

Artifactory High Availability

By

Rajesh Kumar

- From version 3.1, Artifactory supports a High Availability network configuration with a cluster of 2 or more, active/active, read/write Artifactory servers on the same Local Area Network (LAN).

Benefits

Maximize Uptime

Artifactory HA redundant network architecture means that there is no single-point-of-failure, and your system can continue to operate as long as at least one of the Artifactory nodes is operational. This maximizes your uptime and can take it to levels of up to "five nines" availability.

Manage Heavy Loads

By using a redundant array of Artifactory server nodes in the network, your system can accommodate larger load bursts with no compromise to performance. With horizontal server scalability, you can easily increase your capacity to meet any load requirements as your organization grows.

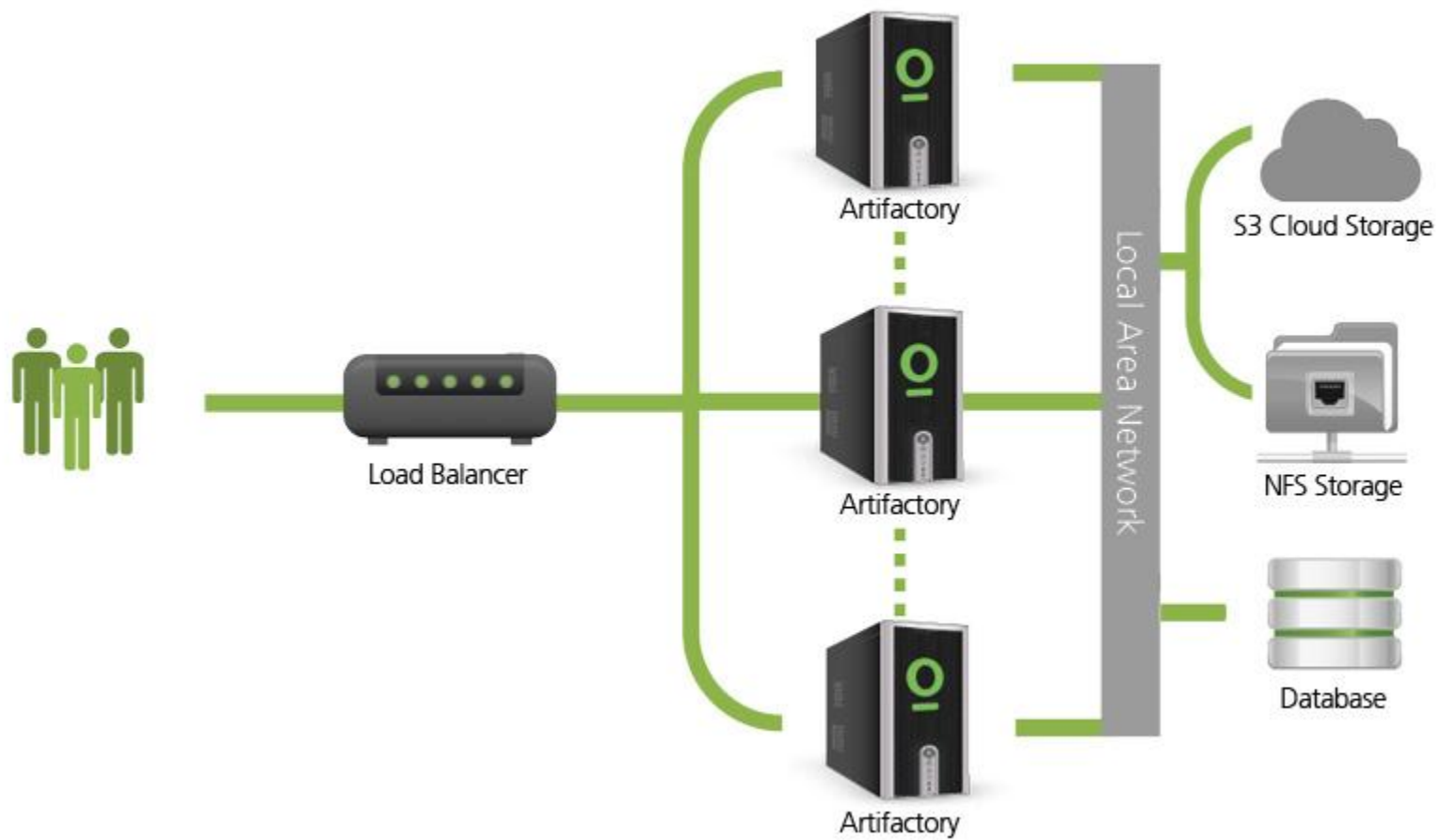
Minimize Maintenance Downtime

By using an architecture with multiple Artifactory servers, Artifactory HA lets you perform most maintenance tasks with no system downtime.

Artifactory HA architecture

- Artifactory HA architecture presents a Load Balancer connected to a cluster of two or more Artifactory servers that share a common database where all the Artifactory configuration files are maintained.
- Binaries may be stored on a Network File System, or using a zoned sharded binary provider as described in [Configuring Sharding for High Availability](#). The Artifactory cluster nodes must be connected through a fast internal LAN in order to support high system performance as well as to stay synchronized and notify each other of actions performed in the system instantaneously. One of the Artifactory cluster nodes is configured to be a "primary" node. Its roles are to execute cluster-wide tasks such as cleaning up unreferenced binaries.

Architecture



Network Topology

Load Balancer

The load balancer is the entry point to your Artifactory HA installation and optimally distributes requests to the Artifactory server nodes in your system. It is the responsibility of your organization to manage and configure it correctly.

Apache load balancer configuration example...

First install the following modules:

```
LoadModule proxy_module modules/mod_proxy.so
```

```
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
```

```
LoadModule proxy_http_module modules/mod_proxy_http.so
```

Then configure as follows:

```
<VirtualHost *:80>
```

```
  ServerAdmin admin@frogs.com
```

```
  ServerName artifactory.jfrog.com
```

```
  ServerAlias *.jfrog.com
```

```
<Proxy balancer://tomcats>
```

```
  # Artifactory server #1
```

```
  BalancerMember http://IP_SERVER_1:PORT route=art1
```

```
  # Artifactory server #2
```

```
  BalancerMember http://IP_SERVER_1:PORT route=art2
```

```
  ProxySet lbmethod=byrequests
```

```
</Proxy>
```

```
  ProxyPreserveHost on
```

```
  ProxyPass /balancer-manager !
```

```
  ProxyPass / balancer://tomcats/
```

```
  ProxyPassReverse /artifactory https://<server name>/artifactory
```

```
  RewriteEngine On
```

```
  RewriteRule ^/$ /artifactory [R,L]
```

```
  LogLevel warn
```

```
  ErrorLog /var/log/httpd/apache-ha-test.error.log
```

```
  CustomLog /var/log/httpd/apache-ha-test.access.log combined
```

```
</VirtualHost>
```

nginx load balancer configuration example...

```
http {  
  ...  
  ...  
  ...  
  upstream artifactory {  
    server IP_SERVER_1:8081;  
    server IP_SERVER_2:8081;  
  }  
  
  server {  
    listen 80;  
    server_name YOUR_SERVER_NAME;  
    ...  
    ...  
    ...  
    rewrite ^/$ http://$host/artifactory/webapp;  
    location / {  
      proxy_pass http://artifactory;  
    }  
  }  
}
```

Artifactory Server Cluster

Each Artifactory server in the cluster receives requests routed to it by the load balancer. All servers share a common database, and communicate with each other to ensure that they are synchronized on all transactions.

Local Area Network

To ensure good performance and synchronization of the system, all the components of your Artifactory HA installation must be installed on the same high-speed LAN.

In theory, Artifactory HA could work over a Wide Area Network (WAN), however in practice, network latency makes it impractical to achieve the performance required for high availability systems.

Filestore

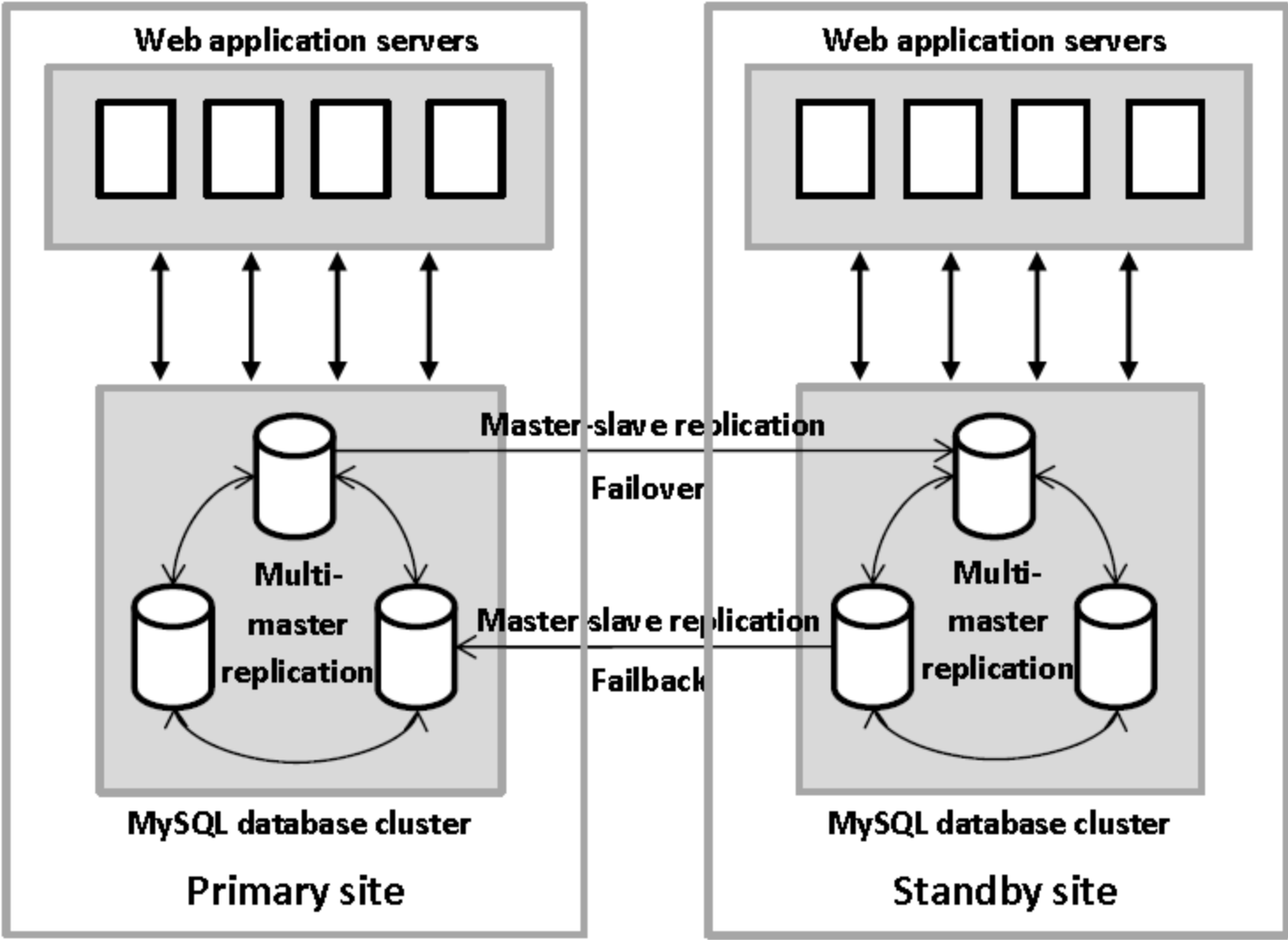
Artifactory HA offers different options for storing binaries. Some examples are:
Local file system in which binaries are stored with redundancy using a binary provider which manages synchronizing files between the cluster nodes according to the redundancy defined.

Cloud storage (currently, Amazon S3 and Google Cloud Storage are supported)

Network File System (NFS)

Database

Artifactory HA requires an external database, which is fundamental to management of binaries and is also used to store cluster wide configuration files. Currently MySQL, Oracle, MS SQL and PostgreSQL are supported. For details on how to configure any of these databases please refer to Configuring the Database.



HA Installation and Setup

- <https://www.jfrog.com/confluence/display/RTF/HA+Installation+and+Setup>