

Topics > Understanding data storage > File storage, block storage, ...

# File storage, block storage, or object storage?

Published February 1, 2018 • 4-minute read

 Copy URL

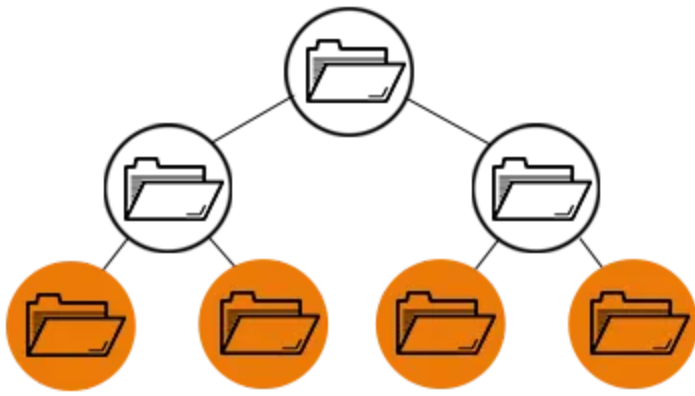
## Overview

Files, blocks, and objects are storage formats that hold, organize, and present data in different ways—each with their own capabilities and limitations. File storage organizes and represents data as a hierarchy of files in folders; block storage chunks data into arbitrarily organized, evenly sized volumes; and object storage manages data and links it to associated metadata.

Containers are highly flexible and bring incredible scale to how apps and storage are delivered.

**Learn more about container storage**

## What is file storage?

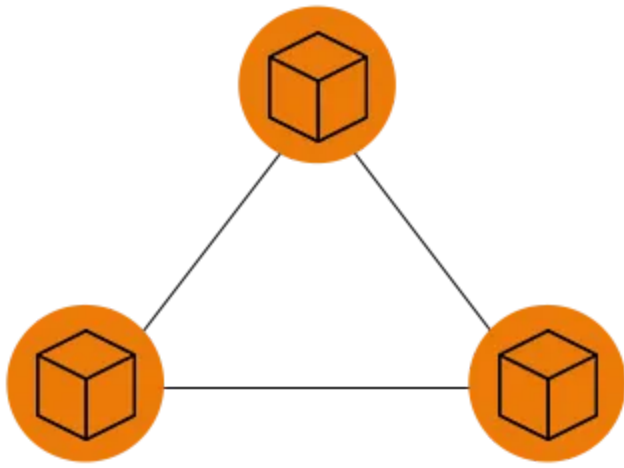


File storage, also called file-level or file-based storage, is exactly what you think it might be: Data is stored as a single piece of information inside a folder, just like you'd organize pieces of paper inside a manila folder. When you need to access that piece of data, your computer needs to know the path to find it. (Beware—It can be a long, winding path.) Data stored in files is organized and retrieved using a limited amount of metadata that tells the computer exactly where the file itself is kept. It's like a library card catalog for data files.

*Think of a closet full of file cabinets.* Every document is arranged in some type of logical hierarchy—by cabinet, by drawer, by folder, then by piece of paper. This is where the term hierarchical storage comes from, and this is file storage. It is the oldest and most widely used data storage system for direct and network-attached storage systems, and it's one that you've probably been using for decades. Any time you access documents saved in files on your personal computer, you use file storage. File storage has broad capabilities and can store just about anything. It's great for storing an array of complex files and is fairly fast for users to navigate.

The problem is, just like with your filing cabinet, that virtual drawer can only open so far. File-based storage systems must scale out by adding more systems, rather than scale up by adding more capacity.

## What is block storage?



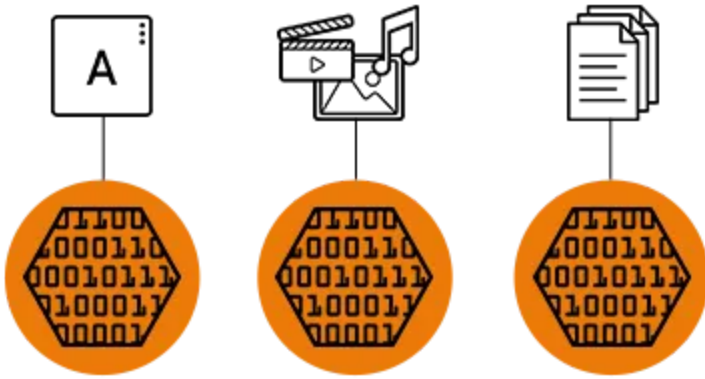
Block storage chops data into blocks—get it?—and stores them as separate pieces. Each block of data is given a unique identifier, which allows a storage system to place the smaller pieces of data wherever is most convenient. That means that some data can be stored in a Linux® environment and some can be stored in a Windows unit.

Block storage is often configured to decouple the data from the user's environment and spread it across multiple environments that can better serve the data. And then, when data is requested, the underlying storage software reassembles the blocks of data from these environments and presents them back to the user. It is usually deployed in storage-area network (SAN) environments and must be tied to a functioning server.

Because block storage doesn't rely on a single path to data—like file storage does—it can be retrieved quickly. Each block lives on its own and can be partitioned so it can be accessed in a different operating system, which gives the user complete freedom to configure their data. It's an efficient and reliable way to store data and is easy to use and manage. It works well with enterprises performing big transactions and those that deploy huge databases, meaning the more data you need to store, the better off you'll be with block storage.

There are some downsides, though. Block storage can be expensive. It has limited capability to handle metadata, which means it needs to be dealt with in the application or database level—adding another thing for a developer or systems administrator to worry about.

## What is object storage?



Object storage, also known as object-based storage, is a flat structure in which files are broken into pieces and spread out among hardware. In object storage, the data is broken into discrete units called objects and is kept in a single repository, instead of being kept as files in folders or as blocks on servers.

Object storage volumes work as modular units: each is a self-contained repository that owns the data, a unique identifier that allows the object to be found over a distributed system, and the metadata that describes the data. That metadata is important and includes details like age, privacies/securities, and access contingencies. Object storage metadata can also be extremely detailed, and is capable of storing information on where a video was shot, what camera was used, and what actors are featured in each frame. To retrieve the data, the storage operating system uses the metadata and identifiers, which distributes the load better and lets administrators apply policies that perform more robust searches.

Object storage requires a simple HTTP application programming interface (API), which is used by most clients in all languages. Object storage is cost efficient: you only pay for what you use. It can scale easily, making it a great choice for public cloud storage. It's a storage system well suited for static data, and its agility and flat nature means it can scale to extremely large quantities of data. The objects have enough information for an application to find the data quickly and are good at storing unstructured data.

There are drawbacks, to be sure. Objects can't be modified—you have to write the object completely at once. Object storage also doesn't work well with traditional databases, because writing objects is a slow process and writing an app to use an object storage API isn't as simple as using file storage.



# Why Red Hat?

Red Hat offers choices for software-defined storage depending on your needs.

Red Hat Ceph Storage delivers SDS on your choice of industry-standard hardware. With block, object, and file storage combined into 1 platform, it efficiently and automatically manages all your data.

Additionally, Red Hat OpenShift Data Foundation is SDS for containers. Engineered as the data and storage services platform for Red Hat OpenShift, Red Hat OpenShift Data Foundation helps teams develop and deploy applications quickly and efficiently across clouds.

**Continue reading** →

## Keep reading

### TOPIC

## Understanding data services

Data services are collections of small, independent, and loosely coupled functions that enhance, organize, share, or calculate information collected and saved in data storage volumes.

**Keep reading** →

## ARTICLE

### Why choose Red Hat storage

Learn what software-defined storage is and how to deploy a Red Hat software-defined storage solution that gives you the flexibility to manage, store, and share data as you see fit.

**Read more** →

## ARTICLE

### What is cloud storage?

Cloud storage is the organization of data kept somewhere that can be accessed by anyone with the right permissions over the internet. Learn about how it works.

**Read more** →

## More about storage

Products

Related articles

Resources



**Red Hat**  
OpenShift  
Data Foundation



**Red Hat**  
Ceph Storage

An open, massively scalable, software-defined storage system that efficiently

Software-defined storage that gives data a permanent place to live as containers spin up and down and across environments.

**Learn more** →

manages petabytes of data.

**Learn more** →

\* The OPENSTACK logo and word mark are trademarks or registered trademarks of OpenInfra Foundation, used under license.



in



---

Products

---

Tools

---

Try, buy, & sell

---

Communicate

---

## About Red Hat

We're the world's leading provider of enterprise open source solutions—including Linux, cloud, container, and Kubernetes. We deliver hardened solutions that make it easier for enterprises to work across platforms and environments, from the core datacenter to the network edge.

## Select a language

 English ▼



About Red Hat

[Jobs](#)

[Events](#)

[Locations](#)

[Contact Red Hat](#)

[Red Hat Blog](#)

[Diversity, equity, and inclusion](#)

[Cool Stuff Store](#)

[Red Hat Summit](#)

---

**© 2024 Red Hat, Inc.**

[Privacy statement](#)

[Terms of use](#)

[All policies and guidelines](#)

[Digital accessibility](#)

[Cookie preferences](#)