Improving SMS Processing Power for the Increasing Smartphone Demand

Core Network Development Department
Atsushi Mizuki, Tetsuya Matsumoto
Tomomi Uemura and Sachiko Kichimi

Demand for smartphones is dramatically increasing. SMS-push[*1] is a method used to offer a range of services, and is also used for many services aimed at smartphones (e.g. sp-mode mail arrival notification).

With the increase in smartphone demand, SMS-push traffic is also rapidly increasing, thus necessitating wide ranging expansion of core network[*2] facilities and securement of the capacity of those facilities in order to send SMS data. For these reasons, we have made developments towards improving the SMS processing power of our core network facilities.

Core network facilities used to send SMS targeted for this development include the Application Serving Node (ASN) (Inter Working Gateway (IWG))[*3] that functions as the Gateway Mobile Switching Center (GMSC)[*4], and the Media Processing Node (MPN)[*5] that functions as the SMS Center (SMSC)[*6].

From among these, ASN (IWG) already has sufficient hardware capability, so we worked to improve the processing power of the software that runs on this hardware. We also furnished MPN with updated hardware to improve processor performance, and we developed software to maximize performance of that new hardware. We carried out exhaustive investigations and analysis of the state of existing processing and resources, and re-evaluated processing, expanded resources and so forth. A detailed description of our approaches follows.

To achieve greater ASN (IWG) performance, we expanded the number of resources[*7] and number of threads[*8], optimized memory and tested actual equipment to evaluate whether performance could satisfy requirements for processing power, connection time, resumption time and plug-in time[*9].

In the process of evaluating performance, we found that there is an issue with lengthened thread stop time with plug-ins causing the call processing buffer to empty, which causes call loss to occur. This is because expanding the number of threads increases the number of threads that restart with plug-ins (issue (1)). This also resulted from latency occurring with the exclusive control required for threads to access common data areas, due to the increase in thread numbers that restart and increases in the amount of calls (issue (2)). **Figure 1** describes the buffer depletion mechanism with plug-ins.

With issue (1), because threads stop with plug-ins, the increase in threads that stop is dependent on the number of call processing requests received during that time. For this reason, we shortened

*1 **SMS-push**: SMS messages used for notification if a network service node performs some operation or starts a process in the handset.
*2 **Core network**: A network comprising switching equipment, subscriber information management equipment, etc. A mobile terminal communicates with the core network via a radio access network.
*3 **ASN (IWG)**: Equipment that performs the interworking required to send and receive SMS messages.

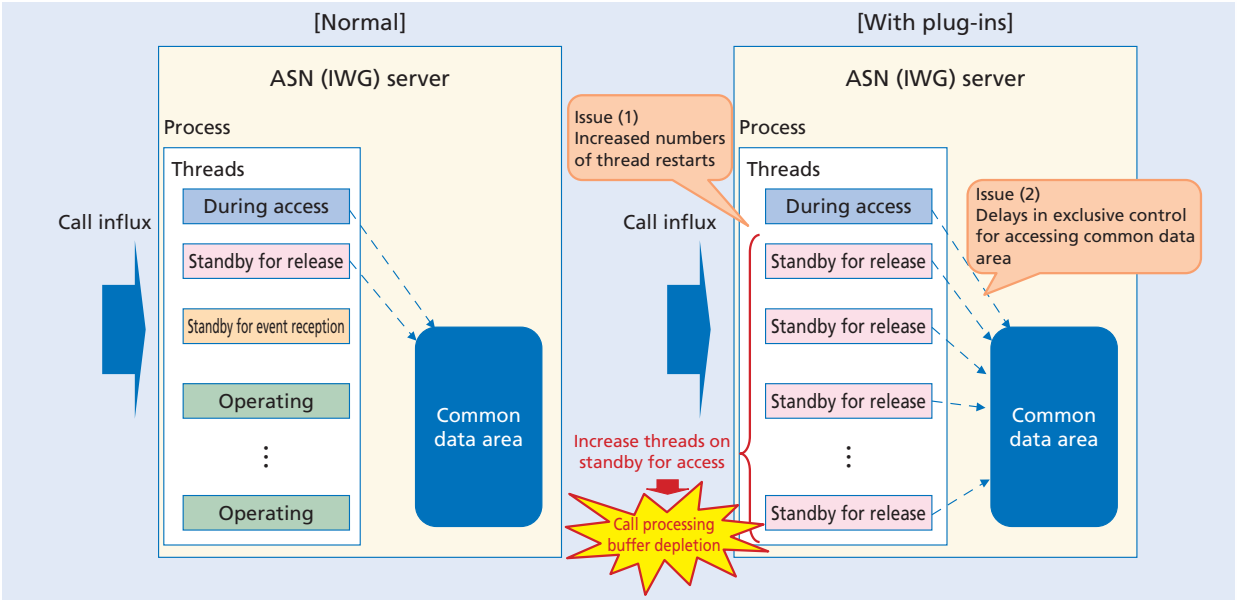

Figure 1 Buffer depletion mechanism with plug-ins

plug-in time and reduced the number of thread restarts by reducing the number of functions included in a single file by splitting files. Regarding issue (2), we reduced latency by making adjustments such as speeding up exclusive control between threads.

The introduction of new hardware with MPN improved processor and memory performance, however to improve the overall performance of SMS processing, we had to thoroughly check communications between all SMS processes to uncover bottlenecks and eliminate them. As a result, we found that the process section that writes to the disk where SMS data is stockpiled (SMS-BOX) can cause a response time out to SMS call processing when write processing is on standby, and as such is a highly likely bottleneck that should be eliminated to improve performance. To eliminate this bottleneck, we had to improve (shorten) write processing. In the write process, processing to acquire request events from call control processing takes less time than the write process to SMS-BOX, and so events for the write process are stored in a buffer and written batches when the amount of data in the buffer reaches a certain level. We considered changing the amount of data stored in the buffer (hereinafter referred to as "write size").

We assessed various write sizes from the following perspectives.

Firstly, we checked the average processing time from request event acquirement to SMS-BOX write completion. The larger the write size, the longer the processing time, however, slight increases did not have any discernible effects on processing time. We determined that it was appropriate to expand write size, since it's possible to increase write size and process a large amount of data at once if write processing time is the same.

We also checked the effects on performance caused by this change, but could not find any influence on CPU or memory usage rates.

Furthermore, we also dealt with the following two issues – issues that should be considered for improving SMS processing power.

*4 **GMSC**: Gateway equipment adjoining the SMS center server, that performs routing to the switching equipment serving the receiving user.
*5 **MPN**: A node of the NTT DOCOMO core network. It currently provides various media services including voice answering, melody call and other voice media services; video media services such as videophone answering; and SMS.
*6 **SMSC**: The SMS Center server, as standardized by the 3GPP. Stores and retransmits SMS messages.
*7 **Number of resources**: The amount of memory and so forth required to perform SMS call processing.
*8 **Number of threads**: The smallest units of processing that make up the parallel processes that are performed by the SMS call processing program.
*9 **Plug-in time**: The time it takes to augment a program if extended software functionality or improvements are required.

(1) Expanding SMS-BOX capacity

Because the number of SMS messages stored in SMS-BOX will overflow in an even shorter time with improvements to processing power if there is a malfunction in an opposing node, we expanded the SMS-BOX capacity.

(2) Restricted flow to opposing nodes

With improvements to local node processing power, because bursts of traffic might be sent that exceed the processing power of an opposing node, if an opposing node has malfunctioned, and there is retransmission from the local node when the opposing node recovers, we set a maximum limit for the number of signals that can be sent to the opposing node per unit time.

The software we used for this development has already been applied in commercial equipment. Through this development, we achieved processing power in ASN (IWG) and MPN double that of the existing. We have achieved the efficient securement of facility capacity and economic facility planning required for the increases in SMS-push traffic that will accompany rising smartphone demand.