- Example: **Netflix** and **Spotify** use Agile methodologies combined with DevOps for continuous delivery and high service availability.

▼ Benefits of Agile and DevOps

- Faster, more frequent delivery of features.

- Improved communication and collaboration between teams.
 - Reduced risk of deployment errors.
 - Ability to adapt to customer feedback and market changes rapidly.
 - Higher-quality software and reduced time-to-market.
-