

Name - Manuj Dhariwal

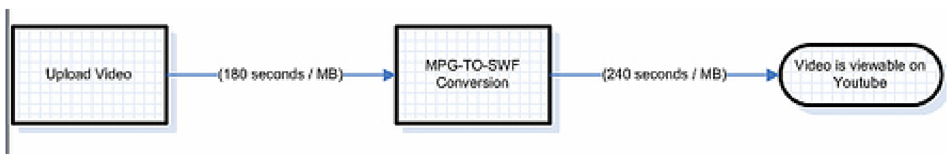
RollNo.- 04020512

For performance analysis of video on demand services I choose the case of youtube.

YouTube has many intriguing queuing properties. This article will primarily look at the mpeg-to-swf conversion and study out the queuing properties of that process.

Assumptions and Data

Below is a very simple, high-level process map of the steps to upload a video on YouTube:



In summary, the user does the following:

1. User uploads video
2. User waits while video is uploaded
3. Upload is completed
4. User waits for video to be converted from MPEG to SWF
5. Conversion from MPEG to SWF completes

Based on a small sample of 20 uploaded videos of an average file size of 3.5 MB, I calculate the mean for uploads to be ~180 seconds per MB and ~240 seconds per MB for the conversion.

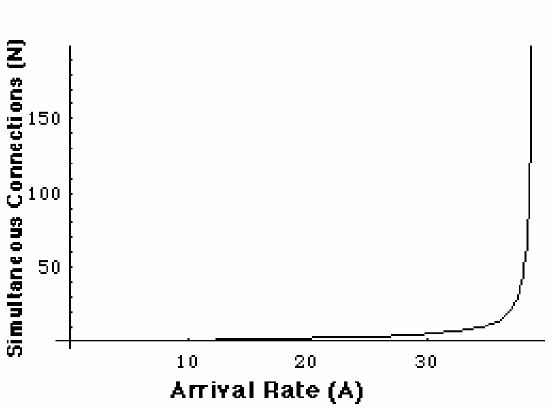
Based on YouTube's own disclosure, we also know that there are on average 65,000 video uploads per day. We do not know the average file size.

Queuing Properties

Important items to note when studying the queuing properties of a system are the following:

1. λ = Arrival Rate, or more specific, the time between arrivals. For most queues, we can assume that the arrival distribution can be approximated by a Poisson distribution; which means that the time between arrivals are not deterministic, but random.
2. μ = Service Rate, or more specific the time for a arrival to be serviced.

A poisson distribution typically looks skewed to the left or to the right — that is because the mean and the standard deviation is the same. Here's a standard picture of a poisson for server utilization:



What we see above is that as there are more simultaneous connections, there is a subsequent arrival rate batching — represented by the poisson curve above.

Presupposition

So, 65,000 video uploads on a 12 hour day, gives us the following:

λ = 65,000 / 720 minutes = **(90 / minute)**
μ = 3.5 * 240 seconds = 840 seconds; 840 seconds / 60 = **(14 conversion / minute)**

Arguably, 14 conversion / minute is very low. Let's just assume that YouTube average service rate is **200 conversions / minute**

Average Number of Videos Waiting to be Converted

The equation to learn about the average number of files in the conversion process is the following:

$C_w = (\lambda^2 / \mu(\mu - \lambda))$

So,

$C_w = [(8100) / (200(200 - 90)] =$ **.36**

So, given the assumptions above, not even 1 video is waiting to be converted from MPEG-to-SWF.

Average Number of Videos in the Conversion Process

$C_s = (\mu - \lambda)$

So,

$C_s = [(90) / (200 - 90)] =$ **.81**

This means, given the assumptions above, that at any point in time, there is 1 video in the conversion system.

Average Time Spent Waiting

$T_w = (\lambda / \mu(\mu - \lambda))$

So, we get:

$T_w = [(90) / (200(200 - 90))] =$ **.004**

This means as videos enter the conversion process, there is hardly any waiting — they are served almost immediately.

Average Time Spent in the System

$T_s = (1 / (\mu - \lambda))$

So, we get,

$T_s = (1 / (200 - 90)) =$ **.009**

This means, then, that as videos are uploaded and enter the conversion queue, they are served almost immediately, without any waiting.

Weaknesses in Analysis

Well, for each input into the YouTube system, one could argue that it has very little impact on resources — this is a common property in telephony and in server modeling. We see the same thing going on with YouTube. The biggest challenge for YouTube is not computing resources, but storage capacity.

Name - Manuj Dhariwal

RollNo.- 04020512

