



Topics > Integration > What is a REST API?

# What is a REST API?

Published May 8, 2020 • 3-minute read

 Copy URL

# What is a REST API?

A REST API is an application programming interface (API) that follows the design principles of the REST architectural style. REST is short for representational state transfer, and is a set of rules and guidelines about how you should build a web API.

## What's an API?

An API is a set of definitions and protocols for building and integrating application software. It's sometimes referred to as a contract between an information provider and an information user—establishing the content required from the consumer (the call) and the content required by the producer (the response). For example, the API design for a weather service could specify that the user supply a zip code and that the producer reply with a 2-part answer, the first being the high temperature, and the second being the low.

In other words, if you want to interact with a computer or system to retrieve information or perform a function, an API helps you communicate what you want to that system so it can understand and fulfill the request.

You can think of an API as a mediator between the users or clients and the resources or web services they want to get. It's also a way for an organization to share resources and information while maintaining security, control, and authentication—determining who gets access to what.

Another advantage of an API is that you don't have to know the specifics of caching—how your resource is retrieved or where it comes from.

[Learn more about APIs](#)

---

**Red Hat resources**

**Keep reading →**

---

## REST

REST is a set of architectural constraints, not a protocol or a standard. API developers can implement REST in a variety of ways.

When a client request is made via a RESTful API, it transfers a representation of the state of the resource to the requester or endpoint. This information, or representation, is delivered in one of several formats via HTTP: JSON (Javascript Object Notation), HTML, XLT, Python, PHP, or plain

text. JSON is the most generally popular file format to use because, despite its name, it's language-agnostic, as well as readable by both humans and machines.

Something else to keep in mind: Headers and parameters are also important in the HTTP methods of a RESTful API HTTP request, as they contain important identifier information as to the request's metadata, authorization, uniform resource identifier (URI), caching, cookies, and more. There are request headers and response headers, each with their own HTTP connection information and status codes.

In order for an API to be considered RESTful, it has to conform to these criteria:

- A client-server architecture made up of clients, servers, and resources, with requests managed through HTTP.
- Stateless client-server communication, meaning no client information is stored between get requests and each request is separate and unconnected.
- Cacheable data that streamlines client-server interactions.
- A uniform interface between components so that information is transferred in a standard form. This requires that:
  - resources requested are identifiable and separate from the representations sent to the client.
  - resources can be manipulated by the client via the representation they receive because the representation contains enough information to do so.
  - self-descriptive messages returned to the client have enough information to describe how the client should process it.
  - hypertext/hypermedia is available, meaning that after accessing a resource the client should be able to use hyperlinks to find all other currently available actions they can take.
- A layered system that organizes each type of server (those responsible for security, load-balancing, etc.) involved the retrieval of requested information into hierarchies, invisible to the client.
- Code-on-demand (optional): the ability to send executable code from the server to the client when requested, extending client functionality.

Though the REST API has these criteria to conform to, it is still considered easier to use than a prescribed protocol like SOAP (Simple Object Access Protocol), which has specific requirements like XML messaging, and built-in security and transaction compliance that make it slower and heavier.

In contrast, REST is a set of guidelines that can be implemented as needed, making REST APIs faster and more lightweight, with increased scalability—perfect for Internet of Things (IoT) and mobile app development.

**[Read more about REST vs SOAP](https://www.redhat.com/en/topics/api/what-is-a-rest-api)**

## Why choose Red Hat?

Red Hat® Integration is an API-centric solution that provides application connectivity and data transformation, service composition and orchestration, real-time message streaming, change data capture, and API management all in one place to connect applications and data across a hybrid infrastructure. It's combined with a cloud-native platform and toolchain to support modern app development.

Red Hat 3scale API Management is a component of Red Hat Integration, and helps you share, distribute, control, and monetize your APIs on an infrastructure platform built for performance, control, security, and growth. Place 3scale components on-premise, in the cloud, or both.

[Learn more](#)

Hub

### The official Red Hat blog

Get the latest information about our ecosystem of customers, partners, and communities.

**Keep reading** →

Recommended for you

## Cloud-native API Administration with Red Hat 3scale API Management

Build and Administer APIs with Red Hat 3scale API Management (DO240) teaches you how to manage, monetize, and secure APIs with Red Hat 3scale API Management.

[View course](#)

## Keep reading

### What is a service mesh?

Read this article to find out more about what a service mesh is, how it works, and the benefits and challenges of implementing a service mesh.

**Read the article** [→](#)

### What is application integration?

Application integration connects different systems and applications by enabling them to work together through exchanging data and using services.

**Read the article** [→](#)

## What is middleware?

Middleware is software that provides common services and capabilities to applications outside of what's offered by the operating system.

**Read the article** [→](#)

# Integration resources

## Related content

### BLOG POST

Evolving our ServiceNow integration: Sunsetting the Notification Service for more capable alternatives

### CASE STUDY

GSE and Blaze transform the electricity market with OpenShift

### BLOG POST

Beyond the horizon: Navigating the bridge between today's tech and tomorrow's AI

### ANALYST MATERIAL

Meeting power grid challenges with new computing technologies

## Related articles

What is a service mesh?

What is application integration?

What is middleware?

What is Apache Kafka?

Why choose Red Hat for middleware?

What is an API?

Red Hat OpenShift application services

What is an independent software vendor (ISV)?

What is a Kafka service?

What is streaming data?

Do banking APIs benefit from cloud technology?

How microservices support IT integration in healthcare

**More about this topic** →



LinkedIn

YouTube

Facebook

X

## Platforms

Red Hat AI

Red Hat Enterprise Linux

Red Hat OpenShift

Red Hat Ansible Automation Platform

See all products

## Tools

Training and certification

My account

Customer support

[Developer resources](#)  
[Find a partner](#)  
[Red Hat Ecosystem Catalog](#)  
[Documentation](#)

## Try, buy, & sell

[Product trial center](#)  
[Red Hat Store](#)  
[Console](#)

## Communicate

[Contact sales](#)  
[Contact customer service](#)  
[Contact training](#)  
[Social](#)

## About Red Hat

Red Hat is an open hybrid cloud technology leader, delivering a consistent, comprehensive foundation for transformative IT and artificial intelligence (AI) applications in the enterprise. As a trusted adviser to the Fortune 500, Red Hat offers cloud, developer, Linux, automation, and application platform technologies, as well as award-winning services.

[Our company](#)

[Newsroom](#)

[How we work](#)

[Open source commitments](#)

[Customer success stories](#)

[Our social impact](#)

[Analyst relations](#)

[Jobs](#)

## Change page language

English

[About Red Hat](#)

[Jobs](#)

[Events](#)

[Locations](#)

[Contact Red Hat](#)

[Red Hat Blog](#)

[Inclusion at Red Hat](#)

[Cool Stuff Store](#)



[Red Hat Summit](#)

[© 2025 Red Hat](#)

[Privacy statement](#)

[Terms of use](#)

[All policies and guidelines](#)

[Digital accessibility](#)

[Cookie-präferenzen](#)