TIMESLOT-BASED RESOURCE MANAGEMENT IN GRID ENVIRONMENTS

Christoph Barz, Uli Bornhauser, Peter Martini, Markus Pilz
Institute of Computer Science IV
University of Bonn, Germany
{barz,ub,martini,pilz}@cs.uni-bonn.de

ABSTRACT
The usage of static timeslots is a well-known approach to handling advance reservations in the scope of Grid resource management. In this paper we enhance the timeslot-based approach by introducing dynamic timeslots and the notion of granularity. The main contribution is the development of an analytical model of the interrelation between user and system parameters in order to describe their impact on system efficiency and responsiveness.

KEY WORDS
Advance Reservation, Network Resource Management, Grid Computing, Mathematical Analysis

1 Introduction

The emerging scientific Grid community uses high performance clusters, storage systems as well as scientific instruments to support large scale distributed applications and workflows. Examples of applications are distributed simulations and sensor data analysis that use the combined computational performance and data storage of multiple clusters. In this context, advance reservations [1, 2] and meta-scheduling [3] enable a reliable coordination of different resources involved with a dedicated quality of service (QoS). This coordinated use of distributed resources requires high speed network connections that can be realized by a bandwidth on demand service capable of advance reservation. We see these network resources as another type of reservable grid resource which can be used for synchronous inter-resource communication or asynchronous file transfers. This paper is motivated by the usage of network resources in the context of advance reservations. However, the presented results are also applicable to other managed resources.

The general service model of advance reservations has previously been described in other studies [1, 2]. A basic form of an advance reservations request in the network domain assures dedicated bandwidth between two end-points for a time interval in the future. For file transfers, where a fixed amount of data has to be transmitted, the concept of malleable advance reservations is being used [4, 5]. Here, only general capabilities of the sender and the receiver such as the maximal transfer rate and timing constraints for the transmission have to be regarded. An example in the area of Grid computing is the transmission of required input data to clusters before the computation can begin (pre-staging).

The processing of advance reservation requests is done in three phases. During pre-processing (phase 1), a topology representing the minimal available resources during the requested time interval is determined. Then in phase 2, a feasible path is determined and admission control is done. If the request is accepted, the resources are reserved in the post-processing phase (phase 3). In case of malleable reservations, phase 1 and 2 may have to be repeated for various reasonable configurations as the solution also includes the search for a suitable time interval and the corresponding transmission rate. Clearly, the resource management needs to be considered closely for an efficient request processing.

The main objective of this paper is a detailed study of the impact of granularity on the responsiveness and efficiency of a reservation system. In section 2 different resource management approaches to support advance reservation are discussed and the concept of granularity is introduced. Section 3 gives a mathematical analysis of the impact of granularity on reservation systems while section 4 validates the results by simulation. Section 5 concludes the analysis.

2 Resource Management Approaches

There are various approaches to resource management for reserving resources in the scope of Grid computing and networking. This section introduces different approaches which are further analyzed in the subsequent sections. Without loss of generality, we will focus on temporal aspects in the scope of allocating network resources.

2.1 Reservation-Based Approach

A basic reservation-based approach uses the set of already accepted reservations for admission control of incoming reservation requests. All accepted requests overlapping the requested time interval are identified [6], which allows to determine if enough resources are available to fulfill the request. This concept has a low memory consumption as it only stores accepted requests which are needed for connection establishment anyway. However, if one reservation request is handled after the other, up to $(i-1)$ accepted requests have to be considered in the worst case when the i^{th} request is handled. This means that the time complexity to determine the available resources for n subsequent requests is $\sum_{i=1}^{n}(i-1) \in \Omega(n^2)$. As a consequence, the approach is favorable if the number of requests is low. To cope with the complexity, a timeslot-based approach is introduced that maintains aggregated resource consumption information.

2.2 Timeslot-based Approach

The *timeslot-based* resource management uses a global timeline divided into a set of *timeslots*. Each slot represents a period of time and holds information about the accumulated resource consumption for every link as depicted in figure 1. The timeslot-based approach can be used to determine available resources independent of the number of accepted reservations. Once a new request is processed, the reservation system has to determine the *active timeslots*, a subset of all timeslots that cover the requested period. The maximal resource utilization of all active timeslots along with the resource capacity determines the resources available to new requests.

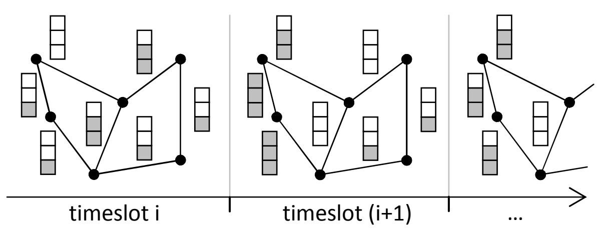

Figure 1. Timeslot-based resource utilization information.

The timeslot-based management of allocated network resources is an established [7, 1, 2] and efficient way to manage utilization information. The approach is already used in different environments for advance reservations, e.g. denoted as *timeslot table* in GARA [8] – a component of the Globus Toolkit which performs the resource management.

2.3 Static and Dynamic Timeslots

A model with n_{ba} timeslots with a constant length c_s is denoted as *static timeslots* model. Here, the boundaries b_{i-1}, b_i of a timeslot are indirectly defined by its index $i, 1 \leq i \leq n_{ba}$ and the *slot length* c_s. Using this model the number of timeslots which are managed by the reservation system is completely independent of the number of accepted reservations; it only depends on the length of the book-ahead interval c_{ba} given by $c_{ba} = c_s \cdot n_{ba}$. The book-ahead interval is a sliding window in which new requests are accepted. It defines the time horizon of the reservation system. While the static approach is very easy to implement, it is inefficient if only a small number of reservations is managed by the reservation system.

Alternatively, timeslots of dynamic length can be used with a length of an integer multiple of c_s where c_s is the smallest slot length the timeslots can be partitioned into. For every accepted reservation, the existing timeslots are divided at the starting time and the ending time, unless the timeline is not already divided at these points. Every time a new reservation is accepted, no more than two new timeslots have to be created. Even if the number of timeslots depends on the number of accepted reservations, it is limited by n_{ba}.

2.4 The Granularity of Resource Management

The mechanism of using timeslots for a faster admission control is enmeshed with the *internal granularity* or *granularity* of the reservation system. This granularity is defined as the smallest Δt which is distinguishable by the reservation system. In addition to this internal granularity an *external granularity* g can be defined that specifies the smallest Δt which is distinguishable by a service taker.

Using the reservation- or dynamic timeslot-based resource management approach the system can theoretically operate with an *infinitesimal granularity*. In this case the starting and ending time of requests are not altered. In case of dynamic timeslots a boundary of a timeslot is set at any point in time a reservation begins or ends. However, the maximum number of timeslots and therefore the processing time and memory consumption depends on the number of accepted reservations. A newly accepted reservation may result in up to two new timeslots. So, in the worst case, the reservation system has to manage $(2n+1)$ timeslots for n accepted reservations.

This dependency can easily be avoided by introducing a *non-infinitesimal granularity* for the timeslots. Only a fixed set of usually equidistant points in time within the book-ahead interval is allowed as timeslot boundaries.

2.5 Adjusting Reservations to Granularity

In general, the usage of a non-infinitesimal granularity makes it necessary to map arbitrary points in time specified in a request on a discrete set accepted by the reservation system. The approach described in [9] rounds the starting and ending time of a reservation to the nearest slot boundaries. Starting and ending times within the interval $[b_i - \frac{1}{2}c_s;\ b_i + \frac{1}{2}c_s[$ are rounded to the slot boundary b_i. A major advantage of this approach is that the average duration of the reservations does not grow compared to the original one, if the starting and ending times are uniformly distributed in the neighborhood of the slot boundaries. This ensures that the average amount of resources allocated for a set of reservations does not increase by the mapping. However, this approach is inappropriate for reliable resource reservation, especially in the area of Grid computing. Changing the reserved time interval to one where the requested interval is not fully included requires a negotiation process. A Grid meta-scheduler [3] takes into account the reservation periods of multiple resources that are co-allocated with the network, e.g. it has to enforce realtime requirements of resources like sensors and guarantee deadlines for data transfers. All starting and ending times of co-allocated reservations need to be aligned.

This reflection leads to another way of mapping arbitrary reservations to a certain granularity. Instead of rounding off requested periods, the reservation system maps reservation requests in such a way that the requested period is entirely included in the assigned one. This can be done by rounding down the starting and rounding up the ending time to the next slot boundary. The main advantage of this approach is that service takers need not be notified and the co-allocation mechanism is not constricted. Furthermore, there exists no use-case in which this approach is more inefficient than extending the required period by $2c_s$.

In the following, the original requested starting and ending times are denoted as t_{start} and t_{end}. The starting and ending times of the assigned period after the mapping are denoted as $\hat{t}_{start}$ and $\hat{t}_{end}$. Consequently, resources for a reservation res are allocated within the interval $[\hat{t}^{res}_{start}; \hat{t}^{res}_{end}[$.

A detailed study of the impact of granularity on the responsiveness and the efficiency of the reservation system is presented in the next sections.

3 Identifying an Adequate Granularity

The *granularity* defined as c_s for a timeslot-based resource management has a direct influence on the efficiency of the reservation system. On the one hand, choosing a coarser granularity results in a higher probability that the assigned periods of two accepted reservations overlap while the requested periods might not. This leads to an increased probability of rejecting new requests. On the other hand, a finer granularity results in a higher number of timeslots having to be managed by the reservation system, coming along with a slower handling of requests.

In section 3.1 the overlapping probabilities for an infinitesimal and non-infinitesimal granularity are analyzed. These results are used in section 3.2 to identify the overhead caused by a non-infinitesimal granularity. Section 3.3 gives a quantitative analysis of the expected number of timeslots in a system using the dynamic timeslot model.

3.1 Overlapping Probability

In the following, the assigned ending time $\hat{t}_{end}$ of a reservation in the book-ahead interval and its duration $\hat{t}_{end} - \max\{\hat{t}_{start}; \hat{t}_0 = \lfloor \frac{t_0}{c_s} \rfloor c_s\} =: d$ are considered. Let t_0 denote the point in time at which the reservation request is received. The assigned starting time is indirectly given by $\hat{t}_{start} = \hat{t}_{end} - d$ which is obviously greater than or equal to the point in time the request is received. For an infinitesimal granularity and requested starting and ending times t_{start} and t_{end} it holds that $\hat{t}_{start} = \max\{t_{start}, t_0\}$, $\hat{t}_{end} = t_{end}$ and $d = t_{end} - t_{start}$.

In the following analysis, the probability of an overlap between a new request req and an accepted reservation res is determined by means of the position of the ending time of the new request. The interval in which the placement of the ending time of the request would cause an actual overlap between the request req and the reservation res is denoted as *overlap-interval* c_{ol} of res regarding req. The overlap-interval can be divided into two parts. The first one is given by $]\hat{t}^{res}_{start}; \hat{t}^{res}_{end}[$. An ending time of the request within this interval leads to an overlapping between its rear part and the front part of the reservation. It is denoted as *reservation-caused* part c^{res}_{ol}, while the other part is named *request-caused* part c^{req}_{ol} of the overlap-interval. It is defined through $[\hat{t}^{res}_{end}; \hat{t}^{res}_{end} + d^{req}[$. If the ending time of the request is located in this part, the overlap appears between the rear part of the reservation and the front part of the request. Both parts of the overlap-interval are depicted in figure 2.

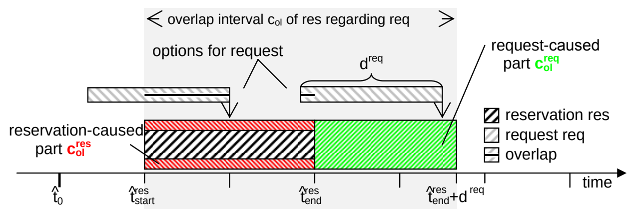

Figure 2. Analysis of the overlap-interval.

Assuming an independent and uniform distribution of the ending times of the reservations, the probability of an overlap between a single reservation and a new request is determined by the ratio of the reservation's overlap-interval regarding the request to the book-ahead interval. To determine the expectation for a set of n reservations, it is sufficient to consider only reservations and requests with the average duration of $\bar{d} = \frac{1}{n} \sum_{i=1}^{n} d^{res_i}$, where the durations can be arbitrarily distributed. Note that only reservations which are accepted by the reservation system are considered, so the ending time has to be in the book-ahead interval. Furthermore, it is assumed that the average duration $\bar{d}$ of the reservations is an integer multiple of the slot length c_s while c_s is an integer multiple of the external granularity g.

3.1.1 Infinitesimal Granularity - A lower Boundary

The lower boundary of the expected number of reservations a new request overlaps can be determined by analyzing a reservation system with an infinitesimal granularity. In this case, the accepted reservations are not adjusted, resulting in no unnecessary overlapping.

For the following analysis, two further helpful identifiers are introduced: The first one denotes the part of the overlap-interval in which the ending time of a new request can fall. It is named *limited overlap-interval*, written as $c_{lim.o}$. Analogously, the part of the book-ahead interval in which the ending time of a request can fall is denoted as *limited book-ahead interval*.

The probability of an overlap between a single reservation and a new request is given by the ratio between the length of the limited overlap-interval and the length of the limited book-ahead interval, as a uniform distribution of the ending times is assumed. The former one is determined by the sum of the lengths of the reservation-caused part and the request-caused part of the limited overlap-interval. The limited book-ahead interval can be computed by $c_{ba} - \bar{d}$, as the earliest possible starting time is the point in time the request is received. The assumption that all ending times are independent allows for the use of a binomial distribution to compute the expectation that a new request overlaps n accepted reservations. So, only the lengths of the reservation-caused and the request-caused part of the limited overlap-interval have to be determined.

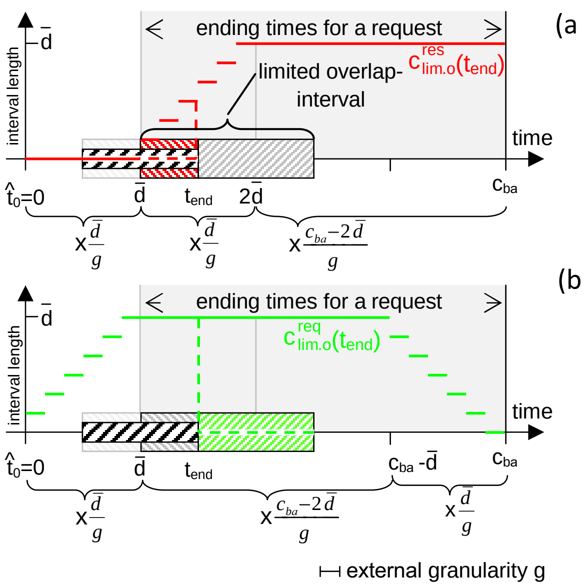

Figure 3. The length of the reservation-caused (red) and request-caused (green) part of the limited overlap-interval as a function of the position of its ending time t_{end}.

To simplify the following analysis, without loss of generality, $\hat{t}_0$ is set to $\hat{t}_0 = 0$. First, the possible ending times for a new request with an average duration $\bar{d}$ are considered. The definition of d ensures that the ending time has to be in $[\bar{d}; c_{ba}]$, justified by the same argument which is used to determine the length of the limited book-ahead interval. With this information, the average lengths of the reservation-caused and the request-caused part of the limited overlap-interval (depending on the position of its ending time) can be examined.

First, the length of the reservation-caused part is determined. For this, the book-ahead interval can be divided into three sections in which the ending time can fall (figure 3.a). Obviously, if the ending time t_{end} of the reservation falls in $[0; \bar{d}]$, there is no overlap with $[\bar{d}; c_{ba}]$. If $t_{end} \in]\bar{d}; 2\bar{d}]$, the length is given by $t_{end} - \bar{d}$, otherwise the length is determined by $\bar{d}$.

The possible ending times for a reservation res are limited to the discrete set of points in time which covers $(\frac{1}{g}c_{ba})$ elements given by $\hat{t}_{end} = ig \mid 1 \leq i \leq \frac{1}{g}c_{ba}$. This is caused by the external granularity g. So, the average length of the reservation-caused part of the limited overlap-interval for a single reservation can be calculated by

$$\begin{aligned}\bar{c}^{\,res}_{lim.o} &= \frac{1}{\frac{1}{g}c_{ba}}\left(\sum_{i=1}^{\frac{\bar d}{g}} 0 + \sum_{i=\frac{\bar d}{g}+1}^{2\frac{\bar d}{g}} (ig - \bar{d}) + \sum_{i=2\frac{\bar d}{g}+1}^{\frac{c_{ba}}{g}} \bar{d}\right) \\ &= \frac{g}{c_{ba}}\left(\sum_{i=1}^{\frac{\bar d}{g}} 0 + g\sum_{i=1}^{\frac{\bar d}{g}} i + \bar{d}\sum_{i=1}^{\frac{c_{ba}-2\bar d}{g}} 1\right) \\ &= \bar{d} - \frac{\bar{d}}{c_{ba}}\left(\frac{3}{2}\bar{d} - \frac{g}{2}\right).\end{aligned}$$

Now, the length of the request-caused part of the limited overlap-interval can be determined analogously to the reservation-caused part. The book-ahead interval can be divided into three sections, as well (figure 3.b).

If t_{end} of the reservation lies in $[0; \bar{d}]$, the length of the request-caused part is t_{end}. Within $]\bar{d}; c_{ba} - \bar{d}]$, the length is $\bar{d}$, while the length for an ending time in $]c_{ba} - \bar{d}; c_{ba}]$ is determined by $c_{ba} - t_{end}$. So, the average length of the request-caused part of the limited overlap-interval can be calculated by

$$\begin{aligned}\bar{c}^{\,req}_{lim.o} &= \frac{1}{\frac{1}{g}c_{ba}}\left(\sum_{i=1}^{\frac{\bar d}{g}} ig + \sum_{i=\frac{\bar d}{g}+1}^{\frac{c_{ba}-\bar d}{g}} \bar{d} + \sum_{i=\frac{c_{ba}-\bar d}{g}+1}^{\frac{c_{ba}}{g}} (c_{ba} - ig)\right) \\ &= \frac{g}{c_{ba}}\left(g\sum_{i=1}^{\frac{\bar d}{g}} i + \bar{d}\sum_{i=1}^{\frac{c_{ba}-2\bar d}{g}} 1 + g\sum_{i=1}^{\frac{\bar d}{g}}\left(\frac{\bar d}{g} - i\right)\right) \\ &= \bar{d} - \frac{\bar{d}}{c_{ba}}\bar{d}.\end{aligned}$$

The expectation for the entire length of the limited overlap-interval is computed by

$$\begin{aligned}\bar{c}_{\,lim.o} &= \bar{c}^{\,res}_{lim.o} + \bar{c}^{\,req}_{lim.o} \\ &= 2\bar{d} - \frac{\bar{d}}{c_{ba}}\left(\frac{5}{2}\bar{d} - \frac{g}{2}\right).\end{aligned}$$

The probability of an average request to overlap an average reservation in a system with arbitrary granularity is determined by the ratio between the expectation for the length of the limited overlap-interval and the length of the limited book-ahead interval. As mentioned before, the latter interval is equal to $[\bar{d}; c_{ba}]$. So, the probability of an overlap between an average reservation and an average request can be computed by

$$p_{ol} = \frac{\bar{c}_{lim.o}}{(c_{ba} - \bar{d})} = \frac{\bar{d}}{c_{ba}\left(c_{ba} - \bar{d}\right)}\left(2c_{ba} - \frac{5}{2}\bar{d} + \frac{g}{2}\right).$$

3.1.2 Non-infinitesimal Granularity

The studies made for a reservation system with infinitesimal granularity are similar to those which are made for a system with non-infinitesimal granularity. The main difference between them is the fact that the difference by which the requested starting and ending times are rounded to the assigned times has to be considered additionally. This difference will lead to a larger overlap-interval compared to that of a system with infinitesimal granularity, which again results in a higher probability that an overlap will occur. Obviously, the difference between this probability and p_{ol} depends on the grain size of the book-ahead interval, defined by the number of slots for a fixed book-ahead interval.

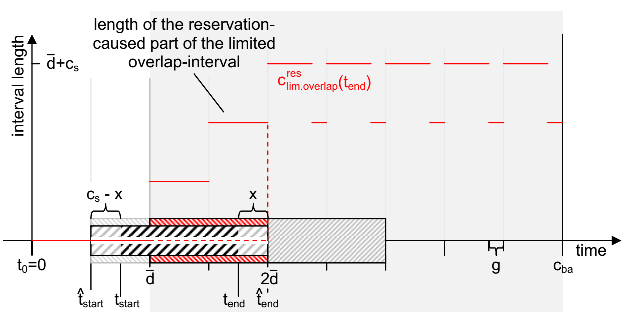

Figure 4. Length of the reservation-caused part of the limited overlap-interval for slotted granularity.

Analogous to a system with infinitesimal granularity, the book-ahead interval can be divided into three sections into which the ending time of a reservation can fall. If t_{end} of the reservation lies in $[0;\,\bar{d}]$, the assigned ending time $\hat{t}_{end}$ is also in $[0;\,\bar{d}]$, so the reservation-caused part of the overlap-interval does not contribute to the limited overlap-interval. An ending time t_{end} falling in $]\bar{d};\,2\bar{d}]$ results in a reservation-caused part of the limited overlap-interval with a length of $\lceil\frac{t_{end}-\bar{d}}{c_s}\rceil c_s$ as shown in figure 4, as $\hat{t}_{end}$ was rounded up to the upper boundary of the slot that contains t_{end}.

The assumption that $\bar{d}$ is an integer multiple of c_s leads to two different sub-cases that can occur, if $t_{end} \in\,]2\bar{d};\,c_{ba}]$. If t_{end} falls on a slot boundary, the starting time t_{start} also falls on a slot boundary, which means that the duration of the assigned period remains unchanged compared to the duration $\bar{d}$ of the requested period. Otherwise, the length is determined by $\bar{d} + c_s$, as t_{end} is rounded up about x and t_{start} is rounded down about $c_s - x$ as depicted in figure 4. Due to the uniform distribution of the ending times, the ratio of occurrence of these two sub-cases is $1 : \frac{c_s}{g} - 1$. So, the average length of the reservation-caused part of the limited overlap-interval using a slotted granularity of c_s can be determined by

$$\begin{aligned}\bar{c}^{\,res}_{lim.o} &= \frac{1}{\frac{1}{g}c_{ba}}\left(\sum_{i=1}^{\frac{\bar{d}}{c_s}} 0 + \sum_{i=\frac{\bar{d}}{c_s}+1}^{\frac{2\bar{d}}{c_s}} \frac{c_s}{g}(ic_s - \bar{d}) + \right. \\ &\qquad \left. \sum_{i=\frac{2\bar{d}}{c_s}+1}^{\frac{c_{ba}}{c_s}} \left(\left(\frac{c_s}{g} - 1\right)(\bar{d} + c_s) + 1\bar{d}\right)\right) \\ &= \frac{g}{c_{ba}}\left(\sum_{i=1}^{\frac{\bar{d}}{c_s}} 0 + \frac{{c_s}^2}{g}\sum_{i=1}^{\frac{\bar{d}}{c_s}} i + \sum_{i=1}^{\frac{c_{ba}-2\bar{d}}{c_s}} \frac{c_s}{g}(\bar{d} + c_s) - c_s\right) \\ &= (\bar{d} + c_s) - \frac{\bar{d}}{c_{ba}}\left(\frac{3}{2}(\bar{d} + c_s) - g\frac{2\bar{d} - c_{ba}}{\bar{d}}\right).\end{aligned}$$

Extending the requested period of the reservation request to the next slot boundaries does not affect the length of the request-caused part of the overlap-interval. The ending time of a new average request req falls into the request-caused part of the overlap-interval only if $t^{req}_{end} \in\,]\hat{t}_{end};\,\hat{t}_{end} + \bar{d}[$. Otherwise, t^{req}_{end} is greater than $(\hat{t}_{end} + \bar{d})$, so t^{req}_{start} is also greater than $\hat{t}_{end}$. The fact that $\hat{t}_{end}$ is a slot boundary ensures that t^{req}_{start} is rounded down to a value equal or greater than $\hat{t}_{end}$; this cannot lead to an overlap with the reservation req. The length of the request-caused part of the limited overlap-interval is considered in the following.

If t_{end} falls in $]0;\,\bar{d}]$, the length of the request-caused part of the limited overlap-interval is $\lceil\frac{t_{end}}{c_s}\rceil c_s$. If $t_{end} \in\,]\bar{d};\,c_{ba} - \bar{d}]$, the length is given by $\bar{d}$, while an ending time in $]c_{ba} - \bar{d};\,c_{ba}]$ results in a length of $c_{ba} - \lceil\frac{t_{end}}{c_s}\rceil c_s$. Thus, the average length of the request-caused part of the limited overlap-interval in case of a non-infinitesimal granularity can be calculated by

$$\begin{aligned}\bar{c}^{\,req}_{lim.o} &= \frac{g}{c_{ba}}\frac{c_s}{g}\left(\sum_{i=1}^{\frac{\bar{d}}{c_s}} ic_s + \sum_{i=\frac{\bar{d}}{c_s}+1}^{\frac{c_{ba}-\bar{d}}{c_s}} \bar{d} + \sum_{i=\frac{c_{ba}-\bar{d}}{c_s}+1}^{\frac{c_{ba}}{c_s}} (c_{ba} - ic_s)\right) \\ &= \bar{d} - \frac{\bar{d}}{c_{ba}}\bar{d},\end{aligned}$$

which means that it remains unchanged compared to a system using an infinitesimal granularity. The entire length of the average limited overlap-interval using a non-infinitesimal granularity is computed by

$$\begin{aligned}\bar{c}_{lim.o} &= \bar{c}^{\,res}_{lim.o} + \bar{c}^{\,req}_{lim.o} \\ &= (2\bar{d} + c_s) - \frac{\bar{d}}{c_{ba}}\left(\frac{5}{2}\bar{d} + \frac{3}{2}c_s - g\frac{2\bar{d} - c_{ba}}{\bar{d}}\right).\end{aligned}$$

Also analogous to a system with infinitesimal granularity, the probability that a new average request overlaps an accepted average reservation is calculated by normalizing the average overlap-interval to the length of the interval, the ending time of the request can fall:

$$p_{ol}^{c_s} = \frac{\bar{c}_{lim.o}}{(c_{ba} - \bar{d})} = \frac{\bar{d}\Big((2+\frac{c_s}{d})c_{ba} - \frac{5}{2}\bar{d} - \frac{3}{2}c_s + g\frac{2\bar{d}-c_{ba}}{d}\Big)}{c_{ba}(c_{ba}-d)}.$$

3.2 Influence on the Reservation Overhead

For a system using non-infinitesimal granularity, the limit $\lim_{c_s \to g} p_{ol}^{c_s}$ tends to p_{ol}, which means that a smaller slot length c_s results in a lower expectation of the number of reservations a new request overlaps with. This means that a small value for the slot length c_s is preferable, as a wasting of resources is avoided.

The assumption that the reservations are distributed independently results in a binomial distribution for the probability that a new request overlaps k of n accepted reservations. The expectation of this distribution is given by $E(n,p) = np$, where p denotes the probability that the request overlaps a single reservation. The factor o_{ol} between the expectations using a non-infinitesimal and an infinitesimal granularity describes the overhead caused by the usage of timeslots. This factor is equal to

$$o_{ol} = \frac{E(n, p_{ol}^{c_s})}{E(n, p_{ol})} = \frac{p_{ol}^{c_s}}{p_{ol}} \geq 1$$

and depends on the length of the book-ahead interval c_{ba}, the average reservation length $\bar{d}$, the slot length c_s, and the external granularity g. A value close to one means that the overhead is low, while a higher value means a larger overhead. The consideration of two different scenarios will illustrate the overhead depending on the granularity, specified by the slot length c_s.

Assuming an external granularity with a grain size of one minute, the first scenario (SI) has a book-ahead interval c_{ba} with a length of 30 days, while the second scenario (SII) has one of half a year. In both cases, an average reservation length $\bar{d}$ of $1\frac{1}{2}$ hours is assumed, while slot lengths c_s between 1 hour and five minutes are considered.

	considered value	$\times g$ $[g = 1min]$
$\bar{d}$	$1\frac{1}{2}h$	90
c_{ba}	SI: $30d$ or SII: $\frac{1}{2}y$	SI: 43200 or SII: 259200
c_s	$1h$ down to $5min$	60 down to 5

Table 1. Values assumed for SI and SII.

An overview of the scenarios is shown in table 1. The values chosen for the studies are based on statistics for grid-jobs from the EGEE project [10].

The left diagram of figure 5 shows the overhead o_{ol} for both scenarios depending on the slot length c_s for different external granularities g. The first important observation is that the length of the book-ahead interval which differs by the factor 6 between both scenarios has nearly no influence on the overhead.

The overhead which is caused by the usage of a slotted granularity depends on the difference between the external granularity (given by g) and the granularity which is caused by the reservation system (given by c_s). In the scenarios considered here, a difference of 60 minutes leads to an expectation of the number of reservations overlapped by a new request which is 33% higher compared to a system using an infinitesimal granularity. Choosing a slot length of 15 minutes results in an expectation that is increased by about 7.5%, while a slot length of five minutes leads to an increase of about two percent. This result is in line with the statement made in [11] which says that a slot length between five and 15 minutes is convenient.

The overhead o_{ol} has a linear dependency on the granularity c_s

$$o_{ol}(c_s) = \frac{\frac{c_{ba}}{d} - \frac{3}{2}}{2c_{ba} - \frac{5}{2}\bar{d} + \frac{1}{2}g}c_s + \frac{2c_{ba} - \frac{5}{2}d + g\frac{2\bar{d}-c_{ba}}{d}}{2c_{ba} - \frac{5}{2}\bar{d} + \frac{1}{2}g}, \quad (1)$$

which can easily be gained by solving $o_{ol} = p_{ol}^{c_s}/p_{ol}$. An easy and good approximation is given if g approaches zero, if the external granularity g is sufficiently small compared to the slot length used by the reservation system. In this case, $o_{ol}(c_s)$ can be approximated by

$$o_{ol}(c_s) = \frac{\frac{c_{ba}}{d} - \frac{3}{2}}{2c_{ba} - \frac{5}{2}\bar{d}}c_s + 1. \quad (2)$$

The right-hand diagram of figure 5 shows the slope of the function defined by equation 2 depending on c_{ba} and $\bar{d}$. The observation made in the two use-case scenarios that the length of the book-ahead interval has nearly no influence on o_{ol} can also be observed here: The slope for a fixed average reservation length is almost constant. Both of the two use-case scenarios are plotted as blue and green dots in the diagram. By contrast, the influence of $\bar{d}$ on the slope, especially for small values of $\bar{d}$, is much greater. So, if the reservation system has to handle shorter reservations on average, a shorter slot length c_s should be chosen, too. For trends and cyclic variations of $\bar{d}$, the slot length could be adapted dynamically. A detailed discussion of dynamic slot lengths can be found in section 3.4.

3.3 Influence on the Number of Timeslots

The terms and assumptions made in section 3.1 are also valid in this section. This means that the ending times of the reservations are uniformly distributed while the durations of the reservations can be arbitrarily distributed. Furthermore, it is assumed that all reservations are independent of each other. The lower boundary of the timeslot S_i is denoted as b_{i-1}, the upper one as b_i. The boundaries of the slots which are defined by the chosen granularity are denoted as $s_j \mid 0 \leq j \leq \lceil \frac{c_{ba}}{c_s} \rceil$. The granularity and thus the minimal slot length c_s is assumed to be an integer multiple of the external granularity g. The probability that the

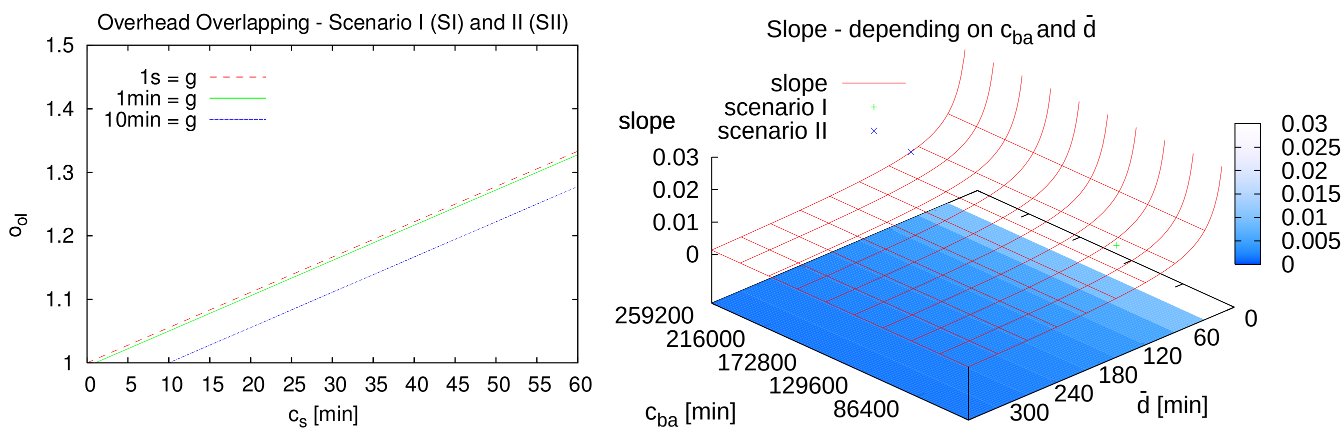

Figure 5. The overhead o_{ol} depending on the slot length (left) and the book-ahead interval (right).

assigned period of an accepted reservation res has the duration d is denoted as $p_{res}(d)$.

The following analysis considers the expectation of the number of timeslot boundaries of the book-ahead interval when using dynamic timeslots. If at a current point in time t_0 no reservation is managed by the reservation system, only a single timeslot $[\lfloor \frac{t_0}{c_s} \rfloor c_s ; \lceil \frac{t_0}{c_s} \rceil c_s + c_{ba}[$ has to be managed. Generally, every inner slot boundary at which at least one reservation begins or ends leads to different utilized resources during the neighboring slots. So, every inner slot boundary inside the book-ahead interval which fulfills this condition is a timeslot boundary as well. A set of n inner timeslot boundaries leads to $n+1$ timeslots that have to be managed by the reservation system. Thus, only the number of inner slot boundaries, at which at least one reservation starts or ends, has to be determined. This is done in the following.

First, a single slot boundary s_i and a single reservation res is considered. Using the assumption of a uniform distribution of the ending times, the probability that the assigned ending time $\hat{t}_{end}$ of the reservation res falls on s_i is determined by

$$p_i^{end} = \frac{c_s}{c_{ba}}.$$

The assigned starting time $\hat{t}_{start}$ of a reservation falls on s_i, if its assigned ending time falls on $s_{i+j} \mid 1 \leq j \leq \frac{c_{ba}}{c_s} - i$ and the duration of its assigned period is equal to $d = jc_s$. So, the probability that the assigned starting time of the reservation res falls on the slot-boundary s_i can be computed by

$$\begin{aligned} p_i^{start} &= \frac{c_s}{c_{ba}} \sum_{j=i+1}^{\frac{c_{ba}}{c_s}} p_{res}((j-i)c_s) \\ &= \frac{c_s}{c_{ba}} \sum_{d=1}^{\frac{c_{ba}}{c_s}-i} p_{res}(dc_s). \end{aligned}$$

The probability p_i that either the assigned starting or the assigned ending time of the reservation falls on the slot boundary s_i is determined by

$$\begin{aligned} p_i &= p_i^{end} + p_i^{start} \\ &= \frac{c_s}{c_{ba}} \left(1 + \sum_{d=1}^{\frac{c_{ba}}{c_s}-i} p_{res}(dc_s) \right). \end{aligned}$$

If a set of n reservations with an ending time in the book-ahead interval is considered, the probability that at least one assigned starting or ending time falls on a slot boundary s_i is equal to 1 minus the probability that neither of them falls on s_i. The assumption that the reservations are independent of each other makes it possible to use the binomial distribution to determine this probability. It is given by

$$1 - \binom{n}{0} p_i^0 (1-p_i)^{n-0} \quad = \quad 1 - (1-p_i)^n.$$

This probability can be used to compute the expected value of the number of timeslots. The expectation is equal to 1 plus the expected number of inner timeslot boundaries. The expected number of inner timeslot boundaries is determined by the expectation of the number of inner slot boundaries where at least one reservation begins or ends. So, the expected number of dynamic timeslots n_s for a probability distribution function $p_{res}(l)$ of the length of the reservations can be computed by summing up 1 and the probability that at least one starting or ending time falls on the slot boundary s_i for all $1 \leq i \leq \frac{c_{ba}}{c_s} - 1$:

$$n_s(p_{res}(l)) = 1 + \sum_{i=1}^{\frac{c_{ba}}{c_s}-1} 1 - (1-p_i)^n$$

$$= 1 + \sum_{i=1}^{\frac{c_{ba}}{c_s}-1} 1 - \left(\frac{c_{ba} - c_s}{c_{ba}} - \frac{c_s}{c_{ba}} \sum_{d=1}^{\frac{c_{ba}}{c_s}-i} p_{res}(dc_s) \right)^n$$

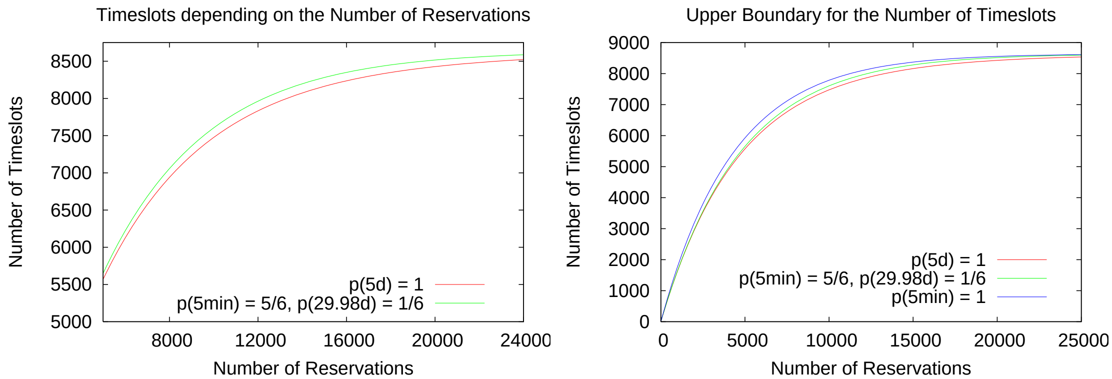

Figure 6. Number of timeslots for different slot sizes.

3.4 Dynamic Adaptation of Granularity

In the following analysis, the expected number of dynamic timeslots is estimated by using the upper boundary n_s.

$$n_s(p_{res}(c_s) = 1) = \frac{c_{ba}}{c_s} - \left(\frac{c_{ba}}{c_s} - 1\right)\left(\frac{c_{ba} - 2c_s}{c_{ba}}\right)^n$$

As shown in figure 5, a slot length c_s in the area of five up to 15 minutes leads to an acceptable overhead between two and 7.5% for the use-case scenarios described in table 1.

As expected, shorter slot lengths result in a greater expected number of dynamic timeslots that have to be managed by the reservation system. The number of timeslots that are handled by the reservation system differs insignificantly for slot lengths between five and 15 minutes, if the number of accepted reservations which are handled by the reservation system is sufficiently small. The difference becomes more significant for an increasing number of reservations, whereas the maximal number of timeslots is limited by $\frac{c_{ba}}{c_s}$, which is shown for both use-cases in figure 6.

The run of the curves leads to the idea of a dynamic adjustment of the granularity for the resource management. If only a small number of reservations have to be managed, a small value for the granularity is chosen, which minimizes unnecessary overlaps. As soon as the number of reservations exceeds a well defined threshold, the reservation system increases the granularity, which limits the number of timeslots which have to be managed from this point on. The reservations which are already accepted by the reservation system are not affected by the adaptation, so a remapping of accepted reservations is not necessary. The adaptation of the granularity can be performed several times for different thresholds. The precise values depend on the length of the book-ahead interval and the number of timeslots the reservation system is able to manage.

4 Simulative Validation

As described in section 3, the usage of a timeslot-based approach with a non-infinitesimal granularity for resource management leads to unnecessary overlaps between the accepted reservations. Equation 1 describes a linear relationship between the unnecessary overlaps and the chosen granularity. This section validates this relationship by means of simulation. The system efficiency is analyzed by means of the bandwidth blocking ratio as well as the system reponsiveness determined by the relative processing time for advance and malleable reservations.

4.1 Reservation System Efficiency

The bandwidth blocking ratio (BBR) [12, 4] is defined as

$$BBR := \frac{\sum_{res \in \bar{R}} a(res)}{\sum_{res \in R} a(res)},$$

where $a(res)$ denotes the amount of data that can be transmitted, $\bar{R}$ the set of rejected and R the set of requested reservations. In case of advance reservations, $a(res)$ is given by $a(res) = c_{bandwidth}(t_{end} - t_{start})$. Clearly, the BBR is only affected if the network load leads to an overbooking of resources. If the BBR increases due to a coarser granularity, the system efficiency decreases as more requests overlap and must therefore be rejected by the admission control. Furthermore, if resources are heavily overbooked the impact of other factors diminishes. Therefore, the network loads in the simulation have been chosen accordingly.

Figure 7 shows the BBR for different granularities and network loads. The BBR is depicted for three different network loads as a function of granularity. Here, the main issue is the slope of the fitted lines. In case of the low network load, the BBR increases slowly. The performance degradation becomes more significant in the medium and high network load scenarios. As can be seen, the fitted line in the medium load scenario has the largest slope. This corresponds to the preliminary note. Overall, the diagrams

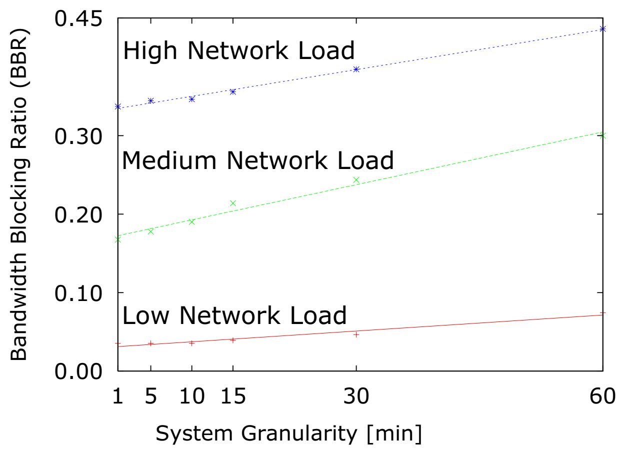

Figure 7. The Request and Bandwidth Blocking Ratio as a function of granularity for an Abilene-inspired topology.

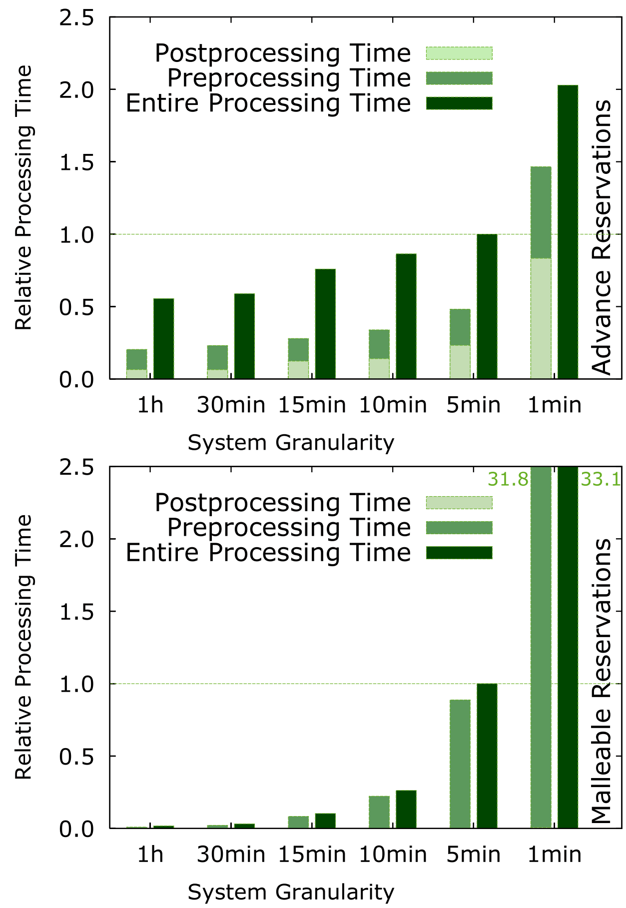

Figure 8. Normalized pre- and post- and entire processing times.

confirm the linear dependency between the number of unnecessary overlaps and the slot length, as derived in section 3.1.

4.2 System Responsiveness

Following section 3.3, a finer granularity used for the resource management results in an increased number of timeslots to be managed by the reservation system, i.e. memory consumption as well as processing time increase. This affects the responsiveness of a system that processes requests online. As the overall processing time depends on the implementation details, the relative processing time is analyzed.

In the pre-processing phase every timeslot covered by the requested time interval has to be considered. If a request is accepted, the allocated resources have to be reserved in the corresponding timeslots in the post-processing phase. Consequently, the processing time is analyzed for both phases. The diagrams of figure 8 depict the average pre- and post-processing times as well as the average entire processing times of advance and malleable reservation requests for different granularities. The processing times are scaled to the average entire processing time for a slot length of five minutes. The results are depicted for a path computation based on shortest distance paths. A widest/shortest strategy leads to similar results (cf. [13]). They confirm that coarser granularities increase the pre- and postprocessing time. Of course, the exact relationship depends on the implementation of the resource management.

The overall processing time of malleable reservations depicted in the lower diagram depends more on the slot length than the processing time of advance reservations depicted in the upper diagram. The set of reasonable configurations of a malleable reservations request depends on the number of timeslots. If a granularity in the range of 5 to 15 minutes and networks being not too large and dense are considered, the time which is needed for the average path computation phase is relatively small compared to the entire processing time.

5 Conclusions and Future Work

The main objective of the paper was a detailed study of the impact of granularity on the responsiveness and efficiency of a reservation system with special focus on network resources for Grid environments.

It has been shown that an admission control mechanism based solely on the inspection of already admitted reservations is only suited if the number of reservation requests is low. The existent concept of timeslot was introduced to store accumulated resource usage information and thus to reduce processing time. Additionally, we established the notion of granularity to decouple the number of timeslots needed from the number of accepted reservations. Here, granularity can be seen as a lower boundary of slot length. For static timeslots with a uniform size the granularity is equal to the slot size. For dynamic timeslots, the slot length is an integer multiple of the granularity. However, this concept needs a mechanism that adjusts reservations to granularity. This was done by rounding the request times to slot boundaries so that the requested interval is included in the reservation.

The resulting overhead was modeled mathematically in section 3 by comparing the overlapping probability of accepted requests in a system with an infinitesimal granularity to the probability in a system with a non-infinitesimal granularity. For systems with a dynamic timeslot mechanism the expected number of timeslots was analyzed.

The efficiency of the reservation system with a non-infinitesimal granularity was evaluated by simulation in section 4.1. Here, a linear dependency between the number of unnecessary overlaps and the slot length, as derived in section 3.1, was confirmed. The section closed with a study of the impact of granularity on the processing times of advance and malleable reservation requests. For advance reservation requests, the efficiency in terms of bandwidth blocking ratio is smaller when using a coarser granularity while the processing time of the requests is only slightly affected. However, for malleable reservations the overhead for a coarser granularity is negligible, as data transmission rates can be adapted to the assigned interval while the processing time benefits to a high extent. This is because the affected pre-processing phase in general needs to be repeated for multiple configurations of the malleable reservation. This causes a trade-off situation between the efficiency of the reservation system for advance reservations and the response time for malleable reservations.

The analysis of the overlapping probabilities in section 3.1 was done for a particular distribution of request ending times. Extending the analytical model to other distributions is one topic of further work. An additional topic is the design and evaluation of strategies to adapt the system granularity automatically.

Acknowledgements

The work in this paper is partially funded by the German Federal Ministry of Education and Research (BMBF) through the VIOLA project under grant 01AK605L and by the European Commission under the 6^{th} Framework Programme (6FP) through the IST-Phosphorus project under grant 034115.

References

[1] I. Foster, C. Kesselman, C. Lee, R. Lindell, K. Nahrstedt, and A. Roy. A Distributed Resource Management Architecture that Supports Advance Reservations and Co-Allocation. In *Proceedings of the International Workshop on Quality of Service*, 1999.

[2] R. Guerin and A. Orda. Networks with advance reservations: The routing perspective. In *19th Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM*, 2000.

[3] Christoph Barz, Thomas Eickermann, Markus Pilz, Oliver Wäldrich, Lidia Westphal, and Wolfgang Ziegler. Co-Allocating Compute and Network Resources - Bandwidth on Demand in the VIOLA Testbed. In *CoreGRID Symposium*, CoreGRID Series, Towards Next Generation Grids, pages 193–202. Springer, August 2007.

[4] Lars-Olof Burchard. *Advance Reservations of Bandwidth in Computer Networks*. PhD thesis, Berlin University of Technology, August 2004.

[5] Christoph Barz, Peter Martini, Markus Pilz, and Florian Purnhagen. Experiments on network services for the grid. In *LCN '07: Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN 2007)*, pages 45–54. IEEE Computer Society, 2007.

[6] A. Schill, F. Breiter, and S. Kuhn. Design and evaluation of an advance reservation protocol on top of RSVP. In *IFIP 4th International Conf. Broadband Communications*, pages 23– 40, 1998.

[7] Lars-Olof Burchard. Analysis of Data Structures for Admission Control of Advance Reservation Requests. *IEEE Transactions on Knowledge and Data Engineering*, 17(3):413–424, 2005.

[8] Alain Roy and Volker Sander. Gara: a uniform quality of service architecture. pages 377–394, 2004.

[9] Domenico Ferrari, Amit Gupta, and Giorgio Ventre. Distributed Advance Reservation of Real-Time Connections. In *Network and Operating System Support for Digital Audio and Video*, pages 16–27, 1995.

[10] The EGEE project. *Statistics for grid-computing jobs within the EGEE project*. `http://egee-jra2.web.cern.ch/egee-jra2/QoS/JobsMetrics/JobMetrics.htm` - Last Check: 2007-11-07.

[11] Alexander Schill, Sabine Kuhn, and Frank Breiter. Resource Reservation in Advance in Heterogeneous Networks with Partial ATM Infrastructures. In *INFOCOM (2)*, pages 611–618, 1997.

[12] Q. Ma and P. Steenkiste. Quality-of-Service Routing for Traffic with Performance Guarantees. In *IFIP Fifth International Workshopon Quality of Service*, pages 115–126, 1997.

[13] Q. Ma and P. Steenkiste. On path selection for traffic with bandwidth guarantees. In *IEEE International Conference on Network Protocols*, Atlanta, GA, USA, 1997.