This version of GitHub Enterprise was discontinued on 2023-03-15. No patch releases will be made, even for critical security issues. For better performance, improved security, and new features, upgrade to the latest version of GitHub Enterprise. For help with the upgrade, contact GitHub Enterprise support.

About cluster nodes

In this article

In a GitHub Enterprise Server cluster, nodes are individual virtual machines (VMs) running the GitHub Enterprise Server software that comprise the instance. Each node runs a set of services.

GitHub determines eligibility for clustering, and must enable the configuration for your instance's license. Clustering requires careful planning and additional administrative overhead. For more information, see "About clustering."

About GitHub Enterprise Server cluster nodes

Each node in a GitHub Enterprise Server cluster is a virtual machine (VM) that runs the GitHub Enterprise Server software. Before you deploy a cluster, you can review hardware requirements, required services, and design recommendations.

Note: GitHub Enterprise Server clustering must be configured with HTTPS.

Hardware requirements

Each node must have a root volume, as well as a separate data volume. These are minimum recommendations. More resources may be required depending on your usage, such as user activity and selected integrations.

Services	Minimum memory required	Minimum data volume free space Required
`job-server`, `memcache-server`, `web-server`	14 GB	1 GB
`consul-server`, `mysql-server`, `redis-server`	14 GB	10 GB
`git-server`,	14 GB	10 GB

`metrics-server`, `pages-server`, `storage-server`		
`elasticsearch-server`	14 GB	10 GB

Services required for clustering

GitHub Enterprise Server comprises a set of services. In a cluster, these services run across multiple nodes, and the instance balances requests between the nodes. The instance automatically stores redundant copies of data on separate nodes. Most services are equal peers with other instances of the same service. The exceptions to this distribution are the `mysql-server` and `redis-server` services, which operate with a single primary node with one or more replica nodes.

For adequate redundancy, use these minimum nodes operating each service.

> **Note:** Your environment's scaling requirements depend on many factors, including the size and number of repositories, number of users, and overall utilization.

Services	Minimum nodes required
`job-server`, `memcache-server`, `metrics-server`, `web-server`	2
`mysql-server`, `redis-server`	2
`consul-server`	3
`git-server`, `pages-server`, `storage-server`	3
`elasticsearch-server`	3

Cluster design recommendations

Clustering allows services that make up GitHub Enterprise Server to be scaled out independently of each other. This flexibility can be used to design and implement a cluster that fits organizations with different scalability requirements. For example, some organizations may need more storage throughput for large or frequent fetches, but web server usage may be relatively low. Another organization may have good performance with fewer storage resources, but need many nodes running `pages-server` or `elasticsearch-server`. Many different combinations are possible. Work with your account representative to determine the best cluster configuration for your specific needs.

- Spread redundant nodes across independent hardware. If you share CPU, memory, or storage devices, you'll reduce performance and introduce single points of failure. Shared networking components can also reduce throughput and increase risk of loss of connectivity in the event of an outage.
- Use fast storage. Storage area networks (SAN) are often optimized for maximum space utilization, availability and fault tolerance, not absolute throughput. GitHub Enterprise Server clustering provides redundancy and availability, and will perform

best on the fastest storage available. Local SSD storage is recommended.

- Establish tiers of nodes that make sense for your organization. An example configuration:
  - Front-end tier with two nodes and the following services:
    - `web-server`
    - `job-server`
    - `memcache-server`
  - Database tier with three nodes and the following services:
    - `consul-server`
    - `mysql-server`
    - `redis-server`
  - Search tier with three nodes and the following service:
    - `elasticsearch-server`
  - Storage tier with three nodes and the following services:
    - `git-server`
    - `pages-server`
    - `storage-server`
    - `metrics-server`