About scaling in Wowza Streaming Engine

When streaming live and video-on-demand (VOD) content from Wowza Streaming Engine™ media server software, the server can handle only so many concurrent streams before it starts to slow down or get overloaded. The number of concurrent streams that Wowza Streaming Engine can comfortably manage varies depending on server hardware, network configuration, stream type, stream bitrate, and connection types. The only sure way to determine the limits for a particular configuration is to perform load tests that show when performance is likely to degrade. When the limit is likely to be reached, you can scale your Wowza Streaming Engine configuration to handle increased capacity. You can also incorporate a load-balancing system to offload player connections to less-loaded servers.

About scaling

Scaling involves using two or more Wowza Streaming Engine servers to make your streams available to larger audiences. Multiple servers arranged this way are called a “server pool” or “server cluster.” Depending on hardware and infrastructure, servers can be added to or removed from the cluster manually based on estimated peak loads, or automatically based on real-time metrics. You can use different methods to scale depending on stream type and target audience, and you can use a combination of methods to suit your needs. Some form of load balancing is also generally required to route player connections to the correct servers in order to balance the cluster.

There’s no one answer for when to scale your Wowza Streaming Engine deployment. When designing a streaming workflow, do try to keep scaling in mind from the start. To help determine when and how best to scale, consider the following:

- The number of video streams that would be available concurrently and their bitrates.
- The estimated audience size and geographical location. Can you control the number of viewers and their location? What would happen if one or more of your videos went viral?
- The short-term and long-term project budget. For your deployment, is it more cost-effective to keep systems online 24/7 to handle peak loads or to use a cloud service that allows you to bring systems online as demand increases and take them offline when demand is low?
Scaling options

The following options can be used for scaling Wowza Streaming Engine configurations. Sometimes you may want to use multiple options simultaneously to handle different demands.

Wowza live stream repeater

**Applies to:** Live streaming

Live stream repeating is a system in which multiple Wowza Streaming Engine servers are configured as either origin or edge servers. Live encoders that produce the content connect to the origin servers, while players connect to the edge servers. The edge servers connect to the origin servers in order to repeat the live stream. A live stream repeater configuration can be used for all types of streaming. To scale up, you can just add more edge servers to the cluster and point them to the streams originating from the origin servers. A separate load balancer is typically used to direct the players to the correct edge server in the cluster.

For more information, see [Scale with Wowza Streaming Engine live stream repeater](#).

Wowza Media Cache

**Applies to:** VOD streaming

The Media Cache feature in Wowza Streaming Engine can be used with multiple servers to provide access to VOD content from a central storage location. Each server that uses Media Cache can access the same content without having to store the content locally. As with live streaming, scaling up is just a matter of adding new servers to the cluster and configuring them to access the central media storage. A load balancer can be used in the same way that it would be for live content.

For more information, see [Scale with Wowza Streaming Engine Media Cache](#).

Wowza HTTP Origin mode

**Applies to:** Live and VOD streaming

Normally, when you enable HTTP-based playback (Adobe HDS, Apple HLS, Microsoft Smooth Streaming, or MPEG-DASH), Wowza Streaming Engine associates each HTTP connection with a unique internal session so that each one can be tracked and different options can be used for the session. After the initial stream request, subsequent requests have the unique session
Wowza HTTP Origin mode is a special, sessionless mode that uses HTTP caching proxies in front of Wowza Streaming Engine to cache the live or VOD content, thus reducing the load on Wowza Streaming Engine. As the name indicates, HTTP Origin Mode is only available for HTTP player connections.

For more information, see Scale with Wowza Streaming Engine HTTP Origin applications.

**UDP multicast**

**Applies to:** Live and VOD streaming

If your viewers are on a controlled private network, you can use UDP multicast to deliver live streams to a wider audience. Multicast differs from regular scaling in that routers in the network provide the capacity instead of a Wowza Streaming Engine cluster. Multicast generally only works in private networks where the routers can be configured to allow multicast traffic to traverse the network. A single stream is sent from Wowza Streaming Engine to the multicast address, and the players all connect to the multicast address to receive the stream. Wowza Streaming Engine supports both RTP and MPEG-TS streaming over UDP multicast; however, RTP is recommended due to MPEG-TS streaming licensing requirements.

**About load balancing**

Load balancing is used to distribute the player connections between servers in a cluster. Simple load balancing can be performed using certain types of DNS services. Software or hardware solutions specifically designed for load balancing are also available. A properly designed load balancing solution should allow easy addition and removal of servers from the cluster and also allow for failure detection. Each type of load balancing has its own strengths and weaknesses and must be chosen to match the server cluster.

For more information, see About load balancing in Wowza Streaming Engine.

**Common scaling and load balancing scenarios**

Occasional live events with a mixture of players
In this scenario, a single server is deployed for day-to-day streaming. Occasionally a larger event that requires more than one server will be streamed. The players will be a mixture of RTMP, RTSP, and HTTP players.

The best way to handle this type of scenario is to configure and run Wowza Load Balancer configured on the original server. When there’s only one server in the cluster (the original server), the load balancer will redirect all connections to the streaming application on itself.

When required, extra servers can be configured as live stream repeater edges, pointed to the original server application, and added to the cluster. The load balancer automatically detects these and starts redirecting connections to them. After the event is finished, the extra servers can be taken offline until they’re required again.

If the event is short, it’s best to use a cloud instance of Wowza Streaming Engine for the extra servers, as they can be brought online and taken down quickly and you only pay for the time they run and they don’t require hardware purchase for the additional servers.

Occasional live events with mainly HTTP players

This scenario is similar to previous one, except most of the players are HTTP-based. The best way to handle this scenario is to configure the Wowza Streaming Engine application as a HTTP Origin application and use Amazon CloudFront to handle all HTTP connections for the event. If the event is regional, then CloudFront should be configured to use the closest region. All RTMP and RTSP connections will go directly to Wowza Streaming Engine while all HTTP connections will be routed to CloudFront for playback.

Large VOD library with constant new releases

In this scenario, there is a large video library with new releases that are very popular. There are a couple of ways to handle this depending on the types of player being used.

As a minimum, use Media Cache to connect at least one Wowza Streaming Engine server to the video library using HTTP. This allows robust HTTP-based storage (such as multiple web servers) or a cloud solution (like Amazon S3 or Rackspace Cloud Files) to be used.

Multiple Wowza Streaming Engine servers can be connected to the video library as required using a similar solution to the first example above—or if the players are mainly HTTP-based, then CloudFront can be used to connect to Wowza Streaming Engine. When CloudFront is used with VOD streams, the regions should be limited on the CloudFront distribution to closely match the intended audience. This ensures that the content is cached effectively on the CloudFront distribution.
Small to medium server cluster

In this scenario, a cluster of Wowza Streaming Engine servers is configured to handle regular loads of up to 50,000 concurrent connections. There is control over the players and webpages that launch them so compatibility with Wowza Load Balancer can be assured.

One or two servers in the cluster are configured as live stream repeater origins and the rest are configured as live stream repeater edges. The origins are also configured as Wowza Load Balancers.

Round Robin DNS can be used to resolve the origin and load balancer domain names.

Encoders connect to the origins and the players connect to the edges via the load balancer.

Use Media Cache on the edge servers to retrieve VOD content from a central video library.

HTTP connections can be served using a cluster of HTTP caching proxies or CloudFront instead of many Wowza Streaming Engine servers. Regular proxies require a different load balancer, and CloudFront could be too expensive over a long period of heavy use.

New clusters can started in a different region where geographical DNS can direct connections to the closest region.

**Note:** Custom code may be required to allow the edges to locate the origin streams.

Large server cluster

Similar to above except that the cluster is sized to handle more than 50,000 connections. There may not be control over the players or webpages that launch them, so they may not be compatible with Wowza Load Balancer.

The main difference between this and the previous solution is how the load balancing is done. Instead of Wowza Load Balancer, one or two third-party load balancers are used to handle a high number of connections. There may be a number of player that aren’t compatible with Wowza Load Balancer but will work with the third-party load balancer.

Again, HTTP caching proxies or CloudFront could be used for the HTTP connections.