Copyright Notice

Institute of Communication Networks and Computer Engineering
University of Stuttgart
Pfaffenwaldring 47, D-70569 Stuttgart, Germany
Phone: ++49-711-685-68026, Fax: ++49-711-685-67983
Email: mail@ikr.uni-stuttgart.de, http://www.ikr.uni-stuttgart.de

Modeling and Performance Evaluation of iSCSI Storage Area Networks over TCP/IP-based MAN and WAN networks

C. M. Gauger*, M. Köhn*, S. Gunreben*, D. Sass* and S. Gil Perez†
*University of Stuttgart, IKR, Pfaffenwaldring 47, 70569 Stuttgart, Germany
{gauger, koehn, gunreben, sass}@ikr.uni-stuttgart.de
†Accenture Services GmbH, Germany, samuel.gil.perez@accenture.com

***Abstract*— This paper provides a concise modeling and performance evaluation of the iSCSI storage area network (SAN) architecture and protocol. SANs play a key role in business continuity, enterprise-wide storage consolidation and disaster recovery strategies in which storage resources are most often distributed over many distant data center locations. In the future, SAN traffic will be transported over IP-based networks, e. g., enterprise virtual private networks, to benefit from converged networks and save cost.**

In these scenarios, the impact of end-to-end delay and QoS of broadband networks on SAN performance is critical and has to be well understood by IT departments when deploying IP-storage solutions and network operators when designing transport network services for SAN applications. In this context, we propose models for iSCSI write requests over TCP/IP networks, e. g., as used in asynchronous mirroring applications. In addition to the analysis for individual requests we present—to the best of our knowledge for the first time—the evaluation of an iSCSI session under a realistic request traffic model with and without interleaving. We analyze the throughput and total request write times for different network dimensions, i. e., round-trip times, and QoS levels, processing delays in the iSCSI layer as well as request characteristics.

I. Introduction

Storage area networks play a key role in business continuity, enterprise-wide storage consolidation and disaster recovery strategies in which storage resources are most often distributed over many distant data center locations. Storage consolidation is mostly motivated by the fact that storage resources account for approximately 40% of IT hardware budgets but often 70% of them remain unused [1]. While the need for disaster recovery and business continuity is obvious in the post 9/11 business world and often regulated by laws, similar requirements can be derived from natural catastrophes. In order to guarantee business continuity in the latter scenario, data has to be mirrored in data centers hundreds or even thousands of kilometers away.

The technical evolution from direct attached storage systems to storage networks is outlined in the following subchapter. A brief overview of new emerging storage services, most important in the context of this work, is given in I-B. The different storage area network architectures are characterized in I-C. In I-D the related work is presented.

A. Storage networking evolution

Based on those requirements and in contrast to the classical *direct-attached storage* paradigm, which considers storage as part of a computer system or a local peripheral, storage is tending to be recognized as a distinct resource, separate from the host. This leads to the new *shared storage* paradigm: storage can be shared across multiple hosts, acquired, and managed independently from them. The Storage Networking Industry Association (SNIA) created a framework, the *Shared Storage Model* [2], for classifying shared storage architectures based on their realization of the application, file/record, and block layers. For instance, the difference between network attached storage (NAS) and SANs, which is often mixed up, can be clearly defined: NAS manages storage at the file level while SANs provide access storage at the block-level.

Today, SANs are mostly deployed within data centers, on campuses or within a metro area. They either do not scale to long-distance operation or require new extensions to do so. In the future, SAN traffic will be transported over TCP/IP networks, e. g., enterprise virtual private networks, to benefit from the cost savings of converged networks.

In these scenarios, the impact of end-to-end delay and QoS of broadband networks on SAN performance is critical and has to be well understood by IT departments when deploying IP-storage solutions and network operators when designing transport network services and SLAs for SAN applications.

B. Storage services

Redundancy management services such as mirroring or backup together with the applications that will help the companies to comply with the new data retention regulation, can be considered as the killer applications that will speed up the adoption of networked storage solutions in the next years. These services, classified into synchronous mirroring, asynchronous mirroring and backup, differ in requirements, especially regarding time constraints.

In synchronous mirroring, data updates must be written both at the primary location and at all secondary locations before a write operation can be considered complete. If the number of secondary sites is very high, if the network is heavily loaded or if the round-trip time is high, this can result in an unacceptable response time for applications. To reduce

the latency experienced by the application, the process can be optimized by concurrently sending data to the remote locations performing local write operations. Due to the high bandwidth requirements and the extremely tight QoS constraints for synchronous mirroring, this service is most likely only used in short-distance scenarios or over dedicated networks.

In contrast to synchronous mirroring, an asynchronous write operation can be considered complete from the perspective of an application as soon as the data have been locally registered in the operations log. Transmission to secondary locations occurs asynchronously, after application execution has resumed. Nonetheless, an acknowledgment after proper replication of the data at the remote sites is required. Due to the relaxed QoS and delay requirements of asynchronous, this service is better suited for application over MAN and WAN networks and thus focused on in our modeling and performance evaluation.

The storage service which has the least strict requirements regarding throughput and delay is the backup service. Here, large amounts of data are transferred offline and tape drives can often be considered a bandwidth bottleneck. Although we do not explicitly consider backup services in the following, throughput models for large request sizes are suitable here.

C. SAN architectures

Today, the most widely used SAN technology is Fibre Channel (FC). FC SANs are high performance storage networks over which SCSI data is transported in the local or campus area. Its distance limitation is due to the flow control algorithm which acknowledges every single transmission as well as due to the standardized transmission components. The FC Protocol supports different classes of service and provides usually an effective throughput of approximately 100/200 MB/s full duplex for 1 Gbps and 2 Gbps FC physical interfaces, respectively. The origin of FC's high performance is the implementation of the complete protocol stack in the hardware of the host bus adapter and the usage of a separate network.

To extend FC SANs beyond the LAN, the first solution would be to replace a FC link with a WAN link using SDH/SONET, ATM or WDM [3], [4]. Alternative solutions, benefiting from converged IP networks, e. g., *Fibre Channel over IP Protocol* (FCIP) and *Internet Fibre Channel Protocol* (iFCP), interconnect FC networks over IP networks and are currently under development and standardization by the IETF [5], [6].

FCIP replaces an FC inter-switch link by an IP network and sends FC frames through statically configured tunnels between the two FC switch ports while still treating the entire FC network as one. In contrast, iFCP is a native-IP protocol used to interconnect FC SANs. Its gateway-to-gateway architecture transports FC frames over IP networks separating FC islands.

The Internet SCSI (iSCSI) protocol is an entirely new architecture standardized by the IETF [7] and was designed to transport SCSI application data over TCP/IP networks, trying to bring closer these, until then separated, worlds of storage area networking and IP.

This new approach has among its main advantages the convergence of messaging and storage networks into a single communications infrastructure, as well as the better scalability of IP networks. In addition, there is the possibility to integrate widely deployed and available IP-related protocols like IPSec and finally opportunity to avoid costly specialized hardware.

In contrast to Fibre Channel, the *intelligence* does not reside in the network but, typical for IP networks, in the end-devices which allows for quicker service introduction scenarios. Also, part of the iSCSI functionality can be implemented in software more cost-efficiently. Thus, it is not surprising that iSCSI is already in the portfolio of important market players like Cisco or Microsoft—for instance, an iSCSI client is included in the Windows 2003 Server Edition.

In the past, neither the networking equipment nor the IP-related protocols were able to strictly satisfy the high bandwidth and low latency requirements of storage access. But today, with the arrival of Gigabit and 10 Gigabit Ethernet, the situation has changed. Gigabit transmission speeds, combined with the use of Virtual LANs and the QoS mechanisms to separate storage traffic from messaging traffic in the network may allow iSCSI to become a real opponent to Fibre Channel, specially in the segment of medium-sized businesses. However, the performance of this all-IP solution across WANs, i. e., in the presence of large propagation delays and QoS parameters typical for IP networks, has to be evaluated.

D. Related Work

Several hardware vendors have performed practical experiments to demonstrate iSCSI capabilities concentrating on short distance scenarios. Alachritec and Nishan Systems demonstrated how iSCSI was able to reach wire-speeds with equipment placed in the same building [8]. In [9], a study of iSCSI performance across a campus is reported.

In [9] and [10], experiments were conducted for hardware and software implementations. Both papers showed that software iSCSI implementations can keep up with FC implementations for local installations, however also identified the strong impact of network quality and distance.

In [11], results of studies for single request transmission are also reported but no information on the model or on the analysis is provided.

Summarizing, most of these reports focus on the experimental side of the application and do not comprehensively address the impact of typical network QoS, traffic models or other protocol-specific parameters. Also, a model of iSCSI under realistic request traffic with and without interleaving has not been reported so far.

Thus, the objective of this paper is to model the iSCSI architecture and to propose models for iSCSI write requests over TCP/IP networks, e. g., as used in asynchronous mirroring applications. In addition to the analysis for individual requests we present—to the best of our knowledge for the first time—the evaluation of an iSCSI session under a realistic request

traffic model with and without interleaving. We analyze the throughput and total request write times for different network dimensions, i. e., round-trip times, and QoS levels, processing delays in the iSCSI layer as well as request characteristics.

The remainder of this paper is structured as follows. Section II describes the iSCSI protocol and introduces models for single request transmission and for dynamic traffic. Then, section III outlines our evaluation scenarios which are then used in a comprehensive performance evaluation in section IV. Finally, section V summarizes this paper and gives an outlook on further work.

II. Architecture and Modeling of the iSCSI Protocol

In this section, we describe models for a single iSCSI write requests and for the superposition of iSCSI write requests over a single TCP connection with and without interleaving.

The iSCSI layer manages the peer-to-peer relation between an *initiator* and a *target* entity in a so-called *session*. It encapsulates SCSI application data in iSCSI protocol data units and sends them over one or multiple TCP connections through the IP network.

An iSCSI session begins with the establishment of the first TCP connection between initiator and target. In the following *login phase* authentication, negotiation of security and operational parameters is performed. After the login phase, the connection enters the so-called *full feature phase*, during which the actual data transfer occurs. Assuming that initiator-target iSCSI sessions run for a long time compared to individual request durations, we only consider the full feature phase in the following.

In the following performance evaluation of the system, the two most important criteria are examined, which are the write time defined as the delay between begin and end of a write operation as the sender has to wait for completion of the request for this time and the throughput as it limits the rate that can be used for transmitting data to the disk.

A. Model for a single iSCSI write operation

Figure 1 shows a sequence diagram of an iSCSI write request. The iSCSI PDU is split up in so-called bursts and transmitted in several rounds according to the flow control protocol. As all bursts of an iSCSI PDU have to be sent over the same TCP connection and as the standard defines a default value of one TCP connection per session, we only consider this case in the following.

Together with the iSCSI write command, a first burst of *unsolicited* data can be transmitted. The amount of unsolicited data is limited by the value b_{init} for which a default value of 64 kB is recommended.

The *solicited* data transfer phase starts, when the sender has received the initial *R2T* (ready to transfer) command. Then, bursts of data up to a maximum size of $b_{\max}$ (default 256 kB) can be sent out for each R2T received.

As we are mostly dealing with the performance of iSCSI transfers over MAN and WAN networks, we do not model delays for processing of iSCSI requests in the initiator and in the target node in greater detail but aggregate them in one single processing delay T_{proc}. This delay is added to each iSCSI burst independent of its length as is also illustrated in Figure 1.

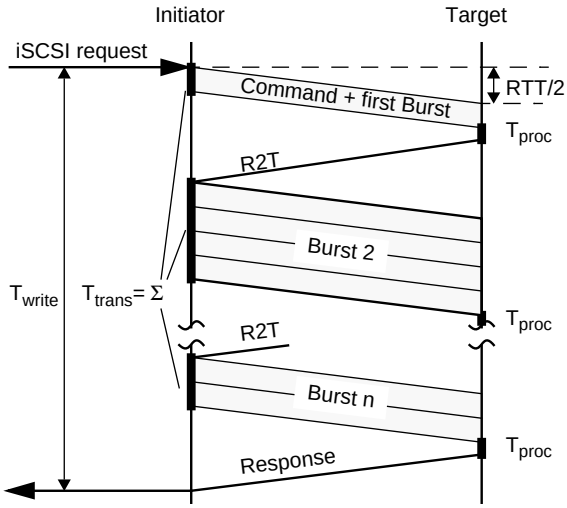


Fig. 1. iSCSI write operation

The duration of an iSCSI write operation, T_{write}, can be derived from Figure 1 as

$$T_{\text{write}} = T_{\text{trans}} + N \cdot (T_{\text{proc}} + \text{RTT}). \tag{1}$$

Here, T_{trans} is the transmission time of a request of size S and RTT is the round-trip time delay of the initiator-target relation. The number of data bursts N, i. e., transmission rounds, for processing an iSCSI request of size S is determined by

$$N = 1 + \left\lceil \max\left(0, \frac{S - b_{\text{init}}}{b_{\max}}\right) \right\rceil. \tag{2}$$

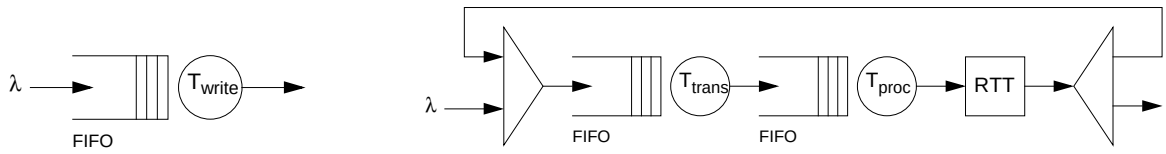
While we assume T_{proc} and RTT to be system parameters, T_{trans} depends on the throughput B of the underlying TCP/IP network. Thus, we can write $T_{\text{trans}} = S/B$.

Following limiting formulas can be given for large and small iSCSI requests. For large write requests, i. e., $S \gg b_{\max}$, the impact of b_{init} and the initial transmission round as well as of the ceiling function can be neglected and we obtain the asymptotic iSCSI throughput

$$\frac{S}{T_{\text{write}}} < \left(\frac{1}{B} + \frac{T_{\text{proc}} + \text{RTT}}{b_{\max}}\right)^{-1} \quad \text{for } S \gg b_{\max}. \tag{3}$$

For small write requests, i. e., $S \leq b_{\text{init}}$, which can be completely transmitted in an initial unsolicited burst, the maximum iSCSI throughput is obtained for a request of size b_{init} and thus following relation holds

$$\frac{S}{T_{\text{write}}} < \left(\frac{1}{B} + \frac{T_{\text{proc}} + \text{RTT}}{b_{\text{init}}}\right)^{-1} \quad \text{for } S \leq b_{\text{init}}. \tag{4}$$



(a) Non-interleaved operation

(b) Interleaved operation

Fig. 2. iSCSI model for dynamic request traffi c

With a given request size distribution $P(x)$ and a mean request size of $\bar{S}$ the mean iSCSI write time can be expressed using (1) as

$$\bar{T}_{\mathrm{write}} = \frac{\bar{S}}{B} + \bar{N} \cdot (T_{\mathrm{proc}} + \mathrm{RTT}). \tag{5}$$

Here, the mean number of transmission rounds is given by

$$\bar{N} = 1 + \sum_{i=1}^{\infty} i \cdot P(x_{i-1} < S \leq x_i) \tag{6}$$

with $x_i = b_{\mathrm{init}} + i\, b_{\max}$.

B. Model for iSCSI write operations under dynamic traffic

So far, the iSCSI model only considers write time and throughput for the transfer of a single request. However, in an operational iSCSI system, many sources, e. g., applications or hosts, may share one iSCSI session and thus requests arrive to the initiator following stochastic processes for interarrival time and request size. We consider both non-interleaved and interleaved operation over a single TCP connection.

1) Non-interleaved operation: Requests for non-interleaved operation are processed serially, i. e., each request has to be completely transferred before the next request transfer can start. Thus, requests might have to be queued in the initiator before transfer. This behavior can be modeled by a FIFO single server queue in which the arrival process follows a request traffic model and the service time represents the complete write time of a request as described in section II-A.

This model is depicted in Figure 2(a) and can be used for analysis as well as for simulation. As we apply interarrival time and request size models based on published empiric data, we used this model for simulations only.

2) Interleaved operation: For interleaved operation, several requests can be processed in parallel which will presumably yield a better utilization of the TCP connection and thus improvement in throughput and write times. Regarding the write time of a request, transmission time and processing time are still strictly serialized. Only the round-trip time component can be parallelized as a request can transmit data while a burst of another request waits for its R2T. Consequently, the single request write time is a lower bound of the write time for the request size S under dynamic traffic.

This behavior can be modeled by a queuing network as depicted in Figure 2(b). Arriving iSCSI requests loop through the system several times and only leave when they are completely processed. In each loop, one burst of the request is served in a tandem queueing system and a delay element. The first single server FIFO queue represents the transmission time T_{trans}, the second single server FIFO queue the processing delay T_{proc}. The delay element representing the round-trip time is modeled by an infinite server queue and can thus accommodate several requests in parallel.

C. TCP Layer Model

Until now the bandwidth provided by the underlying IP network has been abstracted by B. In this section, we describe the selected model for calculating B considering the network's round-trip time and QoS. For the MAC layer, we assume Ethernet which is the most likely choice for iSCSI SANs.

In literature, several models for the throughput of TCP are reported (e. g. [12], [13]). A class of simple TCP models abstracts the network by random packet loss and fixed round trip time and focuses on long-lived TCP connections. We apply such a model as iSCSI initiator-target sessions exist for a relatively long time and thus satisfy the modeling assumptions well. In those models, the throughput is determined by three limiting factors, namely the congestion window, the size of the receiver's advertised window and the bandwidth of the bottleneck link. These limiting constraints yield $B = \min(B_{\mathrm{CWnd}}, B_{\mathrm{RWnd}}, B_{\mathrm{acc}})$.

The impact of the congestion window is modeled for the congestion avoidance phase yielding a TCP throughput of $B_{\mathrm{CWnd}} = C \cdot b_{\mathrm{MSS}} / \left(\mathrm{RTT} \cdot \sqrt{p}\right)$, where $C = 0.93$ is a constant of proportionality (assuming random loss, delayed ACK strategies), $b_{\mathrm{MSS}} = 1460$ B is the maximum segment size (assuming Ethernet), RTT is the round trip time, and p is the packet loss probability [12].

For low loss probabilities, TCP throughput is not limited by the congestion avoidance algorithm but by the receiver's advertised window and is thus given by $B_{\mathrm{RWnd}} = W_{\max}/\mathrm{RTT}$, where $W_{\max}$ is the maximum advertised receiver's window, for which a value of 64 kB is commonly used, e. g., in Linux operating system implementations.

For a very short RTT, TCP throughput is only limited by the bandwidth B_{acc} of the bottleneck link which often is the network access link.

For a greater RTT, TCP throughput is either only limited by B_{CWnd} or only limited by B_{RWnd}, depending on the network loss probability. Comparing the expressions for B_{CWnd} and B_{RWnd}, we can derive a critical loss probability

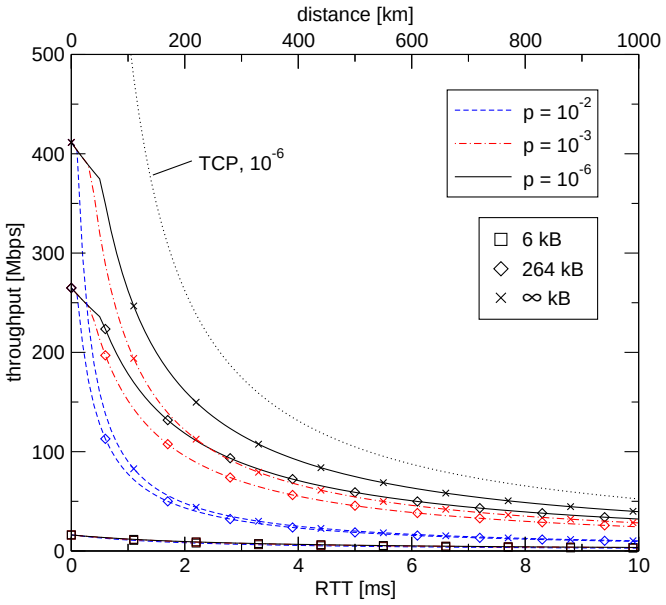


Fig. 3. Single request model: throughput vs. RTT

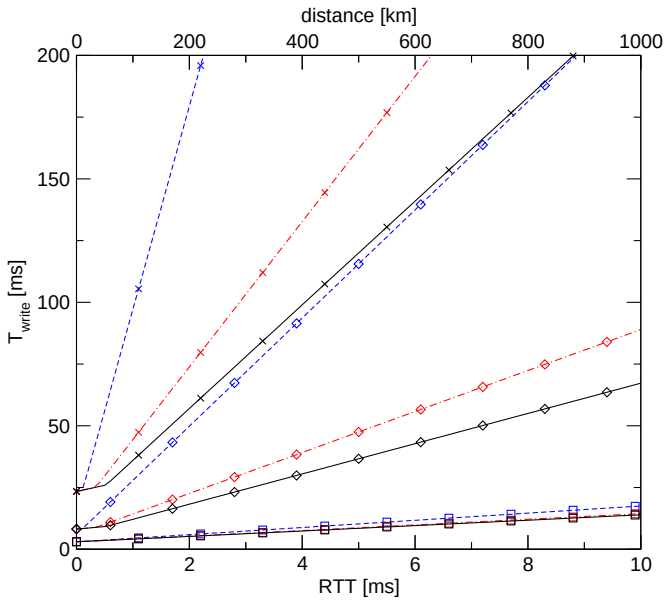


Fig. 4. Single request model: write time vs. RTT

$p_c = (C \cdot b_{\mathrm{MSS}}/W_{\max})^2$ which is independent of RTT. For $p > p_c$ the congestion avoidance algorithm limits the throughput while for $p < p_c$ the receiver's advertised window is dominant and lower loss probabilities do not have any beneficial impact. Using the parameter values introduced so far, we obtain $p_c = 4.29 \cdot 10^{-4}$.

The transition from the case of access bandwidth limitation to limitation by either congestion avoidance or window size occurs for different round-trip times, namely $\mathrm{RTT} = W_{\max}/B_{\mathrm{acc}}$ and $\mathrm{RTT} = C \cdot b_{\mathrm{MSS}}/\left(B_{\mathrm{acc}} \cdot \sqrt{p}\right)$, respectively.

III. iSCSI EVALUATION SCENARIO

For a systematic study of asynchronous mirroring over iSCSI across MANs and WANs, we took a scenario-based approach regarding network QoS, request size, and distance, which translates into round-trip time. These scenarios allow for a comprehensive evaluation due to a broad yet tractable range of values for the key influencing parameters.

As shown in section II-C, TCP throughput is determined by distance, loss probability of the network and the bottleneck bandwidth, which we assume to be 1 Gbps as in GEth.

We discuss three principal scenarios regarding the (airline) distance between storage devices' locations. A short distance scenario could be assumed to have a distance of less than 20 km, i. e., storage devices are located on the same campus. A medium distance scenario is for a distance of approx. 200 km, which can be assumed to be for a metro or regional area. Also, this distance is often considered as a minimum to allow for disaster recovery in case of natural disasters, or terrorist attacks. Finally, a nation-wide or global scenario has distances around or above 1000 km.

As we assume queueing delays in high-speed WANs to be small compared to propagation delays we model the round-trip time by propagation delays only. Consequently, the distances of 20 km, 200 km and 1000 km can be translated into round-trip times of approx. 0.2 ms, 2 ms and 10 ms, respectively.

For the investigation of the impact of network QoS, we restricted ourselves to three different loss probabilities. An uncontrolled, best effort network with a rather high loss probability of $p = 10^{-2}$, an engineered network with loss probability of $p = 10^{-3}$ and a QoS-enabled network, e. g. an IP/WDM solution [14], with a loss probability of $p = 10^{-6}$. In the former two cases, TCP throughput is limited by the congestion avoidance algorithm, whereas, in the latter case the receiver's advertised window is the limiting factor.

Apart from the network specific properties, we also study the impact of request characteristics. Considering the protocol mechanisms for the evaluation of a single iSCSI write request, we use three values for the request size which are representative for the number of rounds a request transmission comprises, c. f. Figure 1. A request of size $S = 6$ kB requires only a single round while $S = 264$ kB yields two rounds. Also, we consider the limiting case of very large request sizes, i. e., $S \to \infty$.

For the performance evaluation of iSCSI write operations under dynamic request traffic, realistic models characterizing the stochastic processes for interarrival time and request size are required. However, only very few appropriate characterizations are available in literature.

In our studies, we use an empirical model based on characterizations reported in [15]. Hsu et al. analyzed I/O traffic on the physical block level for application in design and analysis of storage systems. They consider different computing systems (file server, database, time sharing server) with direct-attached storage devices and different usage scenarios (engineers, graduated students, secretary, managers). As this block-level request model covers a wide range of user applications,

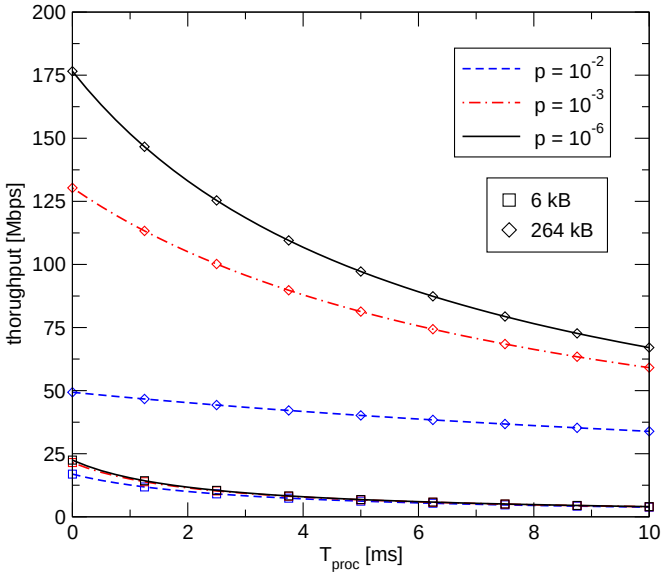


Fig. 5. Single request model: impact of the processing time

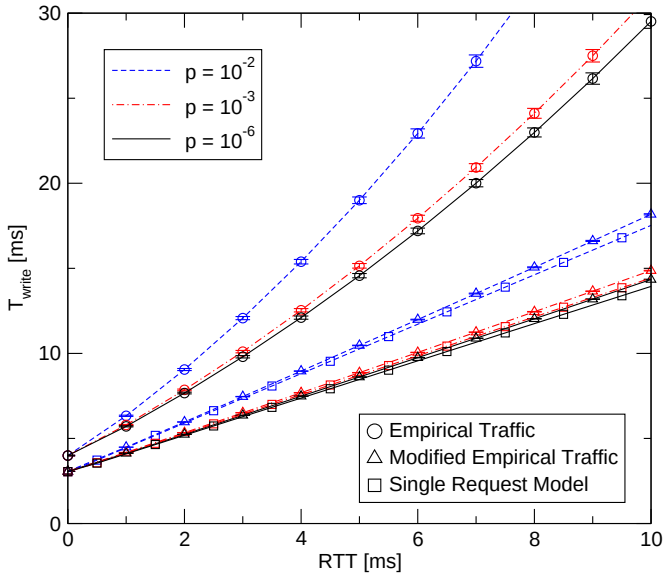


Fig. 6. Non-interleaved model: impact of the traffi c model

we consider it to be also applicable for asynchronous mirroring over SANs.

The traffic has a very bursty, self-similar nature (with a Hurst parameter ranging from 0.79 to 0.9) and there were long intervals with no or very little arrivals. Furthermore, write requests (with a share of approx. 60%) and read request exhibit very similar I/O characteristics.

The request interarrival time process is described by a lognormal distributed random variable, for which parameters were fitted to $(\mu, \sigma^2) = (-4.5, 6.25)$ representing the mean and the variance respectively (Figure 6 in [15]). We use the empiric distribution of the request size (Figure 1 in [15]) with a mean request size of 6 kB (which also corresponds to the small request size scenario introduced above).

IV. iSCSI MODELING RESULTS

In this section, we will discuss the results of a systematic performance evaluation for the single request model introduced in II-A and for the models for dynamic request traffic with non-interleaved and interleaved operation introduced in II-B.1 and II-B.2, respectively.

A. Results for single iSCSI write operations

In Figure 3, the impact of the round trip time on the iSCSI throughput is depicted for the loss probabilities 10^{-2}, 10^{-3}, and 10^{-6} and the request sizes 6 kB and 264 kB. In order to show the upper bound of the iSCSI throughput, the results for infinitely large requests are plotted. Further, for reference, the maximal TCP throughput is shown for a loss probability of $p = 10^{-6}$.

In principle, with increasing the round trip time the iSCSI throughput is decreased in all cases. As shown in (3) for large requests or in (4) for small requests, the iSCSI throughput depends reciprocally on RTT. Also, the bandwidth provided by the underlying TCP connection is decreased for increasing RTTs or for high loss rates.

For small RTTs knees can be seen resulting from TCP's change from being limited by the bottleneck link bandwidth to the limitations due to the congestion avoidance algorithm or the receiver's advertised window. In the short distance scenario, the bandwidth of the bottleneck link dominates always while the in the metro and global scenario, either the congestion avoidance algorithm or the receiver's advertised window limit throughput depending on the loss rate.

The impact of the iSCSI protocol can be seen by comparing the curve for infinitely large requests and the maximal TCP throughput. The gap only results from the round trip and the processing times as during this phase the channel cannot be used for the write request.

Comparing different request sizes, it can be observed that increasing the request size also increases the iSCSI throughput. The case of a infinite request size is the upper bound of the iSCSI throughput for single requests. The dependency on the request size can be explained by the number of rounds that have to be completed for the request. For small requests one complete cycle, carrying only a small payload, is enough while for a large or even infinite request size, all cycles carry the maximum payload. Thus, the ratio between the number of transmitted bytes and the overhead due to the propagation delay is worse for small requests than for large.

Finally, quantifying the impact of the network QoS exhibits an interesting trade-off. Improving the network's service quality can be used to either increase the distance while keeping the throughput constant or to increase the throughput for the same distance. For example, reducing the loss rate by one order of magnitude from 10^{-2} to 10^{-3}, e. g., by introducing traffic engineering, increases the maximum reachable distance by a factor of three for a throughput of around 75 Mbps.

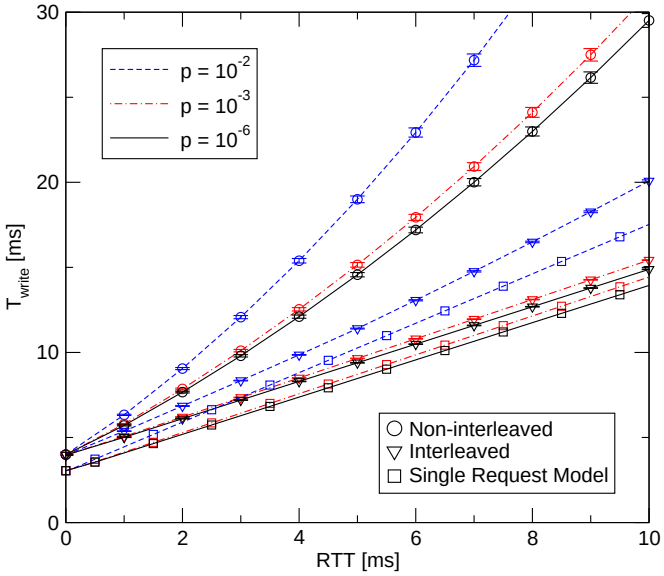


Fig. 7. Empirical traffi c model: impact of interleaving

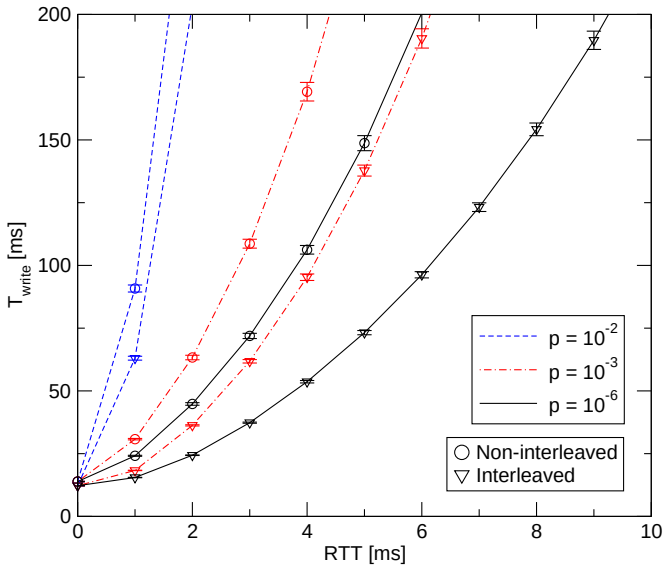


Fig. 8. Empirical traffi c model with large requests (264 kB)

However, for storage applications not only the throughput is relevant, but also the absolute value of the write time is an important performance metric. In Figure 4, the duration of a write request T_{write} is plotted versus the RTT for the three loss scenarios and the request sizes of 6 kB, and 264 kB using the same line styles as in Figure 3. As a infinite request size lead to a infinite write time, here for large requests we use a size of 1024 kB.

Again, the bent separates each curve into two segments. For small distances, the TCP throughput is limited by the bottleneck bandwidth which does not depend on RTT and thus, the transmission time is independent of RTT. So, the slope of the write time only depends on RTT's direct impact on the iSCSI protocol. For larger distances, the larger slope can be explained by the limitation of the TCP throughput by B_{RWnd} or B_{CWnd} that both depend on the RTT.

In II-B.2, we explain that the throughput can be increased by interleaving several independent requests whereas the single request write time can not be reduced without changing the protocol or the time constants. Thus, the write time of a single request as examined here is the best case and we will use this as a reference for studies with dynamic traffic conditions.

As mentioned above, not only does the RTT impact the performance but also does the processing time in the target T_{proc}. In Figure 5, the iSCSI throughput is plotted versus the processing time for the metro scenario and request sizes of 6 kB and 264 kB for all three loss scenarios.

It can be seen that decreasing T_{proc} increases the iSCSI throughput for all scenarios. For small requests, the throughput can be increased by a factor of 4 when reducing T_{proc} from 10 ms to 2 ms, e. g., by introducing offload engines for TCP or iSCSI processing. For large requests, at most a factor of two can be realized as RTT and transmission time have a higher impact. Especially in case of a high loss rate, only marginal performance enhancements can be achieved.

B. Results for the dynamic iSCSI model

After investigating the system for single requests, in the following dynamic traffic models are taken into account. We first use the queueing model introduced in section II-B.1 that does not allow interleaving of requests and compare the results later to the model with interleaving as introduced in section II-B.2. Unless stated differently, we use the empirical traffic model introduced in section III.

In Figure 6, the iSCSI write time is plotted versus the RTT. In general, for the empirical model the mean iSCSI write time increases strictly with the RTT. Compared to the single request model calculated by using (5), the dynamics of the empirical traffic model lead to a higher delay. As the single request model considers the duration of transmission, processing and the RTT, the mean write time in the dynamic case exactly differs by the delay introduced by the queues.

In order to analyze the impact of the specific traffic model, i. e., self-similarity and burstiness, we also plot the curves for a modified traffic model. This traffic model applies the same request length distribution as the empirical model but the request arrival process is now Poisson.

The mean write time for the modified empirical traffic model is clearly smaller than the mean write time for the empirical traffic model and only slightly higher than for the single request model. This is due to the fact that the lognormal distribution leads to a bursty behavior and thus to a higher queueing delay while for this load the queueing delay for a traffic with Poissonian arrivals is low.

The performance of the system can be improved by introducing interleaving of requests. In Figure 7, the mean write time is plotted versus the RTT for the empirical traffic model

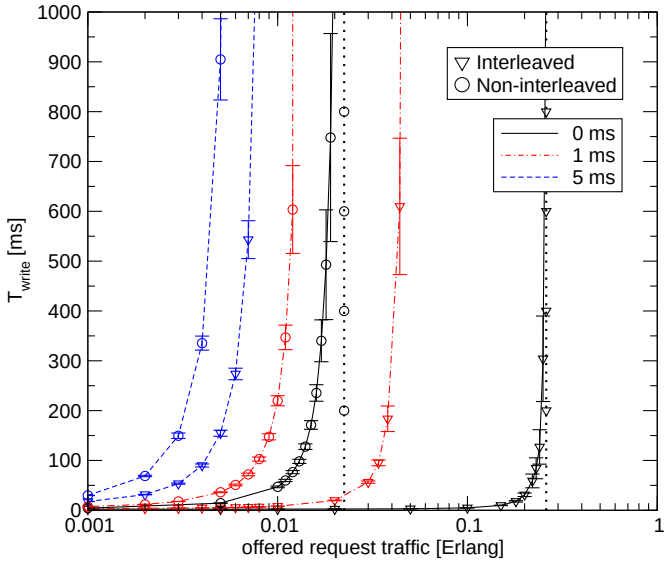


Fig. 9. Impact of offered request traffi c

for the interleaved model. Additionally, the graphs for non-interleaved operation and the curves for single requests are plotted again.

With interleaving, the write time grows almost linearly with RTT. For all loss scenarios, the write time in the interleaved case is much smaller than in the corresponding non-interleaved case and for this load it is only slightly higher than in the reference case. Thus, queueing delay is effectively reduced even with a simple first-come-first-serve scheduling discipline as the iSCSI session is not exclusively used by a single request but shared among several requests.

Having in mind the results for different request sizes as discussed above the question arises whether the request size has an impact on the usability of interleaving. So, the empirical traffic model has been changed keeping the interarrival time distribution while using a constant request size of 264 kB. In Figure 8, the results for non-interleaved and interleaved operation are plotted while the reference curves for the single request model are omitted for clarity of presentation.

It can be seen that the interleaved transmission outperforms the non-interleaved model of the respective QoS scenario. But in contrast to the scenario with small requests, the write time also grows fast for the interleaved case. For the traffic model with small requests most of the transactions can be finished within one single round whereas now always two rounds are needed.

Comparing Figures 7 and 8, it can be observed that in the scenario with small requests the performance is more improved by introducing interleaving than by enhancing the network QoS. In the large request scenario, the benefit from introducing interleaving is comparably small while the system's performance is increased by reducing the loss rate.

Finally, we investigate the impact of the offered request traffic explicitly. In Figure 9, the write time is plotted versus the offered request traffic for a metro scenario, i. e., a round trip time of RTT $= 2\,\mathrm{ms}$, with a loss rate of $p = 10^{-6}$ for interleaved and non-interleaved operation. The processing delays T_{proc} are chosen to be $T_{\mathrm{proc}} = 0\,\mathrm{ms}$, $T_{\mathrm{proc}} = 1\,\mathrm{ms}$ and $T_{\mathrm{proc}} = 5\,\mathrm{ms}$.

It can be seen that the write time grows only slowly for increasing offered traffic as long as the offered traffic is low. As soon as it reaches the maximal throughput, which depends on the scenario, the write time increases very steep. At this point, the system becomes instable as the offered traffic is higher than the capacity of the entire system.

It has to be mentioned here, that the offered request traffic in Figure 9 is determined with respect to the link bandwidth and not the system's bandwidth. So, neither the processing time nor the impact of RTT is considered for calculating the offered request traffic.

For the interleaved case, the throughput is limited by the processing in the target and the capacity of the channel. In the non-interleaved case also the exclusive usage of the channel by a single request and thus the RTT has to be considered. For reference, the asymptotes of the maximal throughput for a processing delay of $0\,\mathrm{ms}$ are depicted as bold dotted lines for the interleaved ($0.26\,\mathrm{Erl}$) and the non-interleaved case ($0.022\,\mathrm{Erl}$).

V. Conclusion and Outlook

In this paper, we proposed, discussed and evaluated models for a iSCSI write requests over TCP/IP networks. Further, we introduced relevant and realistic evaluation scenarios in order to get a broad yet tractable range of values for the key influencing parameters for SAN applications.

The models of the iSCSI write operation comprise single requests as well as dynamic traffic requests with and without interleaving. We model the iSCSI layer based on the IETF protocol specification. For the TCP layer we selected the most suitable model used in literature.

In order to study the achievable performance of iSCSI for asynchronous mirroring and backup services, we defined several realistic scenarios regarding different distances, i. e., ranging from a LAN in a campus area up to a nation-wide or global WAN, and different levels of network QoS.

In a comprehensive performance evaluation, we first systematically studied the principal behavior of the iSCSI system based on the single request model in the different scenarios. Among the impact of RTT on the maximum throughput and the duration of a write request, the trade-off between network QoS and processing power has been pointed out.

Further, the impact of dynamic traffic, i. e., the arrival process and the request length distribution, has been analyzed, especially by applying a realistic request traffic model based on empirical data. Here, we showed the benefits that can be achieved by introducing interleaving especially in case of a small mean request size. Also, we quantified the impact of interleaving on the maximal throughput and identified limits for stable operation.

Future work could compare the performance of iSCSI to FCIP and iFCP in order to show the benefits and drawbacks of

the different architectures. Furthermore, measurements taken in real iSCSI systems under dynamic traffic can show the quality of the model and the derived statements.

VI. Acknowledgments

The authors would like to express their gratitude to M. Scharf for his support regarding TCP modeling. This work was funded by EC in the Integrated Project NOBEL within the VI$^{\text{th}}$ Framework Program of IST.

References

[1] M. Lynch and McKinsey & Company, "The Storage Report, Customer Perspectives and Industry Evolution," June 2001.

[2] Storage Networking Industry Association Technical Council, "SNIA Shared Storage Model. A framework for describing storage architectures," http://www.snia.org, April 2003.

[3] M. Köhn, "Comparison of SDH/SONET-WDM Multi Layer Networks with static and dynamic optical plane," in *Proceedings of the 9th Conference on Optical Network Design and Modelling (ONDM 2005)*, February 2005, pp. 403–412.

[4] C. Gauger, "Viability and performance of Optical Burst Switching," in *Proceedings of the 9th European Conference on Networks and Optical Communications (NOC 2004)*, 2004, pp. 466–473.

[5] C. Monia, R. Mullendore, F. Travostino, W. Jeong, and M. Edwards, "iFCP - A Protocol for Internet Fibre Channel Storage Networking ," http://www.ietf.org/internet-drafts/draft-ietf-ips-ifcp-14.txt, 12 2002.

[6] M. Rajagopal, E. Rodriguez, and R. Weber, "Fibre Channel Over TCP/IP (FCIP)," RFC 3821 (Proposed Standard), July 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3821.txt

[7] J. Satran, K. Meth, C. Sapuntzakis, M. Chadalapaka, and E. Zeidner, "Internet Small Computer Systems Interface (iSCSI)," RFC 3720 (Proposed Standard), Apr. 2004, updated by RFC 3980. [Online]. Available: http://www.ietf.org/rfc/rfc3720.txt

[8] Alacritech, "Achieving Wire-Speed iSCSI Performance," 2001.

[9] Y. Lu and D. H. C. Du, "Performance Study of iSCSI-Based Storage Subsystems," *IEEE Communications Magazine*, August 2003.

[10] S. Aiken, D. Grunwald, and A. R. Pleszkun, "A Performance Analysis of the iSCSI Protocol," in *20th IEEE/11th NASA Goddard Conference on Mass Storage Systems and Technologies (MSS'03)*, 2003.

[11] D. T. Telikepalli, Radha and J. Yan, "Storage Area Network Extension Solutions and Their Performance Assessment," *IEEE Communications Magazine*, April 2004.

[12] M. Mathis, J. Semke, and J. Mahdavi, "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm," *ACM Computer Communication Review*, vol. 27, no. 3, 1997.

[13] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: A simple Model and its Empirical Validation," 2004.

[14] J. Elmirghani and I. H. White, "Optical storage area networks," *IEEE Communications Magazine*, vol. 43, March 2005, feature section.

[15] W. W. Hsu and A. J. Smith, "Characteristics of I/O traffi c in personal computer and server workloads," *IBM Systems Journal*, vol. 42, no. 2, 2003.