Login

Red Hat official product list

This list defines the different types and categories Red Hat uses to organize our products and technologies.

Note

Within our naming guidance, we use "product" as a catch-all term, versus "market product," defined by the Product MDM team specifically as standalone technology we sell separately. Refer to the market product entry for more information.

add-on

An add-on comprises service(s), component(s), feature(s), and/or tool(s) to add new or improved functionality to a base platform/product. Add-ons are sold or attained separately (not built in) and require the base product to use.

Note

"Add-On" is no longer included in (public-facing) product names.

Examples:

- **Red Hat® Ansible® Network Automation:** Includes automation execution environment (component), automation controller (component), and support for specific networking modules (service). Designed for/requires Red Hat Ansible Automation Platform.
- **Red Hat Developer Toolset:** Additional tools that developers can use when to speed Linux application development, diagnose issues, and debug application's code. Designed for/requires Red Hat Enterprise Linux.

bundle

A bundle is a set of 2 or more existing, independent offerings marketed together and sold through a single SKU for easier selling. Calling something a bundle does not signify integration; it is merely the ability to buy multiple SKU'd products, services, add-ons, etc. together, sometimes for a discount.

Examples:

- Any promotional bundle (buy X and Y together, save $Z)
- Red Hat OpenShift® Platform Plus

Note

"Bundle" CANNOT be used as part of a (public-facing) product name.

collection

A collection is a group of non-market products (runtimes, tools, etc.) that are offered or delivered together, usually for developer purposes. A collection can have a SKU. Unlike a bundle, the parts of a collection do not need to be independently available offerings.

Examples:

- **Red Hat JBoss® Core Services Collection:** A set of additional functionality, management, and tools for Red Hat Application Services products, including web connectors, Red Hat's single sign-on (SSO) technology, Apache HTTP server, and Apache Jsvc. Included with most Red Hat JBoss Enterprise Application Platform subscriptions.
- **Red Hat Runtimes:** A single SKU that includes a set of other market products, tools, and components.

Note

We no longer use "suite" in (public-facing) product names. Do not use the term "suite" to refer to a collection descriptively.

community

The foundation of the open source model. A named group of people—either volunteer or paid—who work together to produce an open source project. Often called the same thing as the name of the technology they're creating. Some of these communities are either created, run, or sponsored (aka funded) by Red Hat; others Red Hatters just participate in to represent Red Hat's interests in creating and developing features.

Examples:

- The Fedora community includes Red Hatters, hobbyists, people from other companies paid to help build and maintain Fedora, the Linux operating system.
- The Ansible community includes Red Hatters, hobbyists, and people from other companies paid to help build and maintain the source technology for Ansible automation. For Red Hat, the end product is Red Hat Ansible Automation Platform.

Note

Red Hat usually only names the communities we create. When creating short names for market products or tools that are based on community technology, be cautious to avoid confusion with the community projects or issues with intellectual property.

component

A piece of technology that is combined with others to comprise a whole base platform/market product or tool. A component must be part of a market product, operator, tool, or service and is never standalone or sold separately.

Examples:

- **Red Hat OpenShift Service Mesh:** The code within OpenShift Container Platform that lets the platform support applications running in a distributed microservices architecture and lets those microservices interact with each other.

- **automation controller:** The management interface component that controls the Ansible environment. It includes a user interface (UI), RESTFUL application programming interface (API), role-based access control (RBAC), workflows, and continuous integration and continuous delivery (CI/CD) integrations.

Note

Components should not be given Red Hat-branded names (unless required for specific marketing exceptions, such as the OpenShift Service Mesh example above).

Developer Preview

Similar to the more general Technology Preview, features or products in Developer Preview release status are available for early access to the developer community, so that they can test functionality and provide feedback during the development process. However, these features or products are not fully supported, and documentation might be incomplete. Usually followed by general availability (GA) or limited availability.

Related: Service Preview, Technology Preview

family

A family is a descriptive, often shorthand name for related products, components, tools, services, add-ons, etc. Some market product names are used to refer to their families.

Examples:

- **Red Hat OpenShift:** Used to refer to any or all of the OpenShift products and non-products or a combination (such as Red Hat OpenShift Container Platform). But you cannot buy “Red Hat OpenShift” per se.
- **Event-Driven Ansible:** A name for a set of features, tools, and concepts used to create, manage, and scale event-driven automation for a broad range of IT use cases, including cross-network service delivery, edge, cloud, security operations, and infrastructure solutions. But again, you cannot buy “Event-Driven Ansible”.

- **Red Hat AI:** A name for Red Hat's artificial intelligence (AI) platforms and platform add-ons, excluding Red Hat Lightspeed. “Red Hat AI” is used to refer to those products—currently Red Hat Enterprise Linux® AI and Red Hat OpenShift AI—collectively.

feature

Single or multiple process(es) built into a base product that deliver(s) an outcome (information, performance improvement, etc.) to help(s) the product do what it does or be used in the intended way. A feature is never standalone, as it is an inherent part of a product, tool, or service.

Examples:

- **resource optimization:** Assigns a score to each cloud resource (based on cost, efficiency, etc.) to deliver recommendations to developers on the right size of Red Hat OpenShift cluster to run for each stage of the application life cycle and how to make the best use of the resources.
- **remote execution:** The process/capability by which IT administrators can quickly and easily run an arbitrary command on Red Hat Satellite clients.

Note

Components should not be given Red Hat-branded names (unless required for specific marketing exceptions).

general availability (GA)

The release stage for a product or service when it becomes widely available for market/public use as a fully functional, stable, and supported piece of technology or experience. Reached after development, testing, and (optional) limited release—such as Developer Preview, Service Preview, or Technology Preview—then followed by support and maintenance phases.

Related: limited availability

limited availability

Similar to general availability (GA), the limited availability release stage for a product or service is when it becomes available only for certain customers, industries, or use cases—including through Red Hat's private pricebooks—as a fully functional, stable, and supported piece of technology or experience. Reached after development, testing, and (optional) limited release—such as developer preview, Service Preview, or Technology Preview—then followed by support and maintenance phases.

Related: general availability (GA)

market product

A market product is a single SKU that is made up of applications, appliances, and software packages that provides a unified set of features and capabilities. In other words, a single product. Market products are usually standalone, but some non-standalone tools, etc. can also be considered market products because they are sold separately.

Examples:

- Red Hat Enterprise Linux
- Red Hat AMQ
- Red Hat Quay
- Red Hat OpenShift Container Platform

offering

An offering is the combination of a Red Hat market product(s) with a specific business model (how we sell them), identified by a SKU.

Example:

- Red Hat OpenShift Container Platform with Application Foundations, Standard (16 Cores or 32 vCPUs)

Note

"Offering" CANNOT be used as part of a product name. “Offering” is not a generic term; it has a specific sales-related meaning. Refer to the MDM product data dictionary for more information.

operator

A method of packaging, deploying, and managing a Kubernetes-native application (an application that is both deployed on Kubernetes and managed using the Kubernetes APIs and kubectl tools). It is a software version of a person operating Kubernetes/OpenShift by following the same steps or approach.

Operators are either built into (and developed in cadence with) the base OpenShift platform or are developed separately and can be added to OpenShift. Separately developed operators are installed on OpenShift clusters through OperatorHub, a web console interface/catalog accessed by administrators.

Examples:

- **File integrity operator:** Supports OpenShift Container Platform’s core security functions.
- **nbde tang server operator:** Not required to run OpenShift Container Platform, but can provide additional, optional capabilities.

Caution

"Operator" should not be included in the name except when clarity is needed (for example, if the name is describing a generic process/function).

platform

A platform is a market product used as a foundation for running or building applications, processes, services, or technologies.

Examples:

- Red Hat OpenShift Container Platform

- Red Hat JBoss Enterprise Application Platform
- Red Hat Ansible Automation Platform

portal

A user interface (UI) or central destination for accessing information, technology/products, or services.

Examples:

- **Red Hat Hybrid Cloud Console:** The central location where Red Hat customers can manage their Red Hat product and service subscriptions, create and share automations with Red Hat Ansible Automation Platform, view inventories of their Red Hat Enterprise Linux hosts or OpenShift clusters, and more.
- **Red Hat Marketplace, operated by IBM:** A single dashboard where Red Hat customers can browse, try, buy, and manage certified enterprise software for their clouds running Red Hat OpenShift. Beyond providing a searchable catalog, it automates deployment of the software, provides access to support, and centralizes billing.

portfolio

A portfolio is a descriptive category of products or services. Typically, a portfolio includes many different market products, services, and technologies. These groups do not have SKUs and are not sold.

Examples:

- The Red Hat portfolio (everything we sell)
- *redhat.com* portfolio categories: Cloud, Linux platforms, management, etc.

Note

"Portfolio" CANNOT be used as part of a product name.

professional service(s)

A term for human-delivered engagements or tools for consulting, training, certification, technical account management, and support. Professional services are provided to help customers get the most from their Red Hat product subscriptions by learning new best practices, collaborating with experts to plan deployments, get help troubleshooting issues, or achieve certification on Red Hat products. Some professional services are included with Red Hat offerings (such as standard support), while others can be purchased separately and have their own SKUs.

Professional services make up the Red Hat Services portfolio.

Types of services:

- **Support:** Subscription-based technical assistance that includes incident support ("break-fix"), proactive analytics, documentation, and knowledge sharing.
- **Training:** Teaches students how to install, deploy, and use Red Hat technologies. Training can take place in a traditional classroom, on site, or online.
- **Certification:** Confers proof of technical competence using Red Hat technologies.
- **Consulting:** Direct (virtual or on-site) expert interaction to help customers install or deploy Red Hat technologies, or improve their use.

Examples:

- Red Hat Consulting: The family name for Red Hat's consulting professional services
- Red Hat Consulting: App Migration and Modernization Accelerator
- Red Hat Learning Subscription Basic
- Red Hat Technical Account Management Services (SKU name for Red Hat Technical Account Management)

Note

Do not use “professional services” if you are referring to as-a-Service technology, software-based services, or managed services.

See: service

program

A set of related business measures or activities and the people who are employed to support them to achieve a particular outcome or long-term aim.

Examples:

- **Red Hat Academy:** A program that provides discounted access to Red Hat Training and Certification for academic institutions’ students and faculty.
- **Red Hat Innovators in the Open:** Red Hat’s customer reference program. It comprises the reference materials Red Hat produces showcasing our customers’ success with Red Hat technology.

project

A project is what an open source community creates—software that people can download to install on their computers, phones, servers, etc. It’s not a product, because it’s not sold, but a project’s technology can be incorporated into products sold by Red Hat and other companies.

Examples:

- **Quarkus:** A Kubernetes-native Java framework developed by the Quarkus community. A version modified by Red Hat is part of Red Hat Runtimes.
- **Kubernetes:** Originally developed by Google and now an open source project under the Cloud Native Computing Foundation.

service

Technology that supports a process related to monitoring/analysis, operation, or data collection, etc.; can be self-run or delivered by a third party (such as Red Hat); can run itself or require human input to operate. A service is a type of add-on (not standalone) that provides not just technology but a method of delivery/use.

A managed service is an engagement where Red Hat and/or partner(s) operate and manage a technology deployment for a customer, in contrast to self-managed deployments.

Some of our product names/portfolios include “Services” in the name, such as Red Hat Application Services (our middleware portfolio).

Note

The term “professional services” refers to consulting, technical account management, and similar instructional products from Red Hat, rather than a type of technology.

Examples:

- **Red Hat OpenShift AI:** Runs on top of OpenShift Container Platform to process AI/ML data.
- **Red Hat OpenShift Service on AWS Hosted Control Planes:** Provides access to Red Hat functionality that provisions the Hosted Control Plane service.

See: professional service(s)

Service Preview

Similar to Technology Preview, professional or software services or engagements in Service Preview are available for early access to customers, so that they can test functionality and provide feedback during the development process. However, these services are not fully supported, and documentation might be incomplete. Usually followed by general availability (GA) or limited availability.

Related: developer preview, Technology Preview

solution

Formerly referred to a specific selling model of products plus services, emphasizing the extended buyer's journey. Now used descriptively, rather than as a category.

Examples:

- Application modernization and migration solution
- Red Hat solutions for private cloud
- Red Hat edge solutions

Note

"Solution(s)" CANNOT be used as part of a product name. As they are concepts, rather than market products, solution names should not be capitalized.

sub-brand

A sub-brand is a non-descriptive term or phrase used in a product name in addition to "Red Hat". Sub-branded offerings are ONLY allowed during the brand and product transition after an acquisition, with the exception of our two official sub-brands OpenShift and Ansible.

Examples:

- **3scale:** Red Hat 3scale API Management
- **JBoss:** Red Hat JBoss Enterprise Application Platform
- **OpenShift:** Red Hat OpenShift Container Platform

subscription

A subscription is a pricing and software or service delivery model applied to an individual SKU that gives customers access to updates, bug fixes, support, and any other services associated with that offering for a specific period of time. Subscriptions can be renewed or canceled, per the subscription agreement.

Examples:

- Red Hat Enterprise Linux subscription
- A subscription for Red Hat OpenShift Service on AWS
- Red Hat Learning Subscription Basic
- Red Hat Enhanced Solution Support

technical, customer-facing documentation

Refers to non-marketing, non-sales content used to train customers, assist with installation, etc., such as getting started guides, Jira/Bugzilla entries, support case summaries, Red Hat Training courses, and content found on the Customer Portal and Red Hat Documentation. The audience for these materials is often already familiar with or using our products and services.

technology, technologies

Technologies are code, components, and/or features that compose products. Technology/ies can also describe an area, category, or segment of the IT industry.

Examples:

- OpenStack® technologies from Red Hat
- Big data technologies
- KVM hypervisor technology

Note

"Technologies" (or "technology") CANNOT be used as part of a product name.

Technology Preview

Features or products in Technology Preview are available for early access to customers, so that they can test functionality and provide feedback during the development process. However, these features or products are not fully supported, and documentation might be incomplete. Usually followed by general availability (GA) or limited availability.

See also: developer preview, Service Preview

tool

A technology that supports a specific task(s) or activates and controls a particular function(s). Can be comprised of one or more component(s). Can be built into a product to deliver the product's basic functionality or added on to provide new or improved capabilities. Can be delivered in a tool set or toolchain when needed to support a larger or more complex process or outcome.

Plug-ins and frameworks are types of tools.

Examples:

- **Red Hat support for Spring Boot:** Simplifies web application and microservices development using Spring Framework by providing autoconfiguration, opinionated configuration, or standalone application creation capabilities. Supports features such as health checks and failover.
- **Red Hat Enterprise Linux for SAP Applications:** Includes a tool for installing legacy versions of Java-based SAP applications on Linux x86_64 architecture.
- An application programming interface (API) is a piece of software that lets two programs communicate with each other.

Note

Use "tools" instead of "toolset". "Toolkit" is used when the tools are integrated or installed together using a common mechanism.