Presented at ACM Multimedia 1997, Seattle, USA

Long-term Movie Popularity Models in Video-on-Demand Systems

or The Life of an on-Demand Movie

Carsten Griwodz[2], Michael Bär[1], Lars C. Wolf[2]

[1]s d & m
software design & management
Herrnstrasse 57, D-63065 Offenbach, Germany

[2]Industrial Process and System Communications
Darmstadt University of Technology
Merckstr. 25, D-64283 Darmstadt, Germany

Email: baer@sdm.de, {Carsten.Griwodz, Lars.Wolf}@kom.th-darmstadt.de

Correspondence to: Carsten Griwodz

Abstract

Large scale video-on-demand systems require that the servers offering the video retrieval and playback services are arranged as a distributed system in order to support a large number of concurrent streams. If such a system is hierarchical, an end-node server handles the requests from a particular area, the next server in the hierarchy takes the request over for several end-node servers if those can not answer the request and so on. This architecture provides for cost efficiency, reliability and scalability of servers. The end-node servers store only a limited set of the overall available information which changes over time due to user interests. If a video is requested which is not available, this server contacts the next server in the hierarchy.

To decide the size and location of the video servers and the location of videos in the hierarchy, the access behaviour of users must be considered. Various models for the simulation of user behavior (and thus, of the load induced on the video servers) have been presented in the literature. Only a few of these models are designed to take long-term effects into account because the basis for most of the models are short-term influences on a single video server and the load on this single machine.

In this paper we describe a new user behavior model and show that various assumptions made within other models are unrealistic.

1 Introduction

1.1 Video-on-Demand

Several definitions for the term ‘video-on-demand’ can be found in the literature [Sam95, Litt94a, Ov94]. It is sometimes considered as consisting of subcategories (defined, e.g., by [Litt94a]) such as ‘movies on demand’, ‘interactive video games’, ‘interactive news television’, ‘catalog browsing (interactive shopping)’, distance learning (edutainment)’, and ‘interactive advertising’.

The terms interactive television [Hac96, Ov94, Hod95] and services on demand [Hod95, Hes94a, Hes94b] are often used as synonyms for video-on-demand. The term interactive television is used to express that the new service is an extension to the current practice of broadcast television. A minor interactivity exists already today in the form of pay-per-view channels and video recorders. In this paper, we use the term video-on-demand as ‘movies on demand’, as it is the case in various publications [Hod95, Hac96].

Further, ‘Video-on-Demand’ is used with respect to the delivery of ‘movies on demand’ with varying meanings. These variations can be distinguished by the degree of interactivity they provide to the user [Litt94a, Sam95]. Broadcast, or No Video-on-Demand, is the term used for television as we know it today. It provides no means for user interactivity, which makes the user a passive observer. Pay-Per-View is the service that allows the user to pay only for the movies that he really consumes. An influence on the program of the sending station is not possible with this approach either. In Quasi Video-on-Demand, the consumer can subscribe to channels that suit his personal preferences. The interaction means are restricted to a change of the subscribed groups. In Near Video-on-Demand, the same movie is transmitted over multiple channels with a temporal offset. By changing from one channel to another, the user can jump backward and forward in the movie, and he can chose the starting time of a movie at a granularity of some minutes. True Video-on-Demand allows the consumer to prepare a program completely according to his own preferences. The functionality provided to the user is that of a VCR control, with the possible exception of certain features such as fast forward and fast backward. Within this paper we consider always true video-on-demand because this is the next achievable goal in the interactive television business that provides a major change to user interaction.

We assume that true video-on-demand is not only possible in theory but in practice if the networking infrastructure is sufficiently advanced. On this basis we consider ways to deliver individual video streams directly from a content provider through a broadband network to a customer. We do not distinguish whether the network is shared with other traffic (as might be the case with the Internet as a backbone of this system) or whether it is a network that is dedicated to the video-on-demand system (as might be the case if current cable television infrastructure is recycled to this ends), because we assume that in both cases the required resource capacities in the network are reserved for the transfer.

1.2 Video Server Hierarchies

Under the base assumption that large-area (e.g. nation-wide) video-on-demand systems will come into existence, at least in the long term, it is important to investigate how these can be built in a cost-effective manner while ensuring service quality necessary to retain user acceptance.

The radical approaches towards the establishment of video distribution systems are the server cost-optimizing approach and the network cost-optimizing approach. The lowest server cost is achieved by installing a central server which stores all available movies and requires redundant copies of a movie only because of its limited disk bandwidth. In [Tetz94], it is mentioned as a side-note that this approach may finally prove applicable when sufficient user requests are generated that the use of memory instead of disks is feasible. The lowest network cost is achieved by storing all videos at the users' sites. This is obviously never feasible due to storage capacity constraints.

Tree-shaped structures as in Figure 1 connect various users to a central server through intermediate nodes. If these intermediate nodes have the capability for storing or caching data, we call the tree a server hierarchy. Hierarchies that involve caching are an approach that can be easily adopted in distribution networks as they are currently available for cable TV networks and for other distribution networks of telecom operators that may form the basis of video-on-demand systems. Nussbaumer et al. present in [Nuss95] a cost calculation for distributing video data along caching nodes in a distribution tree that connects the central site and the user sites. Their approach takes both storage and networking cost in account and calculates the level of the hierarchy at which a movie, given its popularity, should be stored in order to minimize the sum of these costs.

Graphs other than trees may be more appropriate as a solution to this problem. E.g., RedHi [Shah97] allows multiple parent nodes in a tree-like structure, which allows load balancing and a stability increase, and [Sheu97] considers an approach that builds queues of clients which serve other clients in order to reduce server load. However, the hierarchical approach has the easiest mapping to existing CATV or Telecom infrastructures.

In order to evaluate the effectiveness of caching and distribution strategies in such hierarchies, we need to investigate ways to reduce storage and networking costs, taking the behavior of the users into account. This implies that among the topics that have to be considered are storage capacities, network and storage bandwidth, which are relevant to costs, and delays or the probability for denial of service, which are relevant to acceptance, have to be considered.

In a simulation model, hit rates of specific movies are relevant. For short-term considerations, the Zipf-distribution [Nuss95] is the accepted model for hit probabilities. For the calculation of optimal results, a model based on this assumes that each movie is magically moved onto the server that is at the optimal position for the current hit probability of the movie, e.g., this is assumed to be done during off-peak hours.

We want to find viable approaches to video distribution and caching in a hierarchical system, which comprises the travelling of movies and not only the optimal placement. Thus, we can not simply subscribe to the generic possibilities for movie movement. Instead, we want to model long-term popularity of movies, which might allow for more efficient travelling of movies. We call these changes of movie popularity in time the life-cycle of the movie.

This analysis requires that we model user interaction on the level of individual users requesting single movies in a realistic order that is hardly smoothed by considering an abstract numbers base of users or by looking at a numbers base that is too large to provide the necessary details in behavior. Making an analysis and checking the behavior of a system model first on this level of detail may allow the replacement of the individual user and movie model to a more general one in more complex systems. As a starting point, however, it is necessary to have an accurate model to find inaccuracies in the considerations on the system model.

In order to understand this, we look at various sources of information including but not limited to earlier literature. In extension of the earlier results, we give evidence that the day-to-day changes in rental probability of specific movies results in major exchange of movie titles in a cache-based algorithm which assumes that low-load hours can be used to update videos according to the findings of the previous high-load period. These changes are also presented in section 4.1. Thus, we propose a simulation model which takes into account the hit rate changes of movies located in systems with few customers, e.g. caching head-end servers.

1.3 Outline of the Paper

In the next section we discuss related work on user behavior models. In Section 3 we consider data sources that can be used for verification of video models, and introduce the data used for our study. In Section 4 we verify that the information derived from our various data sources is both consistent and supports earlier results. We present also the reasons why a long-term video development must be considered rather than a single-day model and present our doubts that these earlier results can provide the basis for a simulation model of long-term video development. In Section 5 we present a new model for the simulation of movie life cycles including the randomness by adding simple user modeling. Finally, we give a summary and an outlook on future work.

2 Related Work

Various approaches towards modeling the load of video servers have been proposed in the literature. Usually, there are no simulation models which consider how much interaction might take place from the point of the user. Rather, an analysis is performed to derive the worst case situation that a server (or network) can cope with. One approach is the modeling of single video streams as they are accessed and played. This is generally done in order to understand how the operation of a single machine or cluster of machines can be optimized. Little and Venkatesh take this approach in [Litt94b] with the goal of optimizing disk I/O operations in a single system. Their approach is to build an analytical model for access probabilities based on the work by Ramarao and Ramamoorthy [Ram91]. In [Tew95], Tewari et al. optimize the I/O utilization in a server cluster and use Poisson processes to model the user accesses to the server, with the mean value chosen according to Little's Law [Litt61]. Golubchik et al. investigate in [Gol96] means for sharing video streams in a video-on-demand systems that holds when VCR controls are permitted to the user. Their user model is analytic and assumes a Poisson arrival process. Their analysis does not need long-term movie development.

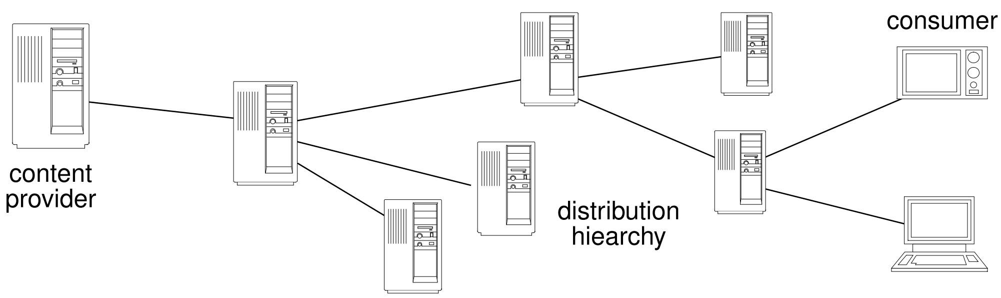

Figure 1: Hierarchical video-on-demand system

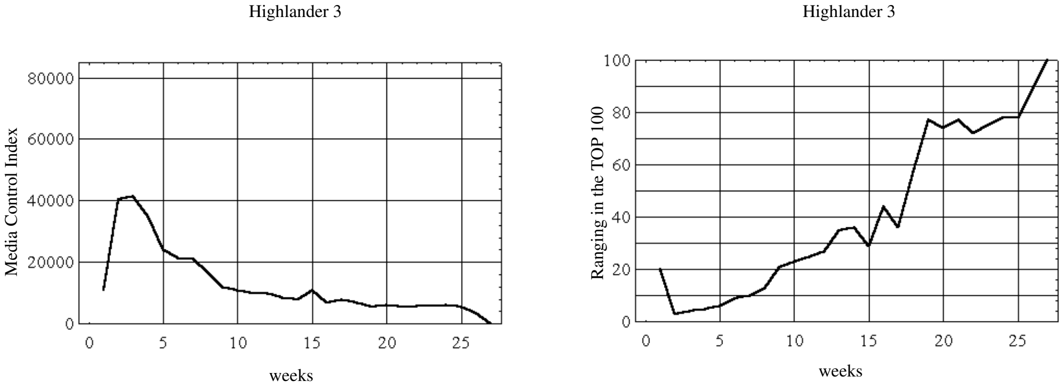

Figure 2: Data from the VideoWoche

These approaches are useful for optimizing playout, stacks or disk operations in a system, but neither take user interaction into account once that a movie is playing nor do they try to model the play time of a single clip in any way.

The modeling of VCR commands requires a model that includes modeling of user habits in applying these VCR operations. Dan and Sitaram analyze in [Dan95] the caching of data in a single server or server cluster under various interactive workloads and models information such as access skew (the distribution of requests on stored clips), the clip length distribution and the viewing time. While they do not consider the aging of individual movies, because they still need a short-term model, they consider the distribution of hits on the available videos and chose the Zipf distribution to model the video popularity. Nussbaumer, aiming at optimizing caching in a single server system or cluster, also assumes the Zipf distribution [Chen92] to model video popularity [Nuss95]. The distribution of videos or blocks of videos over multiple machines for load-levelling or availability purposes has been investigated in [Bern96].

Barnett et al. [Bar95] aim at minimizing the storage costs in a distributed system and apply caching mechanisms to do this. This requires the kind of long-term analysis we also discuss in this paper. They base their considerations for long-term popularity (in the absence of freely available video-on-demand trial results) on numbers from CD sales. The model they derive is a double exponential curve for the distribution of user accesses on videos and a movie popularity development with only one raising and one declining side. They evaluate various caching strategies. From our findings based on movie rentals, their approach is not able to model the user behavior in video-on-demand systems correctly. Data derived from a video rental store and from video rental magazines which are compared in Figure 3 show that such smoothness in popularity changes is only achieved for large user communities.

In [Dan93], Dan and Sitaram try to add to the Zipf distribution a long-term dynamic change in time. They take into account that the distribution of movie rentals at a certain date can be approximated by the Zipf distribution but that the ranking among the movies is changing each day. Their approach for taking this into account is the ordering of the movies. The largest index of a movie is equivalent to the largest index in the Zipf distribution function and the largest probability assignment to this movie. For each day, they apply a permutation function to this ordered list. We show in section 4.3, that this approach is not useful to study caching in hierarchical distribution systems.

3 Sources of Data

In order to create a model of the user interest in movies, i.e. in order to model the movies' development, external information is required. A number of sources have been considered:

- Reports from trials: Unfortunately, there are no reports on video-on-demand field trials publicly available that we know of.
- Cinema Visitor Numbers: The numbers of cinema visitors provide a source of information to model the initial user interest in new movies. This initial user interest could not be derived from movie rentals (see below) in the same quality because the number of copies for rental is limited. On the other hand the number of cinema visitors are not suitable for long-term modeling, because only the most favorite movies remain in the cinemas for a long time, and because only a limited number of movies is shown in the cinemas at a certain time.
- Literature: As described in Section 2, there are only few papers that consider the long-term development of movies. Furthermore, we were sceptical about the validity of the base assumptions and figures in the existing ones. In Section 4, we discuss the long-term models we found in the literature.
- Movie Magazines
  A couple of these magazines publish a weekly ranking, e.g. a top10 ranking of the last week's TV, cinema or rental movies. Among the few that provide more information is in Germany the VideoWoche [Vidw96], which publishes weekly the top 100 movie rentals based upon the Media Control Index. This index is calculated from the rentals of about 1000 movie rental stores nationwide. In that way regional differences are hidden and a broad data base is used. This index has a major disadvantages: The ranking function is not publicly known, there is no guarantee that two movies with the same index value have the same number of rentals. Additionally, since only one data point per movie and week is available, smoothing effects take place.
  While this information does not provide information on the user behavior, it is a good indication of movie development. As an example, the development of the movie "Highlander 3" is given in Figure 2 (note that the lowest rank is best in the right figure).
- Movie Rental Store Numbers
  Examining the rental numbers of a single rental store provides information about the probability of access of few users (some thousand) to a large base of movies. It considers completely unpopular movies as well as long-runners, single-day highlights, and all of these mixed with brand new titles. This accounts for a realistic mixture of movies in various states of their developments, and for a realistic pattern of access to the individual titles as it may be the case in an on-demand system. The biggest problem with these numbers is that only popular movies which are available in multiple copies are useful for understanding the user behavior. Each copy of a movie on its own has a very unclear development of rentals/day.

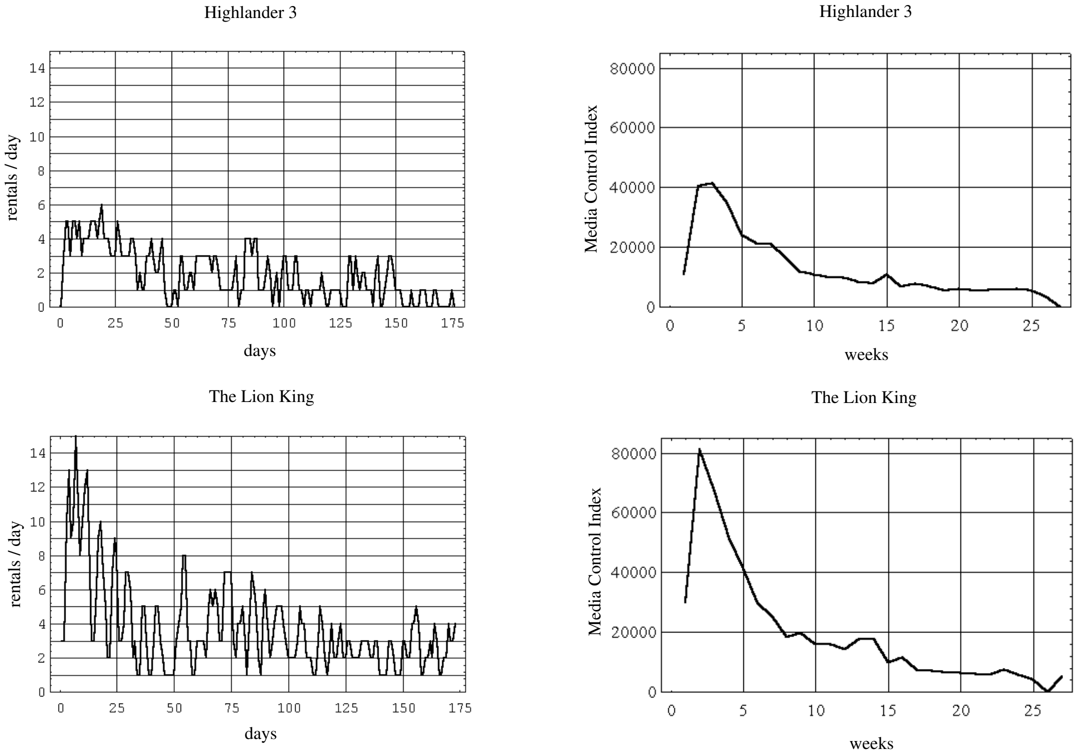

Figure 3: Comparison of rental store (left) and magazine numbers (right)

As the basis of this study, information from the VideoWoche and from a single movie rental store were available (out of a long list of shops who refused to release information). The data of the VideoWoche contained the index value of the top100 movies as well as their ranking. The data from the movie rental store contained for every rental or return transaction the date, time and movie name but no customer or price information. This allowed us to track the number of copies as well as the age of each movie.

4 Verification and discussion

We integrate the knowledge derived from various data sources into our considerations, and verify the consistency of this data first.

4.1 Large and Small User Population Data

We have to assume that the data from VideoWoche magazine and that from the movie rental store have an applicable relation to each other. By the magazine's content, we are limited to the movies that are considered the top 100 country-wide. Figure 3 shows the comparison of the data from VideoWoche magazine (right) and the curves of the rental store (left) for two movies (Highlander 3, Lion King). We have chosen these two movies as examples because they were the ones that experienced the largest number of accesses to themselves of those movies that remained in the top 100 list of VideoWoche for several weeks during the observation period. If movies with a lower number of accesses to them had been chosen, the similarity in trends with the magazine could have been illustrated only by smoothing the rentals, e.g. by showing the average rentals in three days. By selecting these movies, the similaries become visible without any smoothing. We conclude from the qualitative similarity of such curves (smoothing taking place by adding the rental numbers of 15 copies in one place versus the rental numbers of some 10000 copies in 1000 places) that the co-consideration of these two sources to build a single model would be acceptable.

4.2 Small Population Data and the Literature

Section 4.1 demonstrates that the access patterns generated by small user populations can differ widely from the average user's behavior. The following shows that this affects the usefulness of the Zipf distribution for estimating hit probabilities. The Zipf distribution is defined as

$$z(i) = \frac{C}{i}, C = 1/\left(\sum_{i=1}^{N} \frac{1}{i}\right)$$

In this formula, N is the number of available movie titles. i is the index of a movie title in the list of N movies that are sorted in the order of decreasing popularity. It is noteworthy that this distribution, which is typically used as the basis for investigations on video server operations, is completely independent of the number of users that access the set of movies.

To verify the applicability of the distribution to our data, we compare all days of one month in the period covered by the movie data and sort 250 representative movie titles by their popularity at these days. The resulting data is compared with the Zipf distribution for N=250. Figure 4 shows the first 100 entries of the resulting curves. It presents the curves for the two days with the lowest and highest hit rate on the top 10 movies in one month in comparison with the Zipf distribution for the same number of movies.

It becomes visible that the Zipf distribution, although quite similar to the actual rental probabilities, is somewhat optimistic for small user populations. The upper curve that was derived in this month may be restricted by the number of copies available in the rental store. The lower curve is not affected by this but shows that the diversity in user selection is wider than accounted for by the theoretical function. This implies that caching algorithms that are designed under the assumption that the Zipf distribution provides a worst-case or at least an average-case boundary for movie hit probability may underestimate the number of cache misses in a server at a low level of a distribution hierarchy.

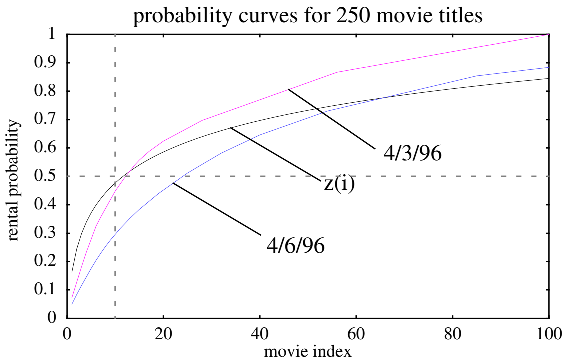

Figure 4: Rental probabilities compared with the Zipf distribution

4.3 Day-to-Day Changes

The Zipf distribution is a static distribution that includes no model for the life cycle of movies. It has already been shown by Barnett [Bar95] that the Zipf distribution in itself is not well suited to simulate real data. Because of this, it is not applicable to investigations that consider temporal changes.

In order to compensate for this, Dan and Sitaram [Dan93] have created a model based on a modified rotation of movie rental probabilities. Each individual movie title is given an index number, and its current popularity is calculated by the Zipf distribution. After a fixed amount of time, new indices are calculated by rotating the indices. To reduce large jumps in the relevance of a specific movie, the left half of the movie indices is swapped before and after the rotation. By adding the rotation, they try to simulate dynamic changes in the rental probability of individual titles for the cache of a single server. They did not try to create this model for long-term variations.

When movies are sorted according to their relevance, the graphs in Figure 5 display the size of relevance changes from one measurement to the next. Graph 1 in Figure 5 shows this for the rotation model, assuming 150 movies. Obviously, it does not provide sufficient realism for long-term considerations.

Another simple approach would be the permutation of movies' relevances between days. Graph 2 shows the relevance changes for a system with 150 movies that assumes a daily permutation.

In comparison to these two models, graphs 3 and 4 provide two examples of relevance changes for real movies from a movie rental store with a small user population. The comparison demonstrates that an algorithm which calculates the location of movies' copies in a distribution tree with respect to relevance can not be verified with either of the two models.

5 Simulation Model

5.1 Modeling the Life Cycle

The data that we have available restricts the realism of our model although we consider it much more realistic for long-term changes than previous models.

We can extract the long-term development of favored movies primarily from the data of the movie magazine, which is based on a large, representative number of movie rental shops and on a large customer base. The raw numbers that we have available from the single rental store provide a verification for these numbers. They can also be used to check the credibility of the model for long-term development of the movies because the rental numbers are continuously collected even after the movie has left the top 100 ranking list. These raw numbers are also the only basis on which we can model the development of movies that never or only pointedly hit the top 100 list and old movies that have reached the bottom line of their development and are still chosen quite frequently.

The access patterns of individual users to movies at their access point (the rental store) can on the other hand not be derived from either of these sources. Rental movies are watched "off-line", i.e.

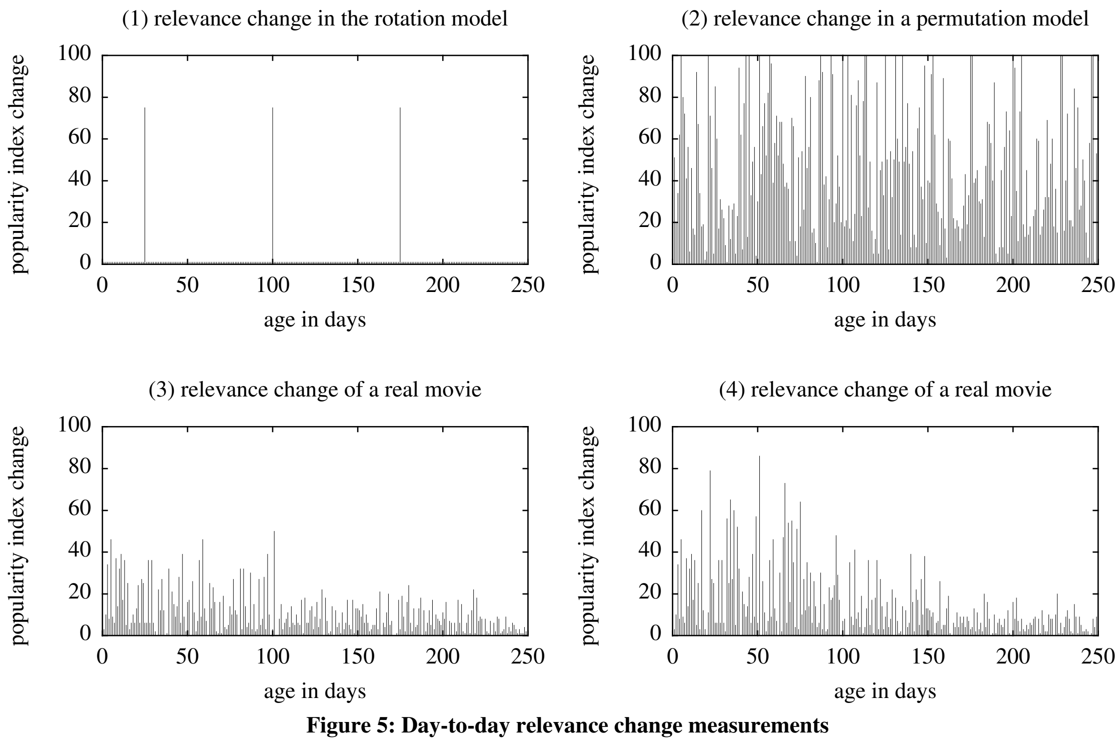

Figure 5: Day-to-day relevance change measurements

not at the time when the movie is taken away from the store. This is especially true because German rental shops that are the basis for these numbers are usually closed on Sundays, on holidays and usually from midnight to mid-morning, but we assume that the tendency of customers to pick more than one movie at the same time is similar in countries without these restrictions. The television viewer would be a better basis to build this model on, but we have not found a useful data base to model this.

We distinguish between movie modeling and user modeling. The behavior of users who want to see a movie and the development of individual movies seem to be decoupled in reality. A user's decision to watch a movie is based mostly on the time he has available rather than the existence of an interesting movie. If a relevant number of movies is inserted into our model at the same time, no equally relevant increase in rentals will be visible. Although all new movies are in the most popular phase of their life cycle the number of accesses to them remains low because the number of users, respectively the time they have available for viewing, is limited.

The independent modeling is also more convenient for future modifications. This is important to us because we hope to improve the user model in the future by collecting data on daily changes in user interests and viewing frequency. Since we don't have this data yet, a simplification that is applied in our model is the independence of user behavior from a time of day. Especially when video-on-demand is available, we assume that there is no reason for a user to arrange his own schedules according to movie timetables. Thus, the time at which a movie is retrieved is completely independent from the choice which is based on the movie's popularity at the time of retrieval.

5.2 Movie Life Cycle

Newly published movies exhibit typically, but not always, a steeply rising start peak of user interest. The observation of the rental behavior shows that all movies share a general decrease of user interest in them, but this decrease is not identical for all movies and it is frequently interrupted by increases of user interest. Once a movie has been inserted into the system, its rental probability will never return to zero. On the contrary, the relevance of old movies can be quite high. We did not make a detailed category study, but marginal checks showed that, e.g., the start peak is less relevant (although existent) for movies rated PG-18 but the sustained relevance remains generally high.

In our approaches to create a model that reflects the long-term behavior of real movies, we had experimented with discrete models to integrate the rental behavior as reflected at a rental store. These models were discarded because they needed too many parameters with no explanation for their necessity.

Splitting the available data into a an underlying curve for long-term behavior of a movie and a random effect that is mainly dependent on the size of the user population led to a more appropriate model. We observed that the underlying curve seems similar to a variation of the exponential curve that is used to describe, e.g., the spread of infectious diseases. We used a parametrized version to take into account the quantitative difference in the number of rentals, the steepness of the loss of interest and the remaining interest in a title. This function is

$$RP(t) = a \times e^{\left(\frac{2}{\sqrt{10}} - \frac{t}{10 \times b} - \frac{b}{t}\right)} + c$$

We found parameters applicable in the case of movie rentals by least square-fitting the function with the movie rental data.

Figure 6 illustrates that the parameters calculated from the least-square fitting are typically small and show no obvious correlation. The check verifies that the coefficient of correlation for

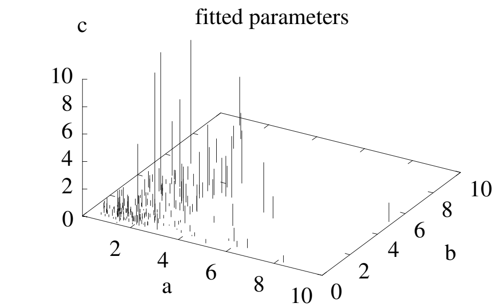

Figure 6: Parameters derived from data

each pair of parameters is tiny (the biggest is 0.01 for a and c, where a can be considered the decline of popularity and c as the remaining popularity).

Because of this observation, we select the parameters for movie modeling independently from each other. For the individual draws that determine the parameters of a newly inserted movie, we use the exponential distribution. That gives us the basic functionality of frequent small and rare big parameters until we have a larger numbers base that allows a more educated decision in this matter.

In Figure 7 we show how daily hits according to our model are distributed and compare them to the real-world rental probabilities that we used in Figure 4 before to illustrate the divergence from the Zipf distribution. From 50 draws, we show the highest and the lowest curve. The behaviour of our life cycle function is yields a more wide-spread hit distribution when compared with the real world data. It is definitely more pessimistic then the Zipf distribution. We observed also that our model will typically generate curves closer to the lower curve, which implies that a distribution of hits over multiple movies is the typical case, while the generation of higher curves, which implies the dominance of a single movie, is a rare occurrence.

5.3 User Population Size

The size of the user population in a hierarchical system is not only relevant as an overall number. The observed increase in the variation of movie popularities for small user groups can be relevant for distribution algorithms. It is not reflected by the movie life cycle and must be modeled independently. Figure 8 demonstrates how the model recreates the smoothing effects of increasing user population sizes on the rental probability of a single movie. The life cycle of the example movie and all other movies used for the experiment (initially 150, 1 new movie each day) is defined as presented in section 5.2. Each draw is a request for one

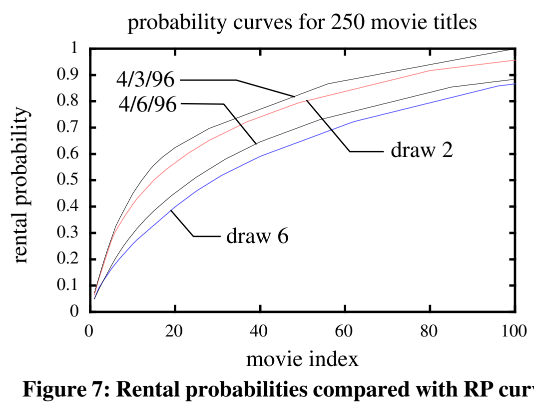

Figure 7: Rental probabilities compared with RP curves

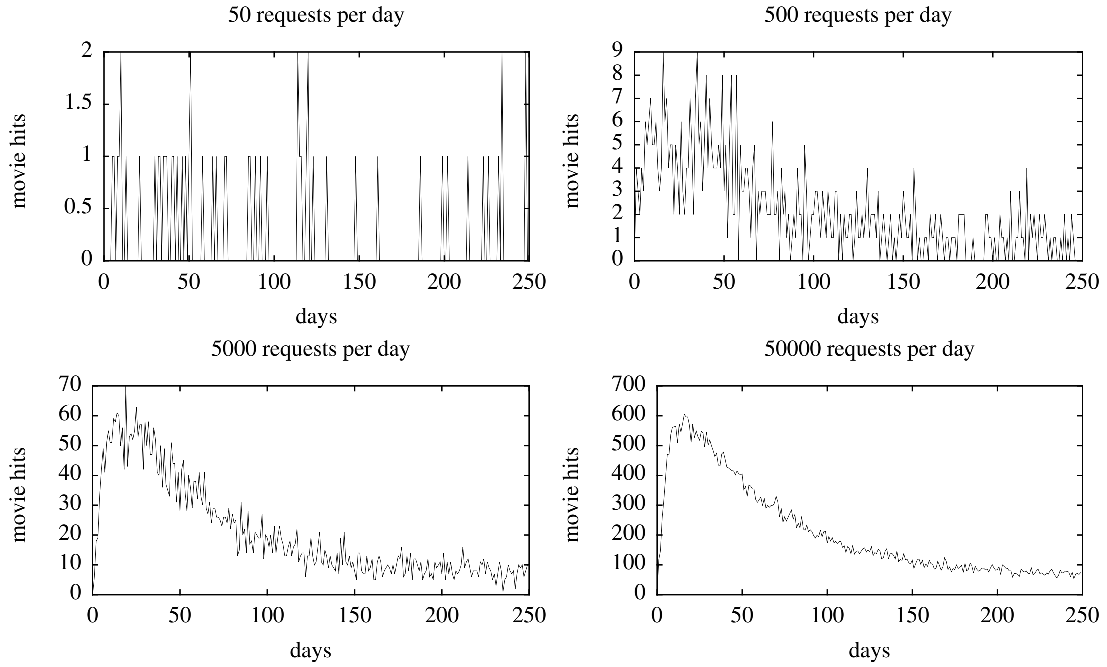

Figure 8: Smoothing effects of growing user populations

movie, the number of hits indicates how many of these draws select the example movie. In contrast to the simulation model that we develop, the number of hits per days was fixed for this experiment.

5.4 Building the Model

Figure 9 shows the simulation model that we are using. It is built with various applications in mind. It can be attached to a multitude of simulations for distribution models, allows the simple addition of an enhanced user model, and implements correctly the life cycle model that was derived from our recent studies. We have also taken care that the addition of community-specific interests can be supported by adding alternative weighting functions to the movie base.

The list of movies that are available to the system is maintained by a component called movie base. The stored information contains the age and the parameters of the specific movie title. From this information, the current popularity of the movie with respect to the overall set of available movies is computed according to the life cycle function RP(t). Since our current model has a temporal resolution of single days, and new movies are inserted at a rate of one in 12 simulated hours, the movie popularities change only twice in a simulated day. We make use of this to implement a faster movie selection by computing a second list of the movies weighted by their popularities. The popularities are normalized so that 1 is the sum of all movie popularities.

A random process, this is a Poisson process until we have more information on same-day user behaviour, decides when a user decides to view another movie. A random number in the interval]0..1] is generated and sent to the movie base. This random number identifies uniquely a movie in the list of movies weighted by their current popularity, and the movie is returned to the user as the selection. This movie title is then used in the simulation as the movie that is requested from the head-end server to which the user is connected.

We keep the global probability of one movie to be selected from a large number in line with the probability defined by its life cycle by using a single generator function. This function selects randomly from a common database for all movies. Otherwise, if we had a random selection mechanism built into the user model, probably with an independent numbers generator in the user itself, bad random number generators could spoil the results.

Due to this approach, a movie has a probability of being selected by a user of the system which is in line with its life cycle function. We assume a true on demand system in which all users are equally informed about the availability of a movie, and the movie is equally advertised towards all users. Since the topology of the video-on-demand system is assumed to be based on physical constraints and not based on user interests, the probability of a movie to be requested is the same for all users. Consequently, the probability for a request to a certain movie is the same for each head-end server that serves the same number of users.

The centralized approach has more advantages. A central movie base allows logging of the selected movie and cross-checking for viability of results. Statistics can be kept with the single movie to the end of the simulation rather than evaluating raw data afterwards. The insertion of new movies is more easily modeled if only one database of movies is maintained, and it reflects better the model of a single distributing source for the specific movie. The number of entries in this movie base is increasing throughout the simulation once per interval (e.g. daily), to account for the number of movies that are actually published.

The introduction of intervals is used for another simplification in our model. The rental probability of all movies is updated after a fixed interval (in the current model after a day). In this way, the weighted list of popularities can be re-used for the requests arriving in one interval instead of recalculating the exact probability value for each user request. This simplification saves computing time and seems permissible because the user behavior is not modeled on a single-day scale.

The user model is based on the probability that a user will want to see a (any) movie. It defines the time at which a user will see a movie. We expect to enhance this request for a movie later with certain additional information, like the category of the movie requested. It will also be enhanced towards modeling time of day. Right now, this part of the model is simple and open for enhancement in future versions of the simulation.

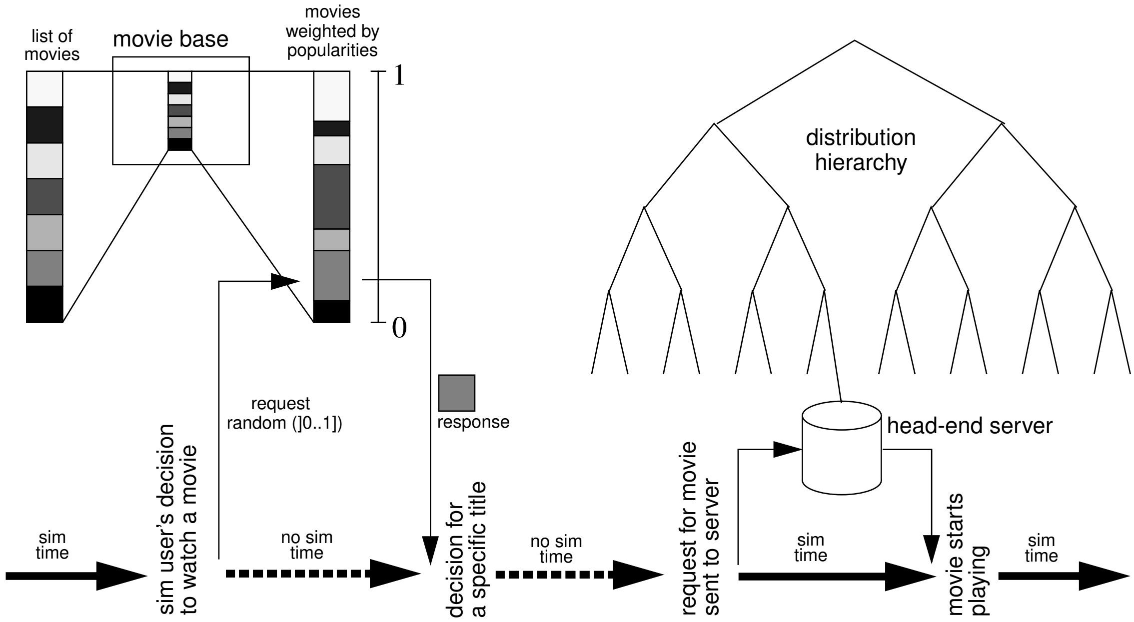

Figure 9: Selecting a movie

Users are modeled as individual entities that initiate the retrieval of a movie. We expect the number of customers of a single video-on-demand server to be approximately constant, with a roughly constant interest in watching movies, i.e. the total number of movies that are retrieved by the customers is fixed. Since there are no restrictions to retrieving any video at any time in a true video-on-demand system, the model does not take any dependencies between various users into account. The only way for users to exhibit similarities is because of inter-dependencies arising from the handling of their requests by the on-demand system.

6 Conclusion

In this paper, we have presented a new approach to movie life cycle modeling. We have shown that the current models for user requests in video-on-demand systems make assumptions that do not hold in the case of long-term investigations on distributed on-demand systems such as large-scale video-on-demand systems.

We have compared data on user requests from various sources, and deduced that two main issues in user request modeling have previously not been modeled adequately. These are the long-term life cycle of on-demand movies and the effects that varying user population sizes have on the usefulness of considering this life cycle at individual access points of the on-demand system.

Based on these observations, we have developed a model for request generation that is used as a component in simulations of video-on-demand systems.

Among the issues to be solved is that we have no parameters for a time-out value that determines when a waiting user will cancel a request. To reduce simulation time, we investigate simplifications for parts of a distribution system to generate a useful background load on higher-level servers.

Our model is able to model the day-to-day changes, but we need additional information on changes of user behavior throughout the day. It is convenient to assume that the popularity of a movie is changing slowly, but children's interests dominate the TV viewing population in the afternoon, while adults' interests dominate at night. Since this may render caches in a trivial caching server completely useless every few hours, we consider this a mandatory extension to our model.

In the future, we will use this component for the evaluation of automated distribution mechanisms in networks of on-demand servers. We have already applied the model to evaluate some well-known caching algorithms in purely hierarchical distribution networks, and we will extend our evaluations to cover both more caching algorithms and other approaches like prefetching techniques and central decision making. We will also address topologies other than strict hierarchies.

References

[Bar95] Scott A. Barnett, Gary J. Anido, H.W. Beadle, "Caching Policies in a Distributed Video on-Demand System", Proc. Australian Telecommunication Networks and Applications Conference 1995, Sydney

[Bern96] C. Bernhardt, E. Biersack, "The Server Array: A Scalable Video Server Architecture", in "High-Speed Networking for Multimedia Applications", Kluwer Academic Publishers, 1996, ISBN 0-7923-9681-2

[Chen92] Y.S.Chen, "Mathematical modelling of empirical laws in computer application: A case study", Comput. Math. Applicat., pp. 77-78, Oct. 1992

[Dan93] Asit Dan, Dinkar Sitaram, "Buffer Management Policy for an On-Demand Video Server", IBM Research Division, RC 19347

[Dan95] Asit Dan, Dinkar Sitaram, "A Generalized Interval Caching Policy for Mixed Interactive and Long Video Workloads", IBM Research Division, RC 20206 (89404), 9/28/95

[Gol96] Leana Golubchik, John C. S. Lui, Richard R. Muntz, "Adaptive piggybacking: a novel technique for data sharing in video-on-demand storage servers", Multimedia Systems (1996) 4: 140-155

[Hac96] Dick Hackenberg, "Interactive TV Gets Serious", http://www.chiatday.com/cd.www/explor/emerg/interactivetv.html, 20. September 1996

[Hes94a] H. Hessenmüller, W. Bartel ua., "Zugangsnetzstrukturen für interaktive Videodienste (Teil1)", Der Fernmelde Ingenieur(1994)8, Verlag für Wissenschaft und Leben Georg Heidecker GmbH, Erlangen

[Hes94b] H. Hessenmüller, W. Bartel ua., "Zugangsnetzstrukturen für interaktive Videodienste (Teil2)", Der Fernmelde Ingenieur(1994)9, Verlag für Wissenschaft und Leben Georg Heidecker GmbH, Erlangen

[Hod95] Winston William Hodge, "Interactive Television: a comprehensive guide for multimedia technologists", McGraw-Hill, 1995

[Litt61] J. D. C. Little, "A Proof of the Queueing Formula L=lW", Operations Research, No.9 pp. 383-387, 1961

[Litt94a] T. D. C. Little, D. Venkatesh, "Prospects for Interactive Video-on-Demand", IEEE Multimedia, Vol. 1, No. 3, Fall 1994, pp. 14-24

[Litt94b] T. D. C. Little, D. Venkatesh, "Popularity-based Assignment of Movies to Storage Devices in a Video-on-Demand System", ACM Multimedia Systems, 1994

[Nuss95] J.P. Nussbaumer, B.V. Patel, F. Schaffa, J.P.G. Sterbenz, "Networking requirements for interactive video on demand", IEEE Journal on Selected Areas in Communication, 13(5), pp.779-787, 1995

[Ov94] "Interactive Television: the Market Opportunity", Ovum Limited, 1994,
ISBN 0 903969 94 7
Judith Jeffcoate, John Mattews, "Interactive Television: The Market Opportunity Update 1", Ovum Limited, 1994
Judith Jeffcoate, Paul Lee, John Matthews, "Interactive Television: the Market Opportunity Update 2", Ovum Limited, 1995

[Ram91] R. Ramarao, V. Ramamoorthy, "Architectural Design of On-Demand Video Delivery Systems: The Spatio-Temporal Storage Allocation Problem", IEEE ICC, 1991

[Sam95] S. Sampathkumar, P.V. Rangan, "Interactive Multimedia Services on Cable TV Networks", it+ti, Vol. 37, No. 4/95

[Shah97] C.Shahabi, M.H. Alshayeji, S.Wang, “A Redundant Hierarchical Structure for a Distributed Continuous Media Server”, IDMS 97 - to appear

[Sheu97] S. Sheu, K.A. Hua, W. Tavanapong, “Chaining: A Generalized Batching Technique for Video-on-Demand Systems”, ICMCS 97 - to appear

[Tetz94] W. Tetzlaff, M. Kienzle, D. Sitaram, “A Methodology for Evaluating Storage Systems in Distributed and Hierarchical Video Servers”, COMPCON 94 Conference Proceedings, pp.430-439

[Tew95] Renu Tewari, Rajat Mukherjee, Daniel M. Dias, "Real-Issues for Clustered Multimedia Servers", IBM Research Division, RC 20108, 6/20/95

[Vidw96] "VideoWoche", Entertainment Media Verlag GmbH & Co.oHG, Stahlgruberring 11a, 81829 München