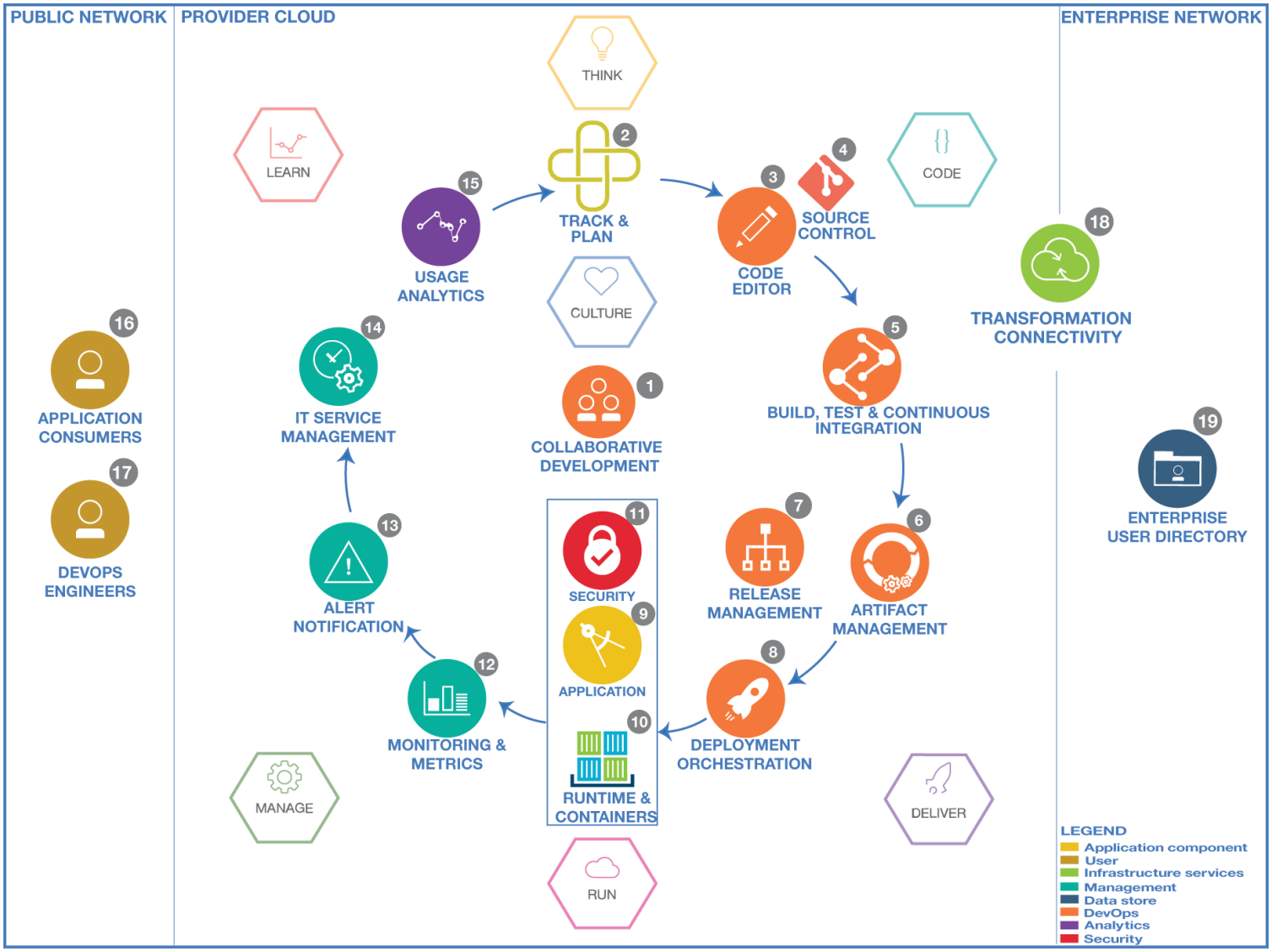
DevOps architecture overview

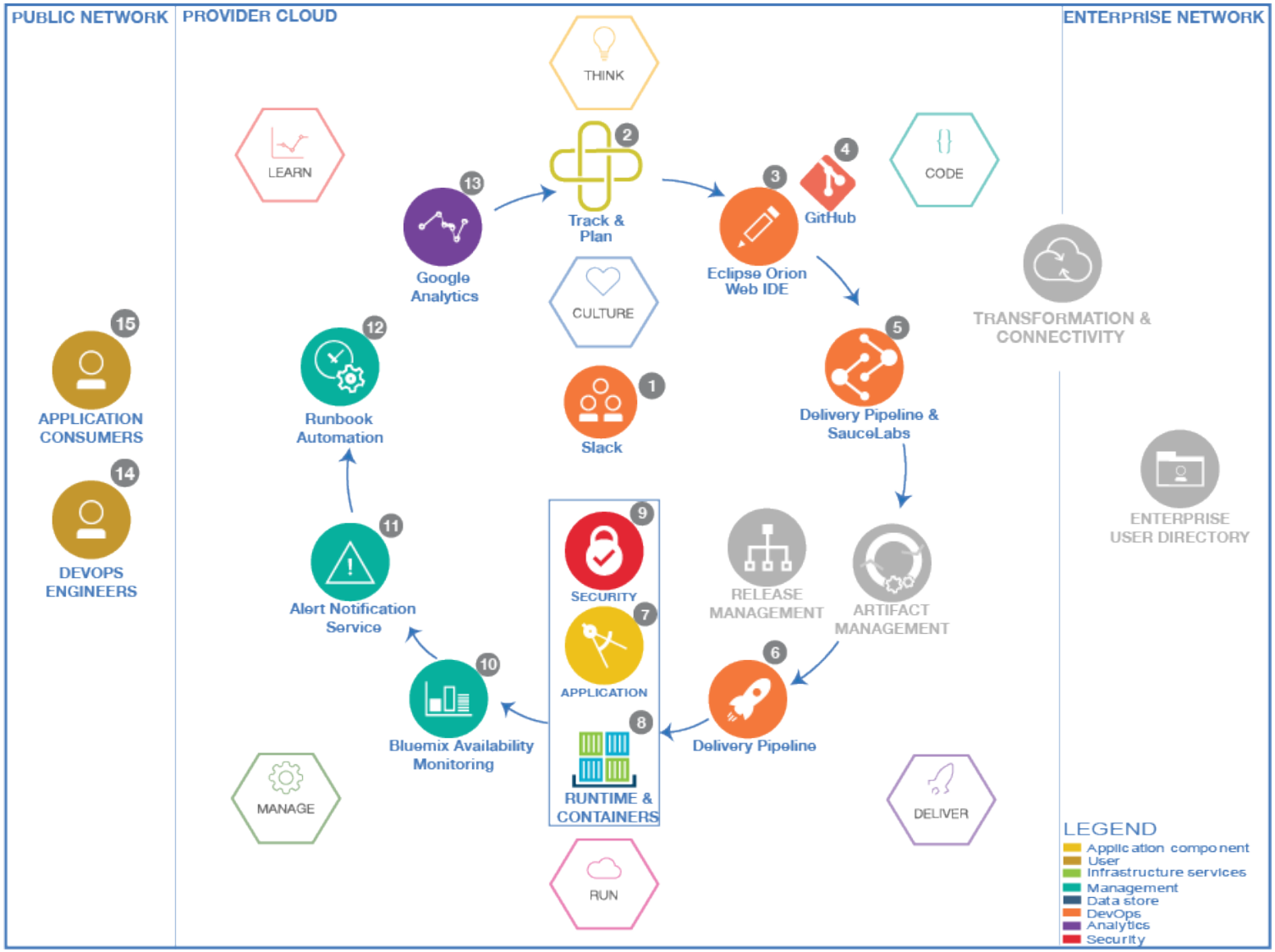
The right practices and toolchains to rapidly execute and scale. IBM Bluemix Garage Method takes the best of Design Thinking, Lean, Agile, DevOps and Cloud to accelerate all phases of the application design, development and delivery lifecycle.



IBM Cloud Architecture Center

Cloud-native implementation

This reference implementation provides a toolchain that enables a team to collaborate and track development, as well as to build, deploy, manage, and learn about the product being delivered.



Runtime flow

1. Collaboration tooling is required to enable a culture of innovation. Developers, Designers, Ops teams, managers will need to be in constant communication. Dev & Ops tooling should also be integrated to post updates as new builds complete, are deployed, and monitoring alerts are fired. The team can discuss these alerts as a group in the context of the tool.
2. As the team thinks of new ideas, responds to feedback & metrics, and fixes defects - work items are created and then prioritized into a ranked backlog. The team will work on items from the top of the list, delivering to production as work completes.
3. Developers now write source code using a code editor to implement the architecture. They need to construct, change and correct applications using a variety of coding models and tools.
4. Source Code Management is used to manage the versions and configuration of assets, merge changes and manage the integration of changes. The tool should support open source style social coding.
5. Developers will compile, package and prepare software assets. The tools should be able to assess quality of code delivered to source control by individual developers and teams. These assessments are performed prior to delivery and associated with automate build systems and includes practices such as code reviews, unit tests, code quality scans, security scans.
6. Binaries and other output from the build are now sent to and managed in a Build Artifact Repository.
7. The release is now scheduled. Tools should support managing, preparing, and deploying releases and release communications.
8. Performing coordination of the manual & automated processes required for the solution to operate effectively is done. The team should strive towards continuous delivery with zero downtime. A/B deployments can help gauge the effectiveness of new changes.
9. Understand the application and options for an application's runtime environment, security, management and release requirements
10. Depending on the application requirements, some or all of the application stack will need to be considered including middleware, the operating system, and virtual machines.
11. Take steps to ensure all aspects of the application and supporting infrastructure is secured.
12. Plan, configure, monitor, define criteria and report on application availability & performance. Predictive Analytics can help to warn of problems before they occur.
13. Alert the correct people or systems when important events occur from monitoring or other systems that emit events.
14. Manage the process for responding to Ops incidents, and delivering the changes required to fix them.
15. Learn how users interact with the application, and the metrics required to measure success using analytics.
16. When the user interacts with the application they can provide feedback on their requirements & how the application is meeting them which is captured by analytics as well.
17. DevOps Engineers will manage the entire application lifecycle while responding to feedback and analytics collected from the running application.
18. The Enterprise network is protected by a firewall, and must be accessed via Transformation & Connectivity services and secure messaging services.
19. The user directory is used by security throughout the flow and contains information about the user accounts for the enterprise.

Components

COMPONENT	DEFINITION	IBM PRODUCT
Collaborative Development	Enables team communication and integrates with DevOps tools for in context discussions.	IBM Connections and Verse, Slack
Track & Plan	Enables work item backlog management.	IBM DevOps Services Track & Plan, Rational Team Concert
Code Editor	Enables developers to write source code usually with a developer environment	Bluemix Web IDE, Rational Application Developer, Eclipse, Atom, Sublime
Source Control	Enables source code management & versioning.	IBM DevOps Services Git, GitHub Enterprise, GitHub
Build, Test, & Continuous Integration	Enables the Compile, package, test, and preparation of software assets.	IBM DevOps Services Delivery Pipeline, Rational Test Workbench, Rational Team Concert, Rational Functional Tester, Rational Performance Tester, UrbanCode Build, UrbanCode Deploy
Artifact Management	Enables management of the output from the build.	Bluemix DevOps Services, Rational Asset Manager, UrbanCode Deploy
Release Management	Enables management, preparation, and deployment of releases.	IBM UrbanCode Release
Deployment Orchestration	Enables processes required to get the release into production.	IBM UrbanCode Deploy, Delivery Pipeline, Globalization Pipeline Active Deploy, IBM Cloud Orchestrator
Application	Represents the domain specific application being developed using devops method	
Runtimes & Containers	Enables an application to run in a runtime environment.	Bluemix runtimes (Nodejs,, JavaLiberty etc.), OpenStack virtual machines, IBM Containers, Auto-Scaling
Security	Enables securing the entire application stack & infrastructure.	Access Trail, Application Security on Cloud, Single Sign On
Monitoring & Metrics	Enables monitoring the entire application stack & infrastructure.	Bluemix Monitoring and Analytics, Bluemix Availability Monitoring, Bluemix Alert Notification, IBM Application Performance Monitoring, New Relic, Mobile Quality Assurance
Alert Notification	Enables alerting the correct team members and systems when events occur.	IBM Alert Notification (See Incident Management Architecture)
IT Service Management	Enables managing the operations of the application stack and infrastructure.	IBM Control Desk, ServiceNow (See Service Management Architecture)
Usage Analytics	Enables understanding how users interact with the system.	IBM Digital Analytics
Application Consumer	Represents the persona that interact with the system.	
DevOps Engineers	Represents the persona managing the entire lifecycle.	
Transformation & Connectivity	Enables services to access hybrid cloud resources.	Bluemix Secure Gateway, Bluemix API Connect, IBM API Connect
Enterprise User Directory	Enables a directory of user accounts to be accessed.	IBM Security Directory Server

Business drivers

01

Line of business leaders need to experiment and quickly determine best alternatives for new business capabilities.

02

DevOps enables the business to experiment with entering new markets, introducing new products, and differentiate existing products with enhancements.

03

DevOps encourages rapid delivery and rapid feedback for quick adjustments.

Requirements

Speed to value

- Enterprises require speed and predictable lead times for application delivery initiatives. The lead time for delivery of business capabilities relates directly to market opportunity.

Enhanced feedback

- Business leaders require constant feedback to learn if market experiments are succeeding and to make quick adjustments. Technical teams need to learn early if there are performance issues with deployed applications.

Secure and managed APIs

- Digital platforms and the innovation edge of an enterprise require secure connections from cloud to backend systems and managed APIs to enable effective and controlled access from external applications.

Flexible architecture

- Application and services architecture must be flexible, de-coupled, and enable independent deployments of business capabilities.

For more information and for the latest IBM solutions and other assets, visit developer.ibm.com/architecture