

# An Embedded Gateway Based on Real-Time Database

Zhiping Jia<sup>1</sup> and Xinxiao Qiao<sup>2</sup>

<sup>1</sup> School of Computer Science and Technology, Shandong University,  
Jinan, 250061, P.R. China  
zhipingj@sdu.edu.cn

<sup>2</sup> Department Of Computer Science and Technology, Shandong Institute Light Industry,  
Jinan, 250018, P.R. China  
qxxyyn@126.com

**Abstract.** Because of the limitation of the traditional gateway in the distributed monitoring system, the access of the protocol translation gateway will increase drastically with the massive usage of Ethernet. By using M/M/1 queuing theories, we present a framework of gateway based on the real-time database. The whole gateway is divided into two parts. One is linked to the monitoring subnet to collect the data with a polling method and deposit them in real-time database of the gateway. The other part is used for accepting the TCP connecting request, accessing the real-time database of the gateway. By analysis of the M/G/1 theory, it can be concluded that this framework can not only update the data more quickly in the distributed observing and controlling subnet, but also reduce the rejection of connections and loss of packet caused by the full queue buffer effectively when the number of TCP links become larger.

## 1 Introduction

Ethernet has already got extensive application in the distributed monitoring system because of its characters such as well-opening, extensive using and cheaper. The traditional monitoring systems mostly adopt Ethernet in the information layer, and generally use different field buses in the controlling layer and equipment layer. However, with the perfection of Ethernet and the development of embedded technology, the embedded network technology is a very fashionable topic for discussion. Ethernet has already permeated through the controlling layer and equipment layer of the monitoring system, which makes the access of the protocol translation gateway increase rapidly. And thus the limitation of the traditional gateway shows at this moment (Fig.1 shows the overall picture of network interconnection of a typical distributed monitoring system).

In this paper, we will reconstruct the protocol translation gateway for the distributed monitoring field, and present a framework which resolves the limitation existed in the traditional gateway when there are too many TCP connections. The framework can improve the network throughput and enhance the characterizes of real-time and security of the distributed monitoring system.

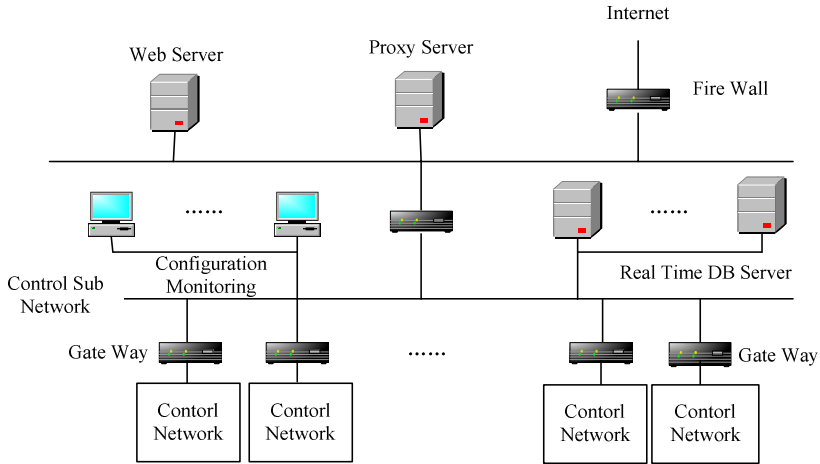


Fig. 1. The Connection Structure of Distributed Monitoring Network

## 2 Analysis of the Framework

The protocol translation gateway is the contractual media between Ethernet and the monitoring subnet. Ethernet protocol can carry on multi-connections and complete the data transmission of multi-connections at one time. However, what the monitoring subnet uses is the serial chain, which can carry on only one connection at a certain moment. And only one connection’s data transmission is allowed. When multi-connections are needed, the connection requests should queue and wait. Nowadays, most protocol translation gateway carries on data transmission in way of principal-and-subordinate ask-and-answer in the monitoring field. Thus, it will cause the block problem of connections when TCP connections are more. The larger the number of connections is, the slower the response time of the whole system is.

The framework of the traditional gateway is shown as Fig.2. When the serial chain is busy, the connect request can only wait. With the popularization of Ethernet in the industry automatic field, the access of the gateway will increase doubly, and then the response time of the whole system will increase, as a result, it will influence the real-time character of the whole system.

## 3 Model Analysis of the Framework

It can be known from the queuing theory [1] [2] [3] that the gateway model shown in Fig. 2 is a classics queue model -M/M/1[4]. In Fig.3, in the moment  $t+\Delta t$ , we set the number of customers who are accepted by system and queuing in the window is  $x$ . And we called it state  $X$ , whose probability is  $P_x(t+\Delta t)$ .

Here, in the block( $t,t+\Delta t$ ):

(1)The probability of one customer reaching is  $\lambda\Delta t+O(\Delta t^2)$ ,and the probability of no customer reaching is  $1-\lambda\Delta t+O(\Delta t^2)$ .

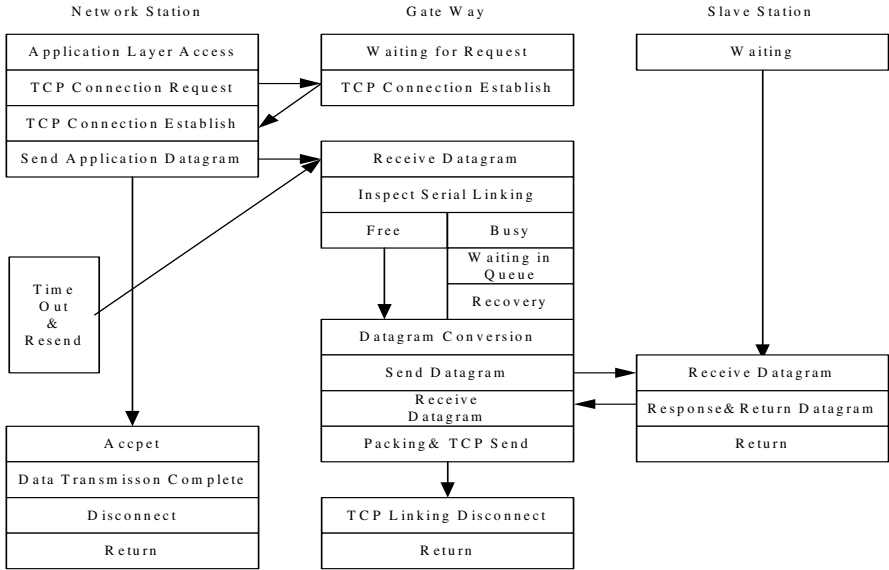


Fig. 2. Framework of the traditional gateway

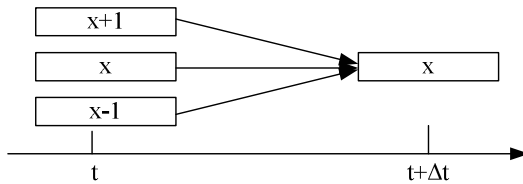


Fig. 3. The transition diagram of state X

(2)When there is customer receiving service, the probability of one customer leaving away is  $\mu\Delta t + O(\Delta t^2)$ ; and the probability of no customer leaving away is  $1 - \mu\Delta t + O(\Delta t^2)$ .

(3)The probability of more than one customer reaching or leaving away is  $O(\Delta t^2)$ , which can be neglected.

According to the whole probability theorem, the probability of state X is:

$$P_0(t+\Delta t) = P_0(t)(1-\lambda\Delta t) + P_1(t)(1-\lambda\Delta t)\mu\Delta t$$

$$\frac{dP_x(t)}{dt} = P_{x+1}(t) + P_{x-1}(t) - (\lambda + \mu)P_x(t)$$

We set  $\Delta t \rightarrow 0$ , and then have a differential equation:

$$\frac{P_x(t + \Delta t) - P_x(t)}{\Delta t} = P_{x+1}(t)\mu + P_{x-1}(t)\lambda - (\lambda + \mu)P_x(t) + \frac{O(\Delta t^2)}{\Delta t}$$

$$\frac{dP_0(t)}{dt} = P_0(t)\mu - \lambda P_1(t)$$

When the serving rate  $\mu$  is greater than the reaching rate  $\lambda$  and  $t \rightarrow \infty$  is stable, the state probability has nothing to do with time, namely, the differential item of differential equation is zero. Then the probability of state X is :

$$P_0=1-\rho$$

$$P_x=(1-\rho)\rho^x \quad x \geq 1$$

Here :  $\rho = \lambda / \mu$

Using the system state probability, we can have the relevant operation index of the system:

(1) The average number of customers in the system is:

$$L_s = \sum_{x=0}^{\infty} xP_x = \sum_{x=0}^{\infty} x(1-\rho)\rho^x = \frac{\rho}{1-\rho}$$

(2) The average number of customers waiting in the queue is:

$$L_q = \sum_{x=0}^{\infty} (x-1)P_x = \sum_{x=0}^{\infty} xP_x - \sum_{x=0}^{\infty} P_x = L_s - \rho = \frac{\rho^2}{1-\rho} = \frac{\rho\lambda}{\mu-\lambda}$$

(3) The expected value of the time customers waiting in the queue is:

$$W_q = L_q / \lambda = \frac{\rho}{\mu-\lambda}$$

(4) The expected value of the time customers waiting and being serviced is:

$$W_c = W_q + 1/\mu = \frac{1}{\mu-\lambda}$$

Because we use the serial circuit to accept the data of subnet communication unit, the communication speed of the serial circuit will cause greater influence on the serving rate.

The time of one service includes: the affirmation time of a connect request, the conversion time of datagram, the transmission delay of message reaching the subnet communication unit, the treatment time of the subnet communication unit, the transmission delay of message in the communication unit and the time of sending the TCP datagram.

If we think that the bandwidth of the upper network of the gateway is above 10M and the communication amount is general, then we can ignore the time of TCP connecting and transmitting delay. If we use the constant-length datagram to be transmitted, we set the datagram length is 20Byte. And we use an asynchronous communication way that has 8 data bits, one beginning bit and one ending bit. When the serial communication speed is between 600bps to 1 Mbps, we separately get the corresponding two-way transmission time, the approximate service rate, and the reaching rate, which are shown in Table 1.

It can be found out from Table 1, that with the serial chain speed increasing, the delay of datagram transmit in the serial chain is reduced, and when the speed is quickly enough, we can't ignore the time of TCP connecting and of transmit. Especially in a situation when the speed of the serial chain is slower and the TCP connections are more, the congestion of requests of the monitoring network will be caused, which slows down the speed of the data reading.

**Table 1.** Serial link transmit speed & delay and restriction of average arriving rate

Transmit Speed ( kbps)	0.6	1.2	4.8	9.6	19.2	56	1000
Datagram Length (bit)	240	240	240	240	240	240	240
Transmit Delay ( ms)	400	200	50	25	12.5	6.25	0.24
Restriction of Average Arriving Rate	0.75	1.5	6	12	24	70	250

The model analysis mentioned above is got under a condition that the queue buffer is infinite. But in the real situation, because of the hardware restraint, the queue quantity allowed is not too large. So when TCP connections are more, it will cause the problem of rejection of connections and loss of datagram, and influences the security of the whole system.

## 4 Improvement of Framework

The serial link has single access characteristic in traditional gateway, that is, only one connection can access at a certain moment, while others need to queue, which causes data access time to become longer. Thus it tends to cause requests congestion, connections timeout and datagram loss when the TCP connections are more, so we provide a new gateway framework based on real-time database. Profited from the swift development of embedded system, this framework can effectively operate and query data and process massive capacity data storage.

Designed a small-scale real-time database in gateway, the whole gateway can be divided into two parts. One part is connected with observing and controlling subnet, collecting data of subnet by a poll method and then storing them into the real-time database in gateway. The other part is a network workstation, which accesses the real-time database in gateway and retrieves data wanted directly, thus avoiding problems of queuing TCP connections caused by monopolization of serial link. The bridge between two parts is real-time database.

Under this way, we set the framework between the gateway and network access workstation to be Client/Server pattern, where the gateway is a database server and network access workstation is client. Gateway application is in state of waiting for connection and responsible to provide data needed, format transformation of datagram, address storage and send, etc. As to control subnet, gateway is thought to be the main site, which retrieves data by polling subsidiary site according to information of real-time database. Fig.4 shows this connection access process.

For example, we adopt MODBUS protocol[5]. Network workstation accepts TCP connection request, establishes connection, and is ready to receive datagram. After that, it parses datagram according to MODBUS/TCP protocol and gets MODBUS protocol frame. Then it parses MODBUS protocol frame, gets request command sent by PC, and queries real-time database according to request command, sends back the data encapsulated by MODBUS and MODBUS/TCP protocol to request client, and then

waits for request of TCP link removal. When it receives removal request, it will remove TCP link.

The other part retrieves commands from database by way of the access pattern of subnet controlling, sends MODBUS frame encapsulated according to MODBUS protocol to serial link, and then waits for return of data. After accepted data, it will parse MODBUS frame and write these data into real-time database, then retrieve the next command, and repeat the process above. As a result, all the data of real-time database update instantly.

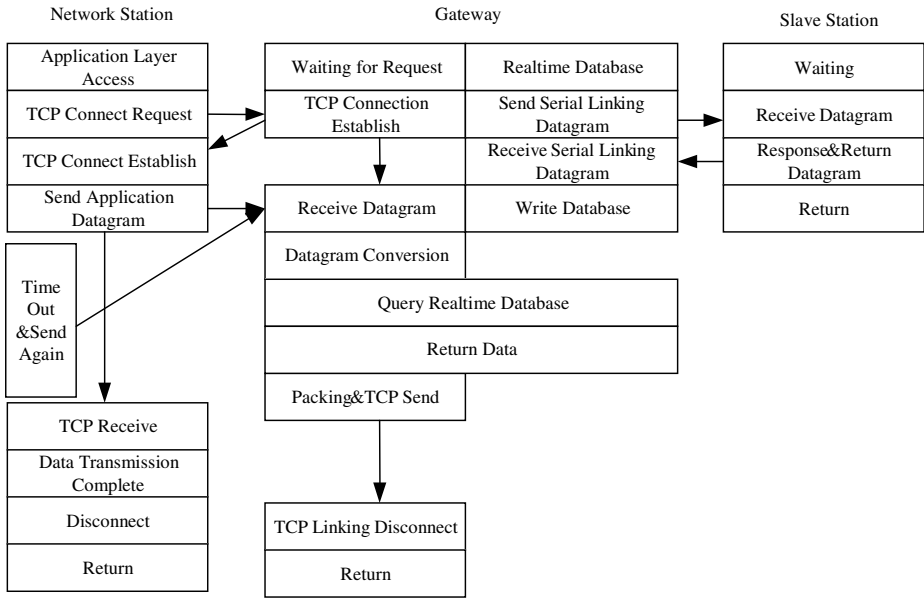


Fig. 4. The improved framework of gateway

## 5 Analysis of Gateway Framework Model with Database

### 5.1 Analysis of Control Subset Model

In the transfer mode of control subnet, the main communication source is gateway, which repeatedly reads data from subsidiary site and stores them in the database of gateway.

The above pattern is M/G/1 model [6][7]. We suppose the probability distributing of n is F(n), the probability mother function of F(n) is G(Z), the mean value is E(n), the variance is V(n), and the probability of accessing node n is P(n).

We suppose access time of every node is  $t_i$  ( $i=1,2,\dots,n$ ), so total service time is:

$$T_n = \sum_{i=0}^{\infty} xP_x$$

After all calculation, the last result is as follows:

$$\begin{aligned}
E(T_n) &= G'(1)H'(0) = E(n)E(t_i) \\
V(T_n) &= T_n''(0) - [T_k'(0)]^2 \\
&= G'(1)H''(0) + G''(1)[H'(0)]^2 - [G'(1)H''(0)]^2 \\
&= E(n)V(t_i) + [E(t_i)]^2V(n)
\end{aligned}$$

The average waiting time is as follows:

$$W_q = \frac{Lq}{\lambda} = \frac{\lambda^2 \rho^2 + (\lambda/\mu)^2}{2\lambda(1-\lambda/\mu)} = \frac{\lambda[V(T_n)]^2 + [\lambda E(n)E(t_i)]^2}{2\lambda(1-\lambda E(n)E(t_i))}$$

From above calculation, we can see that waiting time of client is relative to how much data needed to read in and the speed of serial link. The less data needed to be read in and the faster the serial link is, the swifter of data update speed is and the more real-time will be. In this solution, if the accessing chance of every data is equal, the existing time while the data be accessed is half of client waiting time.

As the operation of database avoiding the single access of serial link, it can improve the throughput of gateway and effectively avoid problem of request congestion.

## 5.2 Analysis of TCP Connection Response

The time of forth-and-back (RTT) of every TCP connection will be shorten because the real-time database is introduced. There is only one connection that can transfer data in serial link at one time in traditional gateway, so the TCP congestions do not exist. By introducing real-time database, it may generate congestion problem for the fast query time in database. As many papers have discussed TCP congestion problem, we will not go deep into these discuss. We just suppose the number of the TCP connection is not large enough to cause the TCP congestion. Thus as to the serial links, the TCP transfer time can be ignore.

As several TCP connections could be established, it will not have problems of TCP rejection of connection and loss of datagram, and this improves the security of system.

## 5.3 Analysis of the Whole Data

We suppose there are 10 control points, every control point averagely access 8 bytes and return 8 bytes (decided by serial link protocol), and the communication speed of control subnet is 19200baud. When the number of TCP connections change, we will get the response time of gateway (Shown in Table.2). While we set 10 TCP connections, we get response time of gateway (Shown in Table.3) when the connection speed of control subnet changes.

According to contrast, we can observe that the improved gateway framework is worse in real-time than traditional framework when there is only one TCP connection but more points in control subnet. But this framework will be better in real-time than traditional framework while the number of TCP connections increases and there are less points in control subnet. Especially, when the number of TCP connections increases, the improvement of real-time character will be in evidence. Indeed, real-time character of gateway is also related to the speed of control subnet.

**Table 2.** Number of TCP connection and response time

TCP Link Number	Response Time of Source Gateway (ms)	Response Time of New Gateway (ms)	
		Return Time of Gateway(ms)	Return Time of Control Subnet (ms)
1	12	<1	60
2	24	<1	60
5	60	<1	60
6	75	1	60
8	115	1	60
10	155	2	60
11	180	3	60
13	245	3	60

**Table 3.** Speed and response time of control subnet

Control Subnet Speed	Response Time of Source Gateway(ms)	Response Time of New Gateway(ms)	
		Return Time of Gateway (ms)	Return Time of Control Subnet (ms)
1200	785	2	375
2400	445	2	205
4800	275	2	120
9600	195	2	80
19200	155	2	60
56000	125	2	45

## 6 Conclusion

From above analysis, we can infer that by means of real-time database, we will effectively solve the problems caused by multiple TCP connections in distributed monitoring field, which include queue request, rejections of connection and loss of control network protocol packets, and improve properties of real-time and security of system. For the gateway mode based on real-time database, we also need some improvements on the following aspects:

(1) Compatibility problem with the traditional gateway. As both gateway have separate advantages and disadvantages, and real-time character is strong or weak under different scenarios, so we should work on these compatibility problems in future models.



(2) Construction problem of the real-time database. In distributed monitoring field, if the real-time property of some variables is inadequate, the process such as alarm and remote control will induce fatal problems. But other variables change slowly, such as pressure and flux. We should establish different hierarchical databases according to varied characteristic of variables.

(3) Capacity problem of the database. According to the above analysis, the more control points in database, the longer each poll time will be. And it will go against the improvement of real-time. How to control the number of points, that is, the capacity of database, to make best the real-time character of the whole system, is also problems needed to be solved in future.

## References

1. N.U.Prabhu: A Bibliography Of Books And Survey Papers On Queueing Systems—Theory And Applications, Queueing Systems 2,393-398,1987
2. Ivan Atencia, Pilar Moreno: A Discrete-Time Geo/G/1 Retrial Queue with General Retrial Times, Queueing Systems 48, 5–21, 2004
3. Muckai K. Girish , Jian-Qiang Hu: An Interpolation Approximation For the GI/G/1 QueueBased on Multipoint Pad'e Approximation Queueing Systems 26,269–284,1997
4. Nikhil Bansal: On the Average Sojourn Time Under M/M/1/SRPT, Operations Research Letters 33,195–200,2005
5. Modbus Messaging On TCP/IP Implementation Guide(Rev 1.0), Modbus.org, 2002
6. Yong J. Kima, Michael V. Manninob: Optimal Incentive-compatible Pricing for M/G/1 Queues, Operations Research Letters 31,459–461,2003
7. Krishna Kumar B., Pavai Madheswari S.: M/sup x//G/1 Retrial Queue With Multiple Vacations and Starting Failures, Opsearch vol.40,no.2,115-37, 2003