UKACC International Conference on Control 2012
Cardiff, UK, 3-5 September 2012

# VSW-SOAP: A SOAP Message Transmission Mechanism Based On Variable Sliding Window

Xiaoxuan Ma, Zhixin Chen
School of Electricity and Information Engineering
Beijing University of Civil Engineering and Architecture
Beijing, China
mxxuan@sohu.com
chenzhixin@bucea.edu.cn

***Abstract*—As an extensible lightweight message processing framework, SOAP has become the lightweight communication protocol for large scale data exchange in the distributed environment. However, one of the most important problems of SOAP is its relatively poor performance which leads to its limit in high performance applications. The paper proposes a SOAP message transmission mechanism based on variable sliding window, which can improve the performance of SOAP and support transferring data of large, flexible or unknown size via SOAP. In addition, data segmentation is designed to transfer large data and variable-size data and variable sliding window is used to control and improve the transmission efficiency of SOAP messages. The algorithm of application level congestion control was proposed and the transmission reliability of SOAP message was discussed with respect to four exception cases. Finally, the two main impact factors for the transmission efficiency based on variable sliding window such as the size of the sliding window and the length of data slice are analyzed and tested. Simulation results show that transmission efficiency is effectively improved by a reasonable parameters set of the variable sliding window.**

***Keywords-SOAP;sliding window;congestion control; transmission reliability***

## I. INTRODUCTION

Data exchange is the common, fundamental and critical requirement of distributed application in network environment. Data exchange is mainly used to solve the interoperability of different data resources in the infomationization, i.e. to solve the integration and sharing of different data resources between different heterogeneous systems. The essence of data exchange is the sharing and exchange of data. The data exchange platform is completely based on Web service technology using SOAP [1] as its communication protocol. SOAP is an extensive lightweight message processing framework based on XML [2], and has become a de facto standard for data exchange in the open network environment.

With the development of distributed computing technology, more different systems and networks at different speed are connected to the Internet, which results in more serious network congestion. Therefore, one of the most important part of Internet QoS[3] is how to deploy the peer communications at different speeds in different network to match capacity of the network. Large scale communications require effective transmission efficiency. Though SOAP provides a powerful interoperability, its own characteristics cause its lower transmission efficiency, e.g. when packaging a SOAP message, the transcoding and parsing will consume a large amount of time; generally, after data are package into a SOAP message, the data package will become more times than the original size; In addition, the time consumption of network transmission will result in the transmission of SOAP messages to take a long time. To the applications which require the higher efficiency of data transmission or large scale data transmission, SOAP can’t be effective to meet practical requirements.

According to the lower transmission efficiency of SOAP message, there are many improvements to improve the transmission efficiency as follows: to improve the transmission efficiency through improved SOAP binding [4]; according to different applications, using the appropriate SOAP message encoding rules [5]; to reduce construction and parsing of SOAP messages [6]; using compression technology to compress the SOAP message [7].

Sliding window technology is mainly used for information flow control, to coordinate the work pace of the sender and the receiver, to avoid the loss of data because the sender sends data too fast to the receiver too late to deal with [8]. This technology is not only widely used in the communication between Internet and the other network, but also used in information exchange of the different layers within the network system. In theory, sliding window can be used in anywhere which requires the flow control.

The paper proposes the VSW-SOAP(Variable Sliding Window-SOAP) to effectively improve the efficiency of data transmission. VSW-SOAP uses variable sliding window mechanism in SOAP message transmission and takes data slice as the SOAP message attachments. The basic data unit in the variable sliding window is the SOAP message with slice attachment.

The paper is organized as follows: Section II analyzes the principle of VSW-SOAP, defines the variable sliding window, discusses SOAP attachment segmentation, the structure of SOAP message and the transmission workflow of SOAP messages; Section III proposes an application level congestion control algorithm based on variable sliding window; Section IV

This work was funded by Beijing municipal outstanding talents Fund(Grant No. 2011D005017000002) and Dr. Start Fund of BUCEA (Grant No.101101815).

discusses the reliable transmission mechanism for SOAP message; Section V analyzes the main factors of VSW-SOAP and gives the simulation results; the conclusion and outlook for further work is given in Section VI.

## II. PRINCIPLES OF VSR-SOAP

### *A. Variable sliding window*

For data transmission, in extreme cases, unbounded input stream requires unbounded memory, which is clearly not applicable. Therefore, a window strategy is used to limit the number of tuples for each input stream stored in the processing of unbounded flow. The sliding window is set an interval on the data stream which only includes part of the new data of the data stream. With the arrival of new data, the window proceeds to replace the old data with new data. Therefore, the sliding window can be seen as a historical snapshot of a limited part of the data stream, while taking advantage of the characteristics of the sliding window to meet the real time and unbounded requirements of data flow.

As a historical snapshot of a limited part of the data stream, sliding window can be divided into time-based sliding window and tuple-based sliding window. The two sliding window usually assume that the window size is unchanged. In previous data transmission algorithms, many algorithms are based on the unchanged sliding window mechanism which achieves considerable results in terms of time and space complexity. However, these technologies have usually ignored the time-varying characteristics of specific issues in data stream. Therefore it is difficult to implement adaptive adjustment of data model when the data distribution characteristics are changeable [9, 10].

Variable sliding window can adaptively adjust the window size according to the change of data stream flow and data distribution in order to achieve a minimum consumption of memory space and processing time. The principle of variable sliding window is as follows: the sender maintains a continuous data set to be sent, called the sending window; at the same time, the receiver also maintains a continuous data set to be received, called the receiving window. The window is used for store data during message transmission. When updating data, the window will slide. In accordance with the principle of the sliding window, the sender has a sending window and the receiver has a receiving window. The sending window is used to implement flow control. The size of sending window represents the maximum number of data sent by the sender when the acknowledgment message is not received. The receiving window is used to control whether data can be received. At the receiving node, only when the serial number of the data is in the receiving window, the data can be accepted. The window size of variable sliding window is determined according to the data flow change. When the flow rate is quickly, reduce the size of the sliding window; otherwise, when the flow rate is very slow, increase the size of the sliding window in real time.

The differences between the SOAP sliding window and the other sliding window lies in: the data stored in SOAP sliding window are the SOAP messages, the function of the sliding window is to control the speed of the SOAP messages, to implement the network congestion control and to ensure a reliable orderly transmission of SOAP message attachment data. After the transmission is finished, the sliding window of the receiver will reassemble the slice data in the received SOAP message attachments into the original data.

### *B. SOAP attachment segmentation*

If the data of the SOAP message with attachments is large, it is difficult to read the whole data into memory only once. Even if the data can read only once, it will be very difficult and time consuming for large data encryption and decryption operations. Especially when the network quality is poor, if there are transmission data errors, retransmitting the entire data is costly. Thus, for the large scale data attachment in the SOAP message, data can be segmented into multiple slices. The data slices can be encapsulated as a new SOAP message attachment using MIME, DIME or XOP. Each time the sender can send a SOAP message attachment containing one or more data slices.

The process of data segmentation and encapsulation as a SOAP attachment is shown in Figure 1.

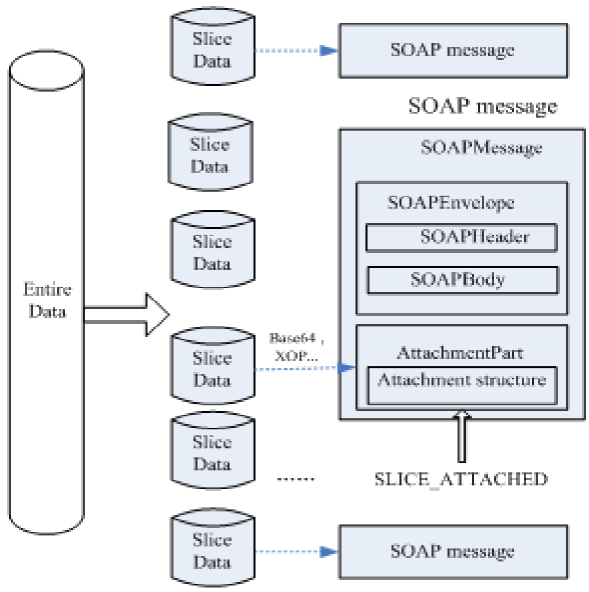


Figure 1. The process of data segmentation and encapsulation as a SOAP attachment

Because the data encapsulated in the SOAP message attachment is not complete data, but one or more data slices, the body of the SOAP message is required to at least include the following information: SOAP message number(SOAPID), the data number to be sent(DataNumber), the slice number (SliceNumber), whether it's the last data (isLastData), whether it is the last slice (isLastSlice). SOAPID is used to uniquely identify the SOAP message number that the data slice belongs to. DataNumber is used to represent the data that the data slice belongs to. SliceNumber is data slice number which represents the current transmission data slice. isLastData represents whether it's the last data. isLastSlice represents whether it's the last data slice. In this way, if there's requirement to transmit large scale attachments data, SOAP message data can be broken down into multiple small SOAP messages with a slice attachment.

By SOAP attachment segmentation, it avoids to read the entire data into memory at one time which is useful to encrypt and decrypt. Besides, the approach also supports flexible or unknown size SOAP attachment.

### *C. The structure of the SOAP message*

In the variable sliding window, the SOAP message is divided into data message and acknowledgment message.

The data message is sent by the sender and each data message contains one or more data slices. The length of slice is determined through consultation between the sender and the receiver. The slices are encapsulated in the SOAP message in accordance with the agreed specification. The basic information of data message is indicated in the body of SOAP message, which mainly includes: SOAPID, DataNumber, SliceNumber, IsLastData, IsLastSlice, the length of each data slice(size), safety information(Encryption). The structure of data message is shown in Figure 2.

```
<TransportRequest>
    <SOAPID/>
    <DataNumber/>
    <SliceNumber/>
    <IsLastSlice>yes/no</IsLastSlice>
    <IsLastData>yes/no</IsLastData>
    <Encryption/>
    <Data> </Data>
</TransportRequest>
```

Figure 2. The structure of data message

The acknowledgment message is sent by the receiver to confirm the data message issued by the sender. The acknowledgment message of the SOAP message body contains the following information: SOAPID, the type of acknowledgment message(ACKType), the data number confirmed(ACKDataNumber), the slice number confirmed(ACKSliceNumber) , IsLastData, IsLastSlice, Specific response information(ResponseInfo).The structure of acknowledgment message is shown in Figure 3.

```
<TransportResponse>
        <SOAPID/>
        <ACKType/>
        <ACKDataNumber/>
        <ACKSliceNumber/>
        <IsLastData>yes/no</IsLastData>
        <IsLastSlice>yes/no</IsLastSlice>
        <ResponseInfo></ResponseInfo>
</TransportResponse>
```

Figure 3. The structure of acknowledgment message

### *D. The workflow of VSW-SOAP*

Based on variable sliding window, the workflow of the SOAP message transmission mechanism can be described as follows:

1) The connection between the sender and the receiver can be established through the three-way handshake. The three-way handshake can be implemented based on predefined transmission protocols and ports. In the three-way handshake, the following information are required to provide: SOAP attachment package specifications, transport protocols, sliding window size, the length of slice, timeout interval, the maximum number of retransmission once timeout, the data slice number in a single SOAP message;

2) According to the conventions built by three-way handshake, the sender and the receiver are initialized, including initialization of the sliding window, starting the timers of the sender and receiver;

3) The sender reads data in accordance with the agreed length of slice, then packages the data slices into data message, places data message in a sliding window to be sent. After the transmission process is finished, the sender will wait for acknowledgment messages from the receiver;

4) The receiver receives and parses the data message. According to the parsing results, the receiver will determine whether to place the data into the sliding window or to discard the data. Once all data have been received, the sliding window will slide and the receiver will send the acknowledge message to the sender;

5) Once the sender receives the acknowledgment message, it will parse the SOAP message and determine whether all data have been sent according to the parsing results. If not all data have been sent, the sender will determine whether to retransmit data message or to slide the window in order to read into the follow-up data.

The sender and the receiver will repeat Step 4 and Step 5 until all data are sent.

## III. THE ALGORITHM OF APPLICATION LEVEL CONGESTION CONTROL

Though the SOAP message transmission mechanism based on variable sliding window can effectively improve the transmission efficiency, it also has some disadvantages, e.g. it can cause network congestion when the data are injected into the network too fast. TCP protocol itself has a congestion control mechanism. TCP congestion control usually takes the following three technologies: slowly start, accelerate the decreasing and congestion avoidance [11]. However, because TCP is the transport layer protocol, if the application layer still uses TCP's three congestion control technologies and frequently changes the size of sliding window, it will lead to inefficiency.

This section proposes an application layer congestion control algorithm based on variable sliding window in order to ensure the smooth transmission of the SOAP message. The algorithm of application layer congestion control is shown in Figure 4.

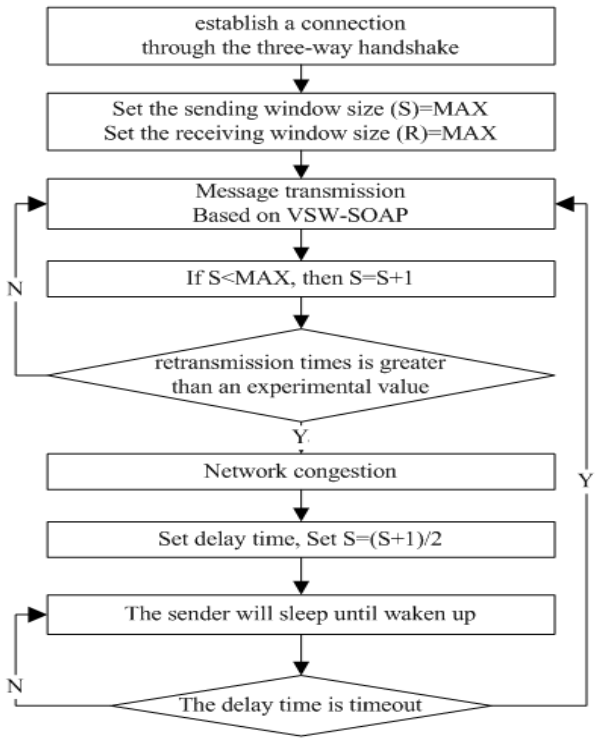


Figure 4. The algorithm of application layer congestion control

1) When the sender and the receiver establish a connection through the three-way handshake, set the maximum(MAX) of the sending window size acceptable to both sides, then set the sending window size (S)=MAX and the receiving window size (R)=MAX;

2) In accordance with the sending window size, a sliding window is used to transmit the SOAP message;

3) Once the sender receives an acknowledgment message, it will determine if the sending window size (S) is less than the maximum (MAX) of sending window. If S is less than the maximum of sending window, then S = S +1; Otherwise, keep the sending window size unchanged;

4) During transmission, if the retransmission times of a SOAP message is less than or equal to an experimental value (e.g. 3 times), the system will consider that network transmission is smooth, then will repeat Step2 to Step 4.

If the retransmission times of a SOAP message is greater than an experimental value (e.g. 3 times), the system will consider network congestion, suspend the data transmission for DELAY_T, set S=(S+1)/2. At this time if the receiver did not receive the message for a long time, the receiver will sleep until waken up by the sender.

5) Determine the delay time is timeout. If the delay time is timeout, then continues to suspend data transmission, or proceeds to repeat Step 2 to step 5 until the data transmission is completed.

During the algorithm of congestion control, DELAY_T means data transmission pause time which depends on the congestion time of current network. If the receiver doesn't receive the sender's data for a long time, it will go to sleep state or termination state, and finally will be wakened up by the sender.

## IV. THE TRANSMISSION RELIABILITY OF SOAP MESSAGE

In order to ensure the transmission reliability of the SOAP message based on variable sliding window, the following four cases will be considered during the specific analysis and design: loss of data message, loss of acknowledgment message, duplicate message and data timeout.

Once the data message is lost, if the sender still do not receive the acknowledgment message after waiting for a certain period of time, it will resend the data message. Once the acknowledgment message is lost, if the sender still do not receive the acknowledgment message after waiting for a certain period of time, it will also resend the data message. When the receiver receives the duplicate data message, the receiver will discard the duplicate message, and resend the confirmation of the maximum data slice. Once the receiver waits for data messages timeout, the receiver will resend the confirmation of the maximum data slice, and then wait for the next timeout. If the receiver waits more than timeout maximum value, it can be seen as failure of the transmission.

## V. SIMULATION

In VSW-SOAP, the main factors affecting the transmission efficiency are as follows: sliding window size S (the sending window size Ssender, the receiving window size Sreceiver), the data slice length L and the network transport protocol. The paper won't discuss the network transport protocol for it depends on its realization.

Ssender is negotiated between the sender and the receiver when establishing a connection. Besides, during the transmission, the sender and the receiver can also negotiate to adjust the sliding window size and the length of data slice based on network bandwidth, packet round trip delay, or their own hardware and software environment. Sreceiver is decided by the receiver. If Sreceiver is less than Ssender, the received data is easy to exceed the receiving window range, resulting in data frequently discarded; on the contrary, if Sreceiver is greater than Ssender, it can cause excessive memory overhead although it can cache more data messages. Thus Ssender is best equal or similar to Sreceiver.

A number of test cases are designed to test that S and L how to affect the transmission efficiency. The tests are conducted using the sliding window size from 1 to 50, and the sliding window size is equal to the receiving window size. Because the length of slice is restricted by memory size, the actual test range of length of slice gradually increases from 20K to 10M.

The relationship between the transmitting time and the sliding window size is shown in Figure 5. During the test, HTTP is used as the transport protocol and the length of slice is set 800K.

It can be seen from Figure 5 that the larger the sliding window size, the shorter the transmitting time. When the sliding window size increases to a certain extent, the transmitting time will reach an extreme and almost cease to decrease. The larger the sliding window size, the more the consumption of resources. When the window size increases to a certain extent, the memory and CPU processing speed will become a performance bottleneck. Under the current test environment, when the window size increases from 5 to 10, the transmission efficiency has been increased to a better state. In addition, the sliding window can eliminate the transmitting time difference due to the RTT. When window size increases to a certain extent, the transmitting time is almost the same under the conditions of different RTT.

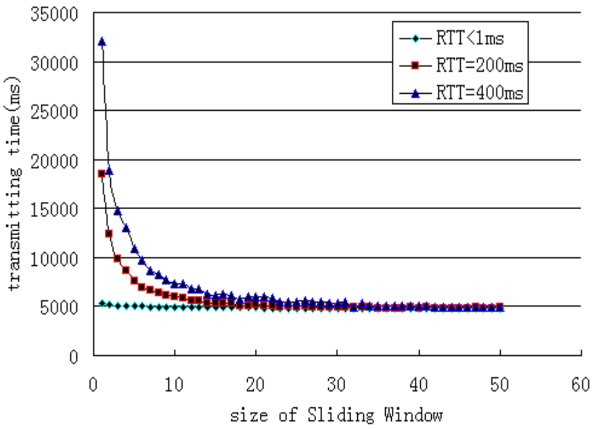


Figure 5. The relationship between the transmitting time and the sliding window size

The relationship between the transmitting time and the length of slice is shown in Figure 6. For the test, RTT is equal to 100ms and the window size is 6. The length of slice gradually increases from 20Kbytes to 10Mbytes.

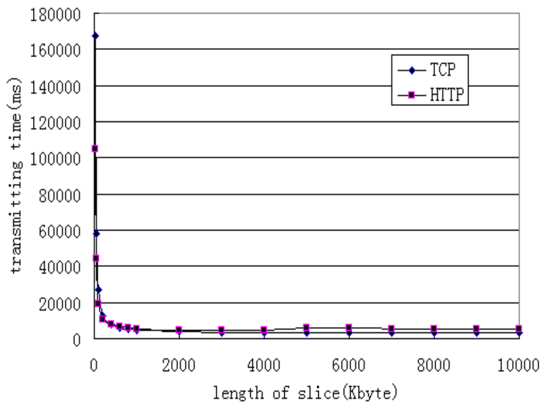


Figure 6. The relationship between the transmitting time and the length of slice

It can be seen from figure 6 that the greater the length of slice, the shorter the transmitting time. When the length of slice increases to a certain extent, the transmitting time will reach an extreme and almost ceases to decrease. Obviously, the length of slice is restricted by the memory, transmission concurrency and CPU processing capacity. Therefore, the length of slice must be set an appropriate value according to practical requirement. Under the current test environment, when the length of slice increases to 400K, the transmission efficiency has been increased to a better state.

## VI. CONCLUSION

As an extensible lightweight message processing framework, SOAP has become a de facto standard for information exchange in the open network environment. As a SOAP message transmission mechanism based on variable sliding window, VSW-SOAP can effectively improve the transmission efficiency. On the basis of the analysis of the problems faced by data exchange based on SOAP, this paper puts forward the SOAP message transmission mechanism based on variable sliding window, designs an application level congestion control algorithm and discusses the transmission reliability of SOAP message. Simulation results show that transmission efficiency is effectively improved by a reasonable set of parameters of the variable sliding window. With the further development of Web applications such as e-commerce, the VSW-SOAP will also play a greater role.

## REFERENCES


[1] Martin Gudgin, Marc Hadley et al. SOAP Version 1.2 Part 2: Adjuncts. W3C Recommendation. W3C Group, 24 June 2003, http://www.w3.org/TR/2003/REC-soap12-part2-20030624

[2] Paul V. Biron, Kaiser Permanente. XML Schema Part 2: Datatypes Second Edition W3C Recommendation. W3C Group. http://www.w3.org/TR/2004/REC-xmlschema-2-20041028. 2004

[3] Lin Chuang, Wang Yuan Zhuo, Ren Feng-Yuan. Research on QoS in next generation network, Jisuanji Xuebao/Chinese Journal of Computers, vol.31, pp.1525-1535, September 2008

[4] Christian Werner, Carsten Buschmann, Tobias Jacker, Stefan Fischer. Enhanced Transport Bindings for Efficient SOAP Messaging. In Proceedings of the IEEE International Conference on Web Services (ICWS), 2005

[5] Dan Davis, Manish P arashar, Rutgers. Latency Performance of SOAP Implementations. In Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, 2002

[6] Hui Liu, Xin Lin, Minglu Li. Modeling Response Time of SOAP over Http. Proceedings of the IEEE International Conference on Web Services (ICWS) , 2005

[7] Kenneth Chiu, Madhusudhan Govindaraju, Randall Bramley. Investigating the Limits of SOAP Performance for Scientific Computing. In proceedings of the 11 th IEEE International Symposium on High Performance Distributed Computing HPDC-11, 2002

[8] Wang BaoJun, Zhan Ying. A survey and performance evaluation on sliding window for data stream. 2011 IEEE 3rd International Conference on Communication Software and Networks, 2011

[9] Chang J H , Lee W S. A sliding window method for finding recently frequent itemset s over online data streams. J of Information Science and Engineering, 20 (4),2004

[10] Zhu X D , Huang Z Q. Conceptual modeling rules extracting for data streams . Knowledge-based System , 21 (8), 2008

[11] Zhou Jianghe, White Langford B. H ∞ controller design for TCP congestion control. 12th Annual Australian Communications Theory Workshop,2011