SBM protocol for providing real-time QoS in Ethernet LANs

Anis Koubaa, Aref Jarraya, Ye-Qiong Song

SBM protocol for providing real-time QoS in Ethernet LANs

Anis KOUBAA Aref JARRAYA Ye-Qiong SONG

LORIA – UHP Nancy 1 - INPL - INRIA Lorraine
2, av. de la Forêt de Haye
54516 Vandoeuvre – France

Email : akoubaa@loria.fr; ajarraya@ensem.inpl-nancy.fr; song@loria.fr

Abstract *- This paper deals with the performance evaluation of LAN-Integrated Service protocol called SBM (Subnetwork Bandwidth Manager), a solution to handle QoS requirements over Local Area Networks. SBM is an RSVP-based protocol, which consists in electing a manager over a LAN segment to map RSVP-flows into an appropriate class of service and handles admission control and bandwidth reservation operations for such flows. To show how SBM is useful for guaranteeing requested quality of service for real-time admitted flows, we simulated the bandwidth reservation and message scheduling in an Ethernet switch for different input flows sharing a same output trunk link. DSBM[1] election has also been simulated in order to evaluate time for DSBM failure recovery over switched and shared LAN topology.*

1. Introduction

With the emergence of bandwidth-greedy and/or time-sensitive applications, the need of guaranteed QoS (Quality of Service) for these applications becomes of prime importance in the underlying networks. For this purpose, many approaches have been developed so far to provide real-time QoS guarantees for time-sensitive applications. In the Internet community, the two widespread approaches are *IntServ* and *DiffServ*. *IntServ* makes use of RSVP protocol with a bandwidth reservation in the routers and the related end-hosts along the path of IP packets to guarantee the end-to-end delay. For scalability reason, in *DiffServ*, an end user just needs to mark in the DS field of each of its packets the desired QoS class to signaling to routers its QoS demand (PHB: Per Hop Behaviour). Unlike *DiffServ*, which provides a per-class guarantee, *IntServ* provides a per-flow guarantee, which may arise the scalability problem in the Internet, but can be suitable to the industrial LAN context where the number of simultaneous flows to be handled should not be very important.

Recently, there is a willing to use Ethernet and its *de facto* upper layer protocols (TCP/IP and standards Internet applications) for factory communications. Although switched Ethernet can be configured to provide real-time QoS at the data link layer [4][5], for being able to take advantages of the upper layer Internet standards, protocols like *IntServ* or *DiffServ* must be deployed. The problem for deploying RSVP over Ethernet LANs is that RSVP stops at router level. To deal with this problem, an extension of IntServ-RSVP called SBM was defined [1] for LAN usage, which is a signaling protocol for RSVP-based admission control over IEEE 802-style networks. It supports the mapping of RSVP-enabled flows to Ethernet LANs providing the required QoS defined by RSVP [8] parameters.

SBM operates as follows:

- **DSBM Election Mechanism**: This procedure leads to designate a manager for a group of LAN-interconnected stations to handle the QoS requests on the managed segment. The elected member is called DSBM for Designated Subnet Bandwidth Manager. The principle is similar to the election of the Root Bridge in IEEE spanning tree protocol. For fault tolerance, the failure of the current DSBM leads to re-election of another DSBM
- **Bandwidth Reservation**: In that case, a single DSBM will manage the resources for those segments treating the collection of such segments as a single managed segment for the purpose of admission control. A station that wishes to send a guaranteed flow over the managed segment must firstly send a request to DSBM, which decides if a bandwidth reservation could be achieved.

SBM is defined to be used in both shared and switched LANs. Nowadays, switched LANs are more and more popular, for this reason we have chosen, within this work, to simulate the performance of SBM protocol over switched Ethernet LANs.

The contribution of this paper is to give a design and simulation framework for performance evaluation of LANs running SBM protocol. The model was developed using OPNET [11] Software. We built a switched network running SBM protocol and evaluated the

[1] DSBM : *Designated Subnetwork Bandwidth Manager* which performs Bandwidth reservation for incoming flows

performance of SBM in terms of message response-time and DSBM re-election time for failure recovery. For showing the importance of bandwidth reservation to provide a fair service and guaranteed QoS for time-sensitive applications, a comparative study between static scheduling (FIFO, SPQ *(Strict Priority Queueing)*) and per-flow scheduling (WFQ [10], PGPS [7]) is done.

2. QoS over IEEE802.3 LANs overview

In this section, we present some QoS features deployed over Local Area Networks. More details can be found in [1], [2] and [3].

2.1 QoS Legacy over Ethernet LANs

Initially, IEEE802.3 style networks do not provide any quality of service guarantees for any kind of traffic. All frames cross networks in best-effort fashion having or not real-time requirements. CSMA/CD protocol for shared half-duplex link does not provide deterministic medium access delay. This is not suitable for real-time sensitive applications and bandwidth-greedy flows.

Enhancements have been achieved by bridging solution, which reduces collision domain size by micro-segmenting the shared segment. Fully switched topologies can give deterministic access delay for the MAC layer as every node has its dedicated link but introduce additional latency upon frame reception and forwarding comparing to hub-repeaters.

The extended Ethernet format supporting *user_priority* tag, defined by *IEEE802.1p/Q* enables traffic classification for IEEE802 style networks. The 3-bit sized *user-priority* field enables differentiation between 8 traffic classes from 0 for lowest priority to 7 for highest priority flows. This field could be used by switches, according to IEEE802.1D standard.

2.2 Bandwidth reservation

Solutions given in previous paragraph by standard do not give any recommendation on how to deploy and handle traffic classes over LAN topology. However, there is much work built for QoS guarantee over IP networks (Internet) and mainly the Intserv and Diffser IETF working groups propositions [8][9]. Though, there is not known standards for bandwidth management over LAN until the SBM proposition given by ISSSL IETF working groups, which defined a framework for bandwidth reservation and QoS handling over IEEE802 networks [1][2][3].

The main idea of this proposal is to use the work carried out by IntServ-RSVP working group and defines the mapping of RSVP and Integrated services onto specific subnetwork technologies. This leads to designate an elected manager for a given segment to make bandwidth reservation for real-time applications. Segment may be *(a)* a shared Ethernet or Token ring bus resolving contention for media access using CSMA or token passing, *(b)* a half duplex link between two stations or switches or *(c)* one direction of a switch full-duplex link. Once the manager is elected by the DSBM election protocol for a given segment, it would obtain information on available resources such as bandwidth of the managed segment. All RSVP-Based reservation requests that transit would be processed by DSBM before forwarding it over the shared segment. The beauty of this protocol is that it supports RSVP protocol, and its implementation does not require many changes to RSVP request processing. A complete description for processing requests and implementation guidelines are detailed in [1].

3. Bandwidth reservation over Ethernet

Traditional Ethernet networks don't use any kind of bandwidth reservation. Traffic, that transit LAN domain, cannot make resource reservation. In best case, real-time traffic could have some better processing within switches using priority-based scheduling such as SPQ. Naturally, in that case, as switch cannot handle more than 8 parallel queues [IEEE802.1p], traffic will be scheduled in *aggregate*. For example, all video streams would be processed within a single queue. Problems exist if there are many streams to be handled within a same priority, especially when real-time constraints are hard. A solution that can be achieved using SBM protocol is to make bandwidth reservation and perform per-flow guarantee with WFQ scheduling algorithm. The manager processes the RSVP reservation request and accepts them whenever there are enough resources.

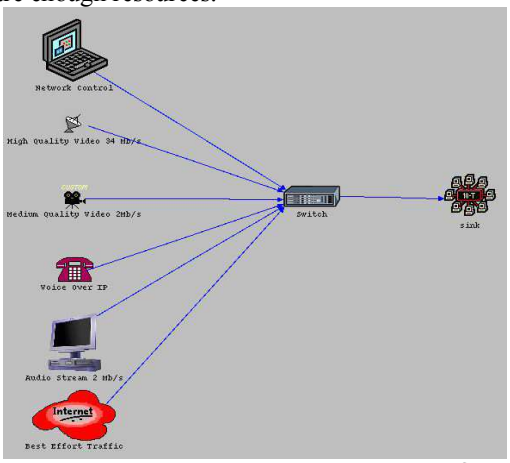

Figure 1. Typical Network Topology for Bandwidth Reservation

The following table shows flow characteristics for streams to be scheduled over the switch-manager.

Table 1. Flow Characteristics

Flow	Data Rate	Frame Length Distribution	Frame Inter-arrival Time (sec)
Network Control	20 kb	exp(1 Kb)	const(0,05)
HQ Video Stream	34 Mb	12 Kb	const(0,3336 * 10-3)
MQ Video Stream	2 Mb	6 Kb	const(2,86 * 10-3)
Voice	64 Kb	8 Kb	ON{exp(1), const(0,05)}/ OFF{exp(1,5)}
Audio	2 Mb	2 Kb	const(1,15 10^{-3})
Best Effort	1 Mb	Uniform(8 Kb, 12 Kb)	exp(0,01)

The total amount of traffic is about 41 Mb/s.

We have developed two scenarios to compare between these scheduling policies in terms of frame response time. The first one is highly loaded scenario with a total load of 0.9 and the second is an almost overloaded scenario with a load near to one (0.999). We further assume that the deadlines of the packets are equal to their periods (inter-arrival time). We mention that with SPQ scheduling, all video streams are handled within a same queue and same thing for audio streams. Here is the table for average and maximum response time for both scenarios.

Table 2. Flow Response Times with different scheduling algorithms

Flow	Deadline (ms)	Average Response Time (ms)						Maximum Response Time (ms)					
		Load 0.9			Load 0.999			Load 0.9			Load 0.999		
		FIFO	SPQ	WFQ	FIFO	SPQ	WFQ	FIFO	SPQ	WFQ	FIFO	SPQ	WFQ
Network Control	50	5,00	0,137	0,46	80	0,1635	9	170	1,357	11,251	387,0	1,611	348,40
HQ Video Stream	0,3336	5,00	0,31	0,294	80	0,4125	0,375	170	**2,569**	0,346	387,0	2,888	0,545
MQ Video Stream	2,86	5,00	0,31	0,325	80	0,4125	1,25	170	2,569	1,955	387,0	2,888	8,452
Voice	50	5,00	0,46	0,66	80	2,95	110	170	15,225	10,79	387,0	34,885	1213,7
Audio	1,15	5,00	0,46	0,221	80	2,95	0,41	170	**15,225**	1,112	387,0	34,885	2,982
Best Effort	10	5,00	175	0,6175	80	3800	17	170	**1899,1**	8,927	387,0	15344	222,40

It could be understood from results that WFQ gives better resource management. With resource reservation and per-flow scheduling, response time is better with WFQ than SPQ for Audio and Video streams. WFQ gives more fair scheduling behavior when the load is very high and can guarantee more narrow delays for lower priority flows without violation of higher priority ones. For example, from maximum response time results under a load of 0.9, WFQ meets the hard real time requirements of HQ and MQ video streams without violating the real time requirements of Network control traffic. Also In this case, all streams, even Best effort, meet their deadlines, which is not achieved with SPQ for the HQ Video Stream, Audio and Best Effort. This is explained by the advantage of per-flow scheduling and the ability to efficiently manage bandwidth resources.

An other fact, is when a congestion situation occurs, WFQ serve all flows with respect to their coefficient even that it does not provide all the requested bandwidth but does not make some flows to suffer from starvation as done by SPQ scheduler that serves only highest priority flows. This is explained by result for maximum response time with 0.999 of load.

Another advantage using SBM protocol is that it enables admission control so that a flow is admitted only if there are enough resources; else, it will be processed in best effort class. Moreover, a policy control could be used with SBM to prevent misbehaved sources from causing network congestion, which can affect the fairness of scheduling.

4. Modeling and performance evaluation of DSBM election algorithm

4.1 Motivation

Fault-tolerance is one of important issue for real-time application. In fact, when a manager is elected, it would manage resource reservation for real-time flows that need special processing to meet their real-time requirements. If the elected DSBM fails *(DSBMDeadIntervalTimer* fires), all the SBMs enter the

Elect state and start the election process. At the end of the election Interval, the elected DSBM sends out an *I_AM_DSBM* advertisement and the DSBM is then operational. This operation must not be too long to not disturb real-time behavior of current reserved flows. All reservations should be transferred to the new-elected DSBM. Next paragraph describes our SBM protocol model under OPNET simulator and results of time-to-recovery for switched and shared topology.

4.2 DSBM election Model

We implemented the DSBM election algorithm using OPNET simulation environment as described in [1].

The process model we developed will run on each Ethernet station, which communicates with each other with message. We define an SBM frame that will be used to store DSBM address and the priority of each SBM node. This information will be used by other stations to update their *LocalDSBMAddrInfo*, which represents in the end the information of the Best DSBM *(DSBM Address, DSBM Priority*). A complete description of model design and implementation is given in [6]. This process model treat two type of messages defined in [1]:

- ***DSBM_WILLING*** message sent by an SBM station to declare its candidacy to election process,
- ***I_AM_DSBM*** message sent by the DSBM itself for other SBM clients on the managed segment to declare itself as manager and to make sign of life every *RefreshIntervalTimer* period.

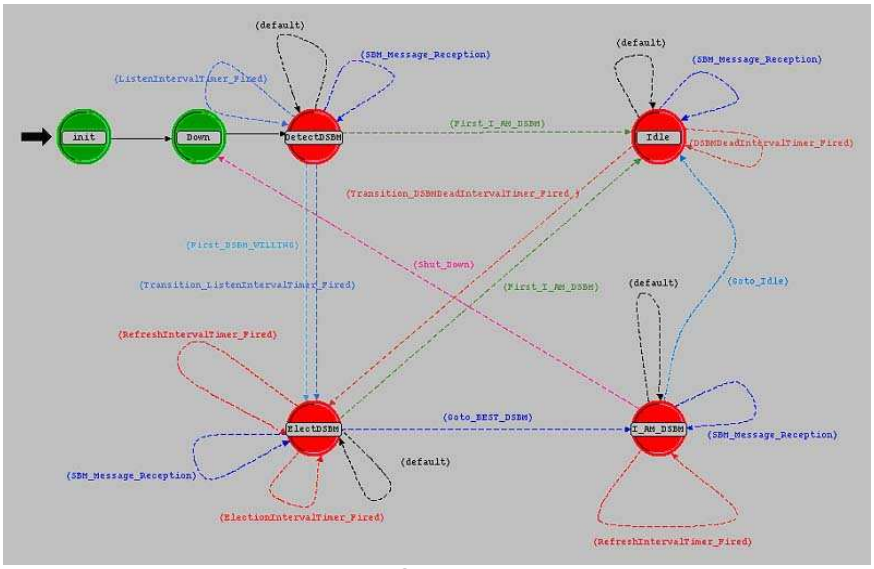

Figure 2. DSBM process model

4.3 Scenarios description and main results

We have run simulations for different Ethernet architectures, shared and switched.

In shared architecture, the elected DSBM would make resource reservation over managed segment, whereas in switched topology, DSBM could serve as a manager for the entire network when a centralized implementation of DSBM is used [3]. For both topologies, we have tried the election process with 2, 4, 8 and 16 SBM nodes to simulate the recovery-time, additional load resulting from sending *DSBM_WILLING* and *I_AM_DSBM* messages.

In all scenarios we have chosen these values for SBM Timers. There is recommendation for *ElectionIntervalTimer* in [1] to be set to 5 seconds. Suggestions are made for other timers but no recommended values.

Table 3. SBM Timers

Timer	Values
ListenIntervalTimer	3
DSBMDeadIntervalTimer	15
ElectionIntervalTimer	15
RefreshIntervalTimer	5

Simulations show the effect of increasing number of SBM candidates for DSBM election process. We collect through trace files, time needed for each station to make knowledge of the elected DSBM. In fact, this time is less that *ElectionIntervalTimer*. Once all stations know the elected DSBM, only the latter will continue to send *DSBM_WILLING* messages until *ElectionIntervalTimer* has to be fired.

The table below shows the time needed to discover the DSBM *i.e.* the instant from which only the DSBM sends *DSBM_WILLING* advertisements. Actually, all SBM stations know the DSBM, but declaration is done only after *ElectionIntervalTimer* expiration.

Table 4. Time-to-recovery (ms)

Topology	Shared	Switched
2 Nodes	0.059	0.024
4 Nodes	0.871	0.232
8 Nodes	3.981	1.612
16 Nodes	6.511	3.452

It is recommended that *ElectionIntervalTimer* is set at least to *DSBMDeadIntervalTimer i.e.* 15 seconds [3]. However, from our result this timer may be set to be fired just when *RefreshIntervalTimer* is fired and only the DSBM sends *DSBM_WILLING*. At this time, all DSBM clients know the new elected manager and should transfer their requests to the new DSBM. This makes quick recovery from Failure State and *RSVP PATH* has to be updated to insert the new DSBM instead of the failed one.

We make the following suggestion to quicken the recovery from a failure. At a start of election procedure, all stations generate their candidacy and send it through the LAN segment. This leads to a burst of *DSBM_WILLING* messages. In fact, at every *DSBM_WILLING* message reception, the station must send back a new *DSBM_WILLING* frame if it finds itself better that the received candidate. With the randomness of accessing media for a shared topology, the burst size may cause too much collision. We notice that the messages will be received in random order, which could cause more *DSBM_WILLING* generation. Then, once an SBM client sends its candidacy for the first time, there is no need to re-send its candidacy whenever it receives a *DSBM_WILLING* message.

To reduce this problem over a wide-scale topology, we suggest for an SBM station to not send a new *DSBM_WILLING* in every reception of candidacy message. After the first DSBM candidacy message, SBM station should send a new *DSBM_WILLING* message only after *RefreshIntervalTimer* fire or after a number of successive *DSBM_WILLING* message receptions. This would enhance election process by reducing collisions on shared segment and leads to faster recovery in case of DSBM failure.

For switched topology, there is no collision problem, but for large topology it may be useful to reduce the number of *DSBM_WILLING* messages, to not have overload of switch buffers.

5. Conclusion

In this paper, we have presented a performance evaluation framework for IEEE802.3 style network using SBM protocol as manager of LAN resources. We have evaluated the importance of bandwidth reservation and message scheduling algorithms over a Ethernet LAN to give better response time for real-time sources. The second part dealt with the DSBM election algorithm and proposed some enhancement to achieve faster recovery from a failure state of DSBM.

Based on these results, future work is to build a complete framework for integrated service over an Ethernet network. A possible continuation of this work is to build a QoS framework to support Diffserv request over LAN topology and this presents the advantage to not have a centralized manager for resources as in SBM approach.

6. References

[1] R. Yavatkar, D. Hoffman, Y. Bernet, F. Baker, M. Speer, *SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks* RFC 2814, May 2000

[2] M. Seaman, A. Smith, E. Crawley, J. Wroclawski " *Integrated Service Mappings on IEEE 802 Networks*" May 2000

[3] A. Ghanwani, W. Pace, V. Srinivasan, A. Smith, M. Seaman "*A Framework for Integrated Services Over Shared and Switched IEEE 802 LAN Technologies*" May 2000

[4] Song, Yeqiong, "*Time Constrained Communication Over Switched Ethernet*", 4th IFAC International Conference on Fieldbus Systems and their Applications - FeT'2001. (Nancy, France).

[5] Koubaa, Anis and Song, YeQiong, "*Evaluation de performance d'Ethernet commuté*", In 10th Conference on Real Time and Embedded Systems. (Paris, France). 2002. 9 p.

[6] Aref Jarraya, Yé-Qiong SONG, Anis KOUBAA « *Evaluation de performance des réseaux Ethernet pour les applications temps-réel* » LORIA-TRIO Internal Report Juin 2002

[7] K.Parekh, G.Gallager, *"A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case",* IEEE/ACM Transactions on Networking Vol1, NO3, June1993

[8] R. Braden,L. Zhang,S. Berson,S. Herzog, S. Jamin, *Resource ReSerVation Protocol (RSVP)* 8, September 1997

[9] An Architecture for Differentiated Services

[10] A.Demers, S.Keshavt, S.Shenker, "*Analysis and Simulation of Fair Queuing Algorithm*", Proceeding ACM SigComm 89, pp 1-12

[11] http://www.opnet.com