Environments as a Service Platform

On-demand & Ephemeral Pre-production Environments

While in the cloud-native world a lot of infrastructure has been slowly abstracted and automated, but pre-production environments are still created using hand-crafted scripts. To make matters worse, because of shared development and test sites, current software development processes work sequentially; forcing developers to wait for an environment to become available in order to test and validate a change. This grinding process delays releases.
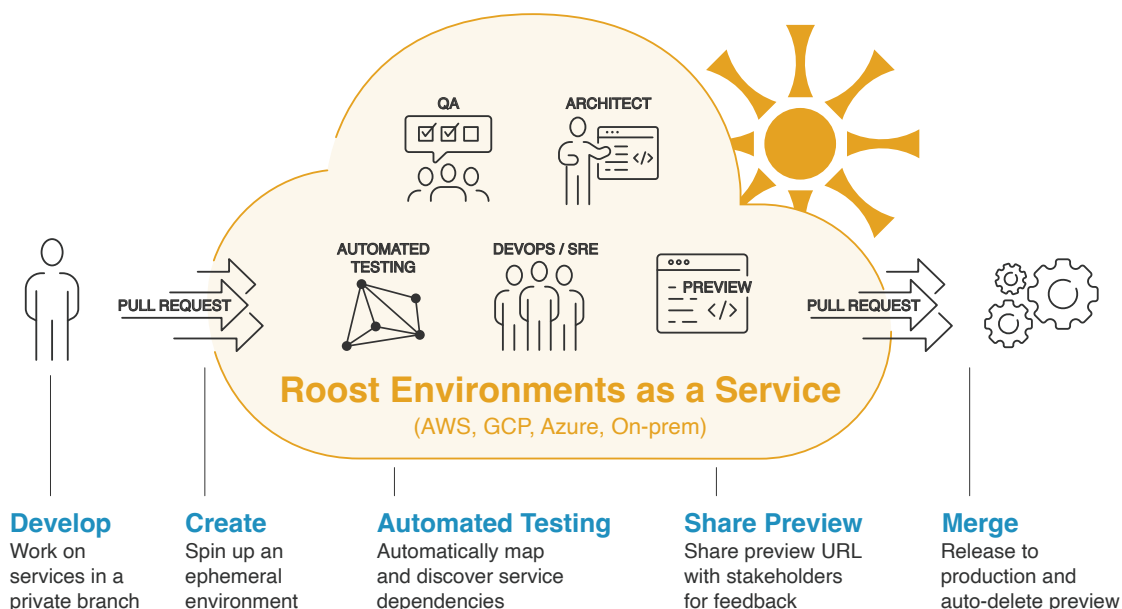
With the Roost platform, engineers no longer need to worry about creating environments or custom scripts for testing. Instead, they can focus on writing code because environments and testing are handled automatically and transparently.

Since the Roost environment is defined by a pull request (PR), it is isolated and private to the developer, so numerous PRs can be run in parallel. This avoids testing bottlenecks and increases release velocity.
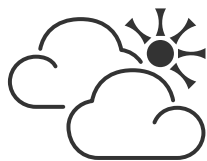
No more staging environments. No more long queues.

Achieve Real-time Release Velocity for Every Change

By inspecting source-code repositories, the Roost Platform auto-discovers environment configuration and automatically tests code changes using the latest versioning.

Roost Environments as a Service
(AWS, GCP, Azure, On-prem)

Develop
Work on services in a private branch

Create
Spin up an ephemeral environment

Automated Testing
Automatically map and discover service dependencies

Share Preview
Share preview URL with stakeholders for feedback

Merge
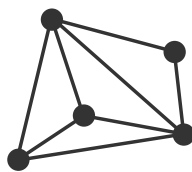Release to production and auto-delete preview

Core Capabilities of the Roost Platform

Ephemeral Pre-production Environments

Instantly create an ephemeral pre-production environment at every pull request, feature branch, or insertion point in the DevOps / GitOps pipeline using the same micro services, containers, sidecars, etc. as production.

AI-enabled Auto-Discovery & Automated Testing

The Roost platform continuously scans source-code repositories and discovers environment configuration and automatically tests changes using the latest versioning.

This PROPRIETARY PROCESS is Roost's "secret sauce" to avoid integration issues later in production.

Sharable Preview Environments

With Roost preview environments, developers share a custom URL with stake-holders so they can review changes before it is shipped to production. Stakeholders become part of the review and development process right from the start.

Roost Platform Differentiators

Automated Scanning: Automatically maps & discovers environment configuration (e.g. Helm charts, Docker files, Kubernetes YAML, Terraform) required to instantly test and validate code changes.

Fast and Accurate Testing: Source code repositories (e.g. GitHub, GitLab, BitBucket) are continuously scanned ensuring change validation testing is done upon request and performed with the latest versioning.

Reduce Costs: Avoid running environments 24/7 by scheduling length of deployment time. Eliminate hours required for custom script creation done by each individual developers.

Accelerate Release Cycles: Since the Roost ephemeral environment is defined by a pull request and private to the developer numerous PRs can be run in parallel. No more QA bottlenecks!

Fast Feedback: Stakeholders (QA, product, engineers, architects, SREs, etc.) can review and validate changes BEFORE a pull request is merged. No more surprises in production!

Streamline Collaboration: Developers can share preview environments with stakeholders (via a custom URL) so they become part of the development process.

Roost is an Environments as a Service platform that on-demand creates an ephemeral pre-production environment that automatically accesses all required containers and micro services required to test and validate code changes.

080122