**Progress® Chef**

Chef Edge Management

Eliminate application delivery failures, accelerate deployment frequency, and drive business agility to the edge.

Progress Chef

Progress® Chef® Edge Management is a combination of automation tools enabling companies to apply a technology-agnostic and modular approach to defining, packaging and delivering applications at the edge. The solution further helps in eliminating deployment failures, accelerating release schedules and configuring security & compliance profiles for your edge devices.

Large enterprises in industries like manufacturing, retail, healthcare, etc. use Chef Edge Management to improve the quality and frequency of their application deployments on edge devices. The solution further helps in decreasing operational overhead, shortening the testing cycles and increasing the productivity of enterprise IT teams.
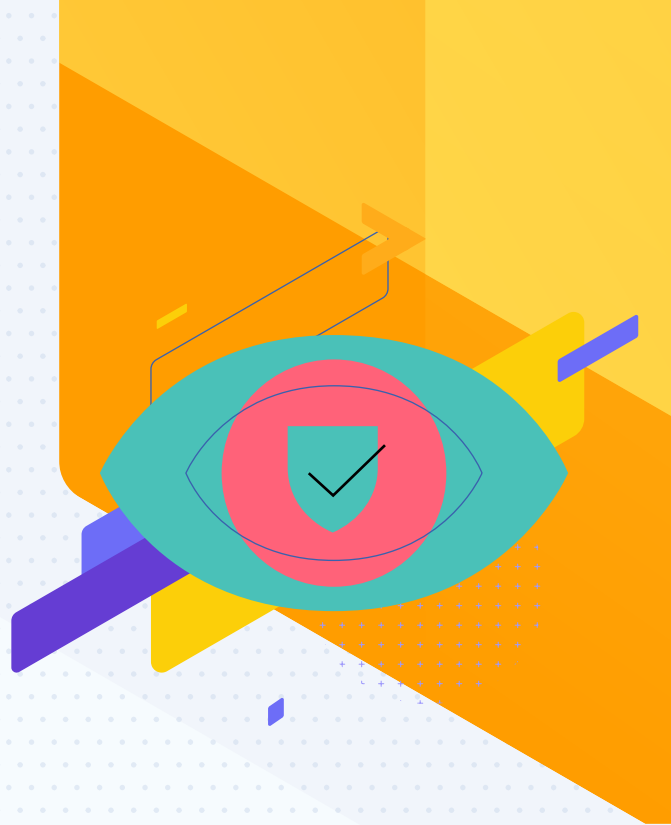
Key Features

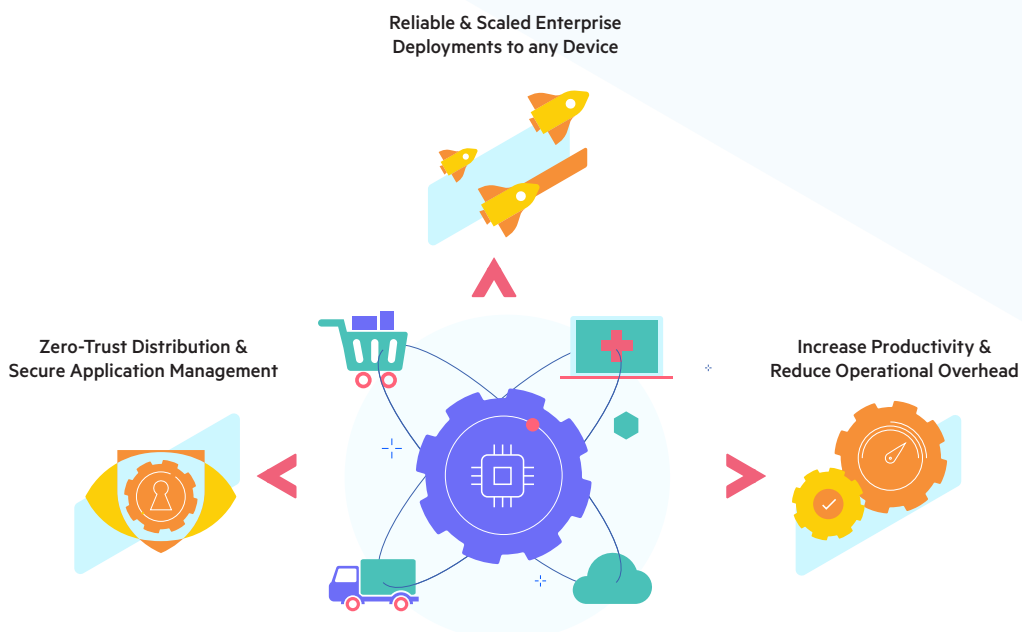### Zero-Trust Distribution & Secure Application Management

- Enhanced device and application security

- All modifications are auditable in code

- Continuous compliance and remediation

- Pull based deployments where an agent periodically pulls package updates, monitors compliance and remediates configurations periodically to ensure everything is always up to date

Reliable & Scaled Enterprise Deployments to Any Device Located Anywhere

- Lightweight, immutable, portable & discrete application artifacts

- Peer-to-Peer distribution capabilities

- Visibility into the deployment status across the entire ecosystem of devices

- Single toolchain & process for the cloud/data center as well as the edge

- Pre-packaged content for creating immutable application artifacts

Increase Productivity & Reduce Operational Overhead

- Easy rollbacks, reduced testing cycles & application rebuilds

- Flexibility & fine-grained control without disrupting business operations

- Minimize the amount of effort and time required to maintain applications over their lifespan

- One agent for deployment, monitoring and service management

Reliable & Scaled Enterprise Deployments to any Device

Zero-Trust Distribution & Secure Application Management

Increase Productivity & Reduce Operational Overhead

High-level Deployment Workflow

Chef Edge Management helps standardize the processes for development, operations and security by providing a single workflow and toolchain to build, deploy and run applications, manage device configurations and security on a massive scale at the edge.
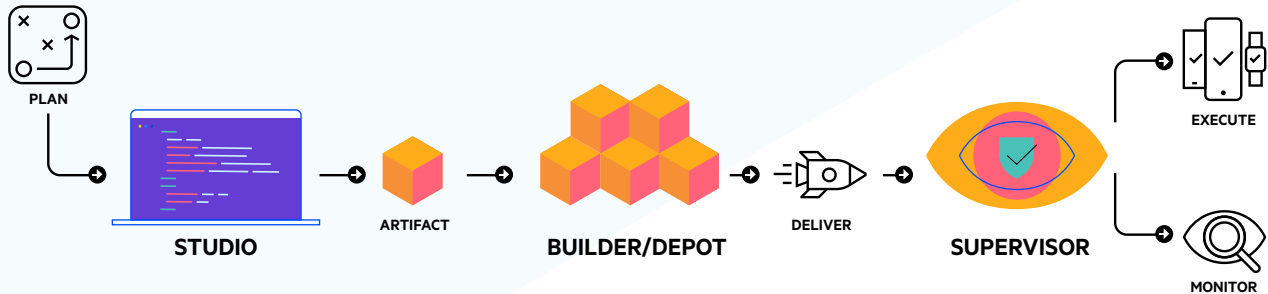

The diagram above shows the basic architectural flow of Chef implementation at the Edge. The first step is defining a plan file for the workload (could be a custom or third-party application, configuration, or compliance policies) to be deployed. **The Plan file** declares everything it needs – build and run time dependencies, instructions to install, pre-check conditions, etc. All this information is packaged into a signed immutable **artifact**, made within a clean-room environment—**the Studio**—, and then uploaded to the **Builder/Depot**. The plans can be easily readable and written, as they are based on Bash/PowerShell.

Furthermore, Chef offers and maintains a substantial collection (750+) of pre-packaged libraries for both Windows and Linux. Users can start utilizing these readily available libraries in their plan files as runtime and build time dependencies, enabling new users to begin quickly. Additionally, one can leverage the vast amount of available community content for defining configurations and compliance policies. The artifacts built with Chef are exceptionally lightweight, as they solely include the necessary dependencies. Artifacts remain discrete and

run consistently on all supported devices because they are unaffected, even if a specific version of a library required for the application is absent on the device. An artifact encompasses all dependencies with desired versions bundled within and does not need to rely on device libraries.

The **Supervisor** is a process manager who starts and monitors services defined in the plan file. It also receives and acts upon information from the other Supervisors to which it is connected. The artifacts uploaded to the builder are pulled by the Supervisor running on the device, which **executes and monitors** the application on the device. The Supervisor can connect to additional supervisors from other devices to form a ring and share configurations and updates with each other, enabling peer-to-peer communication.
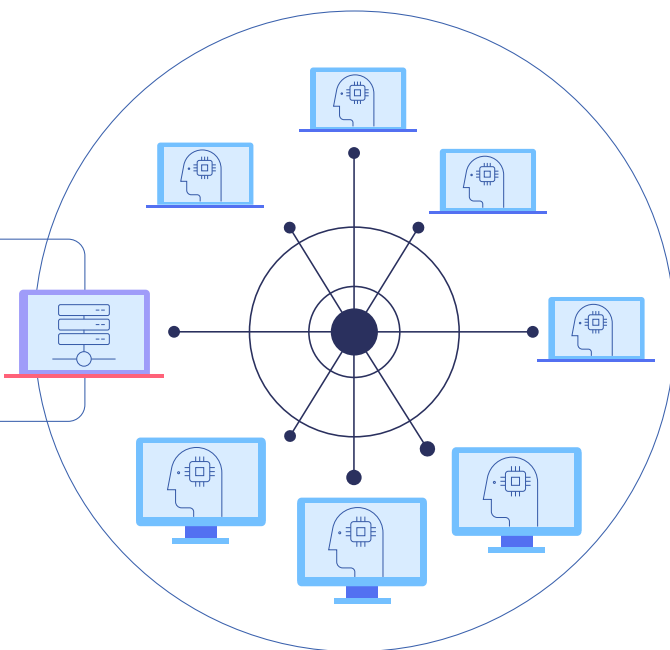
With Chef Edge Management, users can leverage peer-to-peer communication along with depot and packaging capabilities to set up devices in a particular manner. This minimizes the number of devices requiring an internet connection in each location, thereby reducing the threat surface area. This approach provides two benefits – First, it mitigates download failures for devices operating in isolated network environments with limited computing resources and low memory. Second, there are fewer downloads to perform since the Supervisors connected to each other take care of the required updates.

Define in code what the app needs to be built, deployed and maintained.
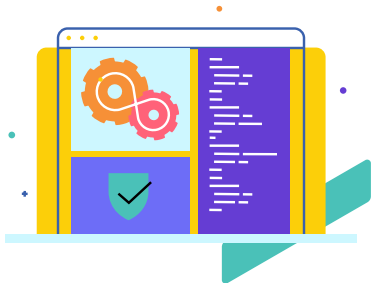
Use **whatever tools** you want to for pipelining.
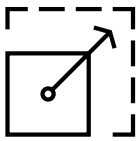
Deliver to any environment.
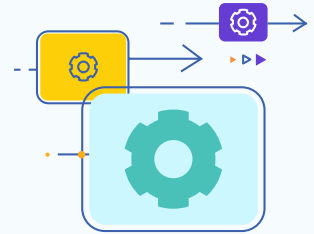
Package a single artifact that can run anywhere.

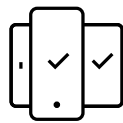**Manage** deployment with channels and patterns.

Validate in real time.

Chef Edge Management Benefits

Portability & Scalability

Lightweight immutable artifacts with all the dependencies, configurations and preconditions bundled within ensure that deployments in production are consistent, reliable and secure even at a massive scale.

Reliability & Consistency

Application artifacts that provide a guaranteed runtime that can be reliably deployed and managed in any device and any location including low bandwidth environments.

Shortened Test Cycles

Since all the dependencies are bundled within, these artifacts are portable and can run consistently in any environment. As a result, these artifacts help eliminate the need to test in multiple environments and avoid long testing cycles.

Reduced Operation Overhead

A single lightweight agent that runs on individual edge devices enables application execution, management and overall monitoring of the ecosystem. This means reduced operational overhead in the form of fleet maintenance and real-time visibility into failures and fleet status.

Flexibility

A Policy-as-Code approach, combined with support for building applications with required dependencies, provides immense flexibility to create application artifacts that adhere to business requirements. Additionally, the support for custom deployment patterns enables controlled upgrades and installations without interrupting business operations.
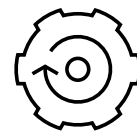
Quick Rollbacks

Perform easy and quick rollbacks when problematic code changes are deployed without manual interventions or field visits thus preventing interruptions in business service delivery.

Transparency

The visibility dashboard offers real-time visibility across all environments, allowing teams to organize and display data from different devices intuitively.

Security

Configure settings, perform periodic compliance checks and automatically fix misconfigurations so that devices remain continuously secure and compliant. Peer communication capability also ensures that lesser data is transmitted with fewer devices connected to the central server/internet. Lesser the data transmitted, lower is the security risk.
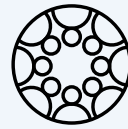
Pre-Packaged & Community Content

Access an extensive collection of curated Plan files, that serve as dependencies for creating application artifacts and CIS benchmark policies maintained by Chef. Additionally, tap into community-contributed content for configuration and compliance policies.

Transform Software Distribution at the Edge with Chef

Achieve 99.999% successful deployments.
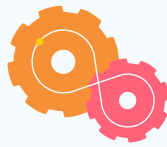
Accelerate deployments from weeks to hours, and save thousands of hours of rework.

Deploy apps that are consistent, reliable and secure.

Get Started Today
Visit the web site to learn more.

About Progress

facebook.com/getchefdotcom
twitter.com/chef
youtube.com/getchef
linkedin.com/company/chef-software
learn.chef.io
github.com/chef
twitch.tv/chefsoftware

Progress®