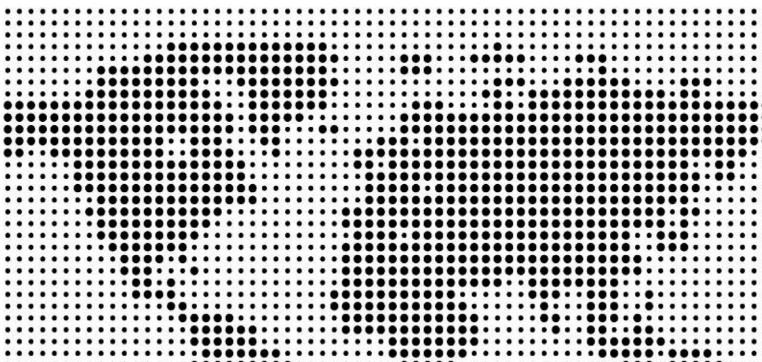
Managing Background Data Traffic in Mobile Devices



Qualcomm Incorporated
January 2012

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

Table of Contents

Executive Summary

The wide spread adoption of smart phones and availability of easily downloadable free and paid applications continues to drive the increase in data and signaling volume on mobile networks. Applications on smart phones periodically connect and disconnect to/from the network for updates. Each connection/disconnection attempt requires several message exchanges between the smart phone and the network. All these message exchanges generate signaling load on the network. This signaling load becomes a costly overhead especially when the amount of data per connection is relatively small as in the case of many common applications such as news, weather, social networking, etc. This ever increasing data and signaling load puts a strain on the operators' network.

Operators are considering a number of options to increase network capacity. These options include new techniques to improve spectral efficiency such as those being introduce in 3GPP, acquiring additional spectrum, and offload to Wi-Fi or femto cells. However all these improvements may not be sufficient to keep up with the data and signaling load. Therefore, additional complementary solutions that intelligently manage traffic can be used by operators to manage their resources efficiently.

One such complementary solution we have developed is called Network Socket Request Manager (NSRM) that will soon be commercially available. NSRM aggregates multiple application connection attempts into fewer connection attempts thereby reducing the signaling load on the network and reducing the amount of time the device spends in connected state. This technique not only reduces the signaling generated by the device but it also reduces standby power consumed by the device. Our results show that for a mix of twelve commonly used applications, NSRM achieves 30% reduction in the number of connections per hour in the device (in background mode[1]) if the smart gating period is ten minutes and connect duration is one minute. Our simulations indicate that 30% reduction in standby power can be achieved which translates into power savings in the device. These signaling reduction

[1] For the purposes of this study when the user is not actively using the device we consider it to be in background mode. For example, one such scenario occurs when the screen is off and the user does not have any audio applications running. Smart gating period is the amount of time after which all the pending application requests are allowed to connect to the network. Connect duration is the amount of time during which the gate is kept open

gains and power savings depend on the mix and type of applications running on the device.

[1] Introduction

There has been a significant increase in smart phone penetration in the mobile handset space in recent years. In 2010, 41% of mobile subscribers in the US owned smart phones. This number increased to 46% in 2011, and it is expected to reach 56% by the end of 2011 [1]. Adoption of smart phones is a global phenomenon. ABI Research predicts annual total world market shipments of smart phones in 2016 to be close to a billion [2]. This wide spread adoption of smart phones continues to drive large increases in data and signaling volume on mobile networks. In 2010 the average worldwide smart phone data consumption was 528 megabytes per month. This is expected to grow to more than 3.4 gigabytes per month by 2015 [4]. In addition, growth in smart phone signaling traffic outpaced growth in data traffic by 40% [6].

This ever increasing data and signaling load puts a strain on the operators' network. New techniques being introduced in 3GPP to improve spectral efficiency of HSPA+ and LTE networks are reaching theoretical limits. LTE-Advanced based Heterogeneous Networks improve spectral efficiency per unit area. Carrier Aggregation is another method by which operators increase capacity. Operators face challenges in acquiring large amounts of new spectrum due to limited availability of spectrum and large capital expenditures associated with it. However all these improvements may not be sufficient to keep up with the data and signaling load.

This paper focuses on a complementary solution to manage network traffic called Network Socket Request Manager (NSRM). NSRM gates application requests on smart phones when they are in background mode. By gating only background mode requests, NSRM does not affect user experience. NSRM does not gate any requests when the phone is in active mode i.e. being actively used by the user. Many applications such as news, social networking, weather, etc. generate requests even when the phone is in background mode. NSRM aggregates the initiation of these requests when the phone is in background mode from different applications and gates them. This gate can be opened immediately upon the user turning their screen on, or periodically based on a timer. When the gate is opened, all the pending requests are bundled into a batch and sent over the network which also reduces the number of connection attempts. The bundling of these requests reduces data and signaling (overhead) that would be incurred if each individual application update accessed the network separately.

[2] Mobile Applications on Smart Phones

The number of mobile applications available for smart phones is growing very rapidly. There were 10.9 billion application downloads in 2010. This is predicted to reach 76.9 billion downloads in 2014 [3]. On average, US smart phone users have sixty five applications [7]. Popular applications include news, weather, social networking, location based, advertising, or instant messaging. Some applications generate traffic more frequently than others.

A recent Qualcomm study of smart phone application traffic in background mode on a commercial network found for example that some news applications generate four to six requests per hour, compared to social networking applications which generate one to four per hour, and location based applications which generate two to three per hour. Given this variable behavior, we can see that these applications wake up anywhere from once a day, to once every five minutes. These requests are kept open for a very short duration (order of a few seconds) and there is small amount of data exchanged with each such request initiated in the uplink direction. From the same study, we also found a weather application connecting for less than 3 seconds, and uploading less than 2 kilobytes of data. Social networking applications for example connected for two to four seconds and uploaded one to three kilobytes of data. It is important to note that these behaviors may change with time; however, the application behaviors may never become consistent due to different requirements.

Frequent requests with small amount of data per request can load the network with extra signaling and drain battery life in smart phones. The smart phone radio has to frequently move from its idle state to connected state for each connection attempt. To save battery life radios prefer to stay in idle state as long as possible, and move to connected state occasionally to transfer large amounts of data and then go back to idle as quickly as possible. There have also been ongoing improvements in the cellular network standards such as 3GPP to introduce timers that can help with chatty application behavior. Figure 1 shows one such timer called the inactivity timer which allows the network to indicate to a smart phone to move back to idle state when it expires. If the amount of data per request is small, then the radio stays on with a high overhead to data ratio. This wastes network resources and increases signaling.

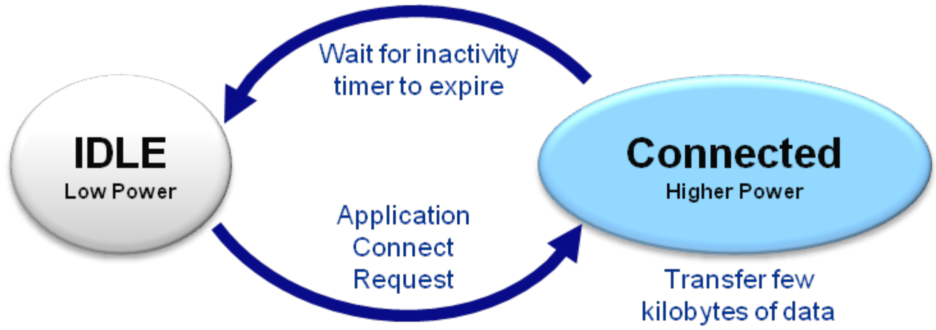


Figure 1 Inactivity Timer Driven Radio State Transitions

Majority of application requests are initiated from within the device which is referred to as a pull mechanism. Pull requests result in downlink data where an application updates the news, weather, stock quotes, etc. Some applications like email can also be notified by the network via a push method where data from a server may be periodically pushed. However, email applications also work in a pull mode. It is therefore important to have an intelligent traffic management solution in the smart phone that manages periodic pull requests. Such a solution can be complemented by a server based solution that manages trigger based push requests.

In addition to generating traffic during user interactions with the smart phone many applications generate traffic regularly even when the smart phone is in background mode. NSRM provides significant gains for the network and device battery life by managing this background mode traffic. NSRM's goal is to save network resources by minimizing the number network connection attempts, and to save battery life by minimizing the time spent in connected radio state. These goals have to be accomplished without impacting user experience and without causing harm to the normal application behavior.

In the last three years, over three hundred thousand applications were developed [5]. With such a large eco system there are thousands of application developers and many applications becoming available rapidly. Operators have started taking steps to add requirements on applications that can reduce the network impact. However given the plethora of developers and operating systems, these requirements will take time to propagate the entire eco-system. Even with these requirements, it is likely that many applications will cause disruption in the network since networks are not implemented in a homogeneous way. A solution such as NSRM implemented in a smart phone that effectively manages signaling traffic issues for all applications alleviates the burden from the application developers and allows the network operator to configure this solution suited to their network.

[3] Traffic Management Components

NSRM is designed to not impact user experience. NSRM uses the concept of active and background mode to determine opportunities for smart gating. Figure 2 shows how these are defined.

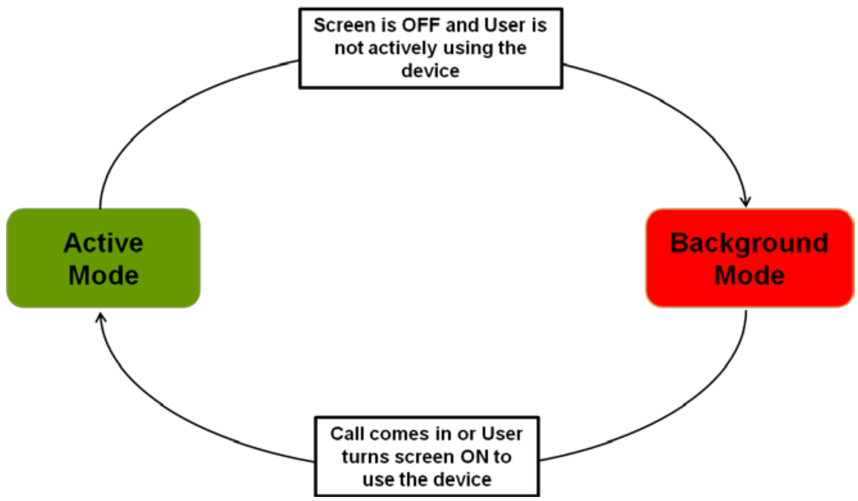


Figure 2 Active and Background Modes

3.1 Active Mode

For the purposes of this study we define active mode whenever the user is actively using the device. This includes talking to someone in a voice call, or browsing through emails, exchanging instant messages, using applications, watching a video on the screen, or listening to music, etc. So if there is any user activity, or when the device is paged, or the screen is on, or the microphone/speaker is in use, we consider the device being in active mode.

3.2 Background Mode

For the purposes of this study when the user is not actively using the device we consider it to be in background mode. The device screen is the primary user interface and acts as an indicator of the state of the device. Most of the time during the course of a day the user is not looking at the screen and it is off. This could happen during the day when the device is sitting at a desk, or in a pocket, or purse and not performing any request for the user. It can also happen during the night when the device is being charged. In addition to screen being off, other aspects of the device are also considered to determine a background mode. These include the radio state being idle, the speaker/microphone being off, and interfaces such as USB, GPS, etc. not in use.

3.3 Network Overhead

When the device is in background mode, without NSRM many applications wake up periodically and send data over the network such as news updates etc. Each such request incurs a certain overhead compared to the amount of data that is sent. Typically these applications generate very small amount of data per request, however, since the device is in background mode, there is extra overhead involved in waking up the radio, moving it from idle to connected state, and performing the transmission over the air and waiting for some time before going back to idle mode based on network timers such as the dormancy or inactivity timer. Figure 3 shows that given these background requests the amount of overhead relative to each data burst is high. The chattier the application, the higher this overhead-to-data ratio.

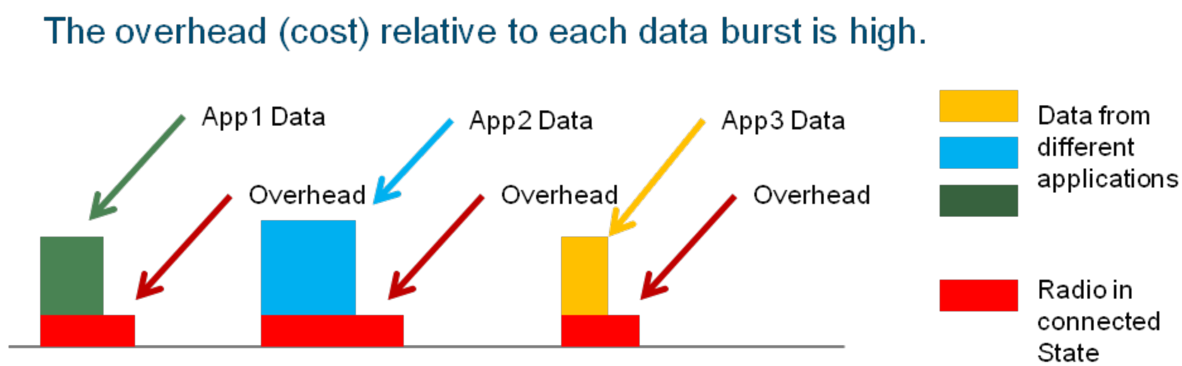


Figure 3 Effect of Inactivity Timer on Overhead

[4] Network Socket Request Manager

NSRM is an efficient method to reduce network overhead. Reduction of the network overhead is achieved by applying a gating mechanism to the applications' request before it initiates a connect. This gating mechanism must ensure it does not delay any such application requests that can negatively impact the application behavior or user perception of the application performance. A delay in the application request may sometimes result in the application malfunctioning depending on the expected outcome for the application.

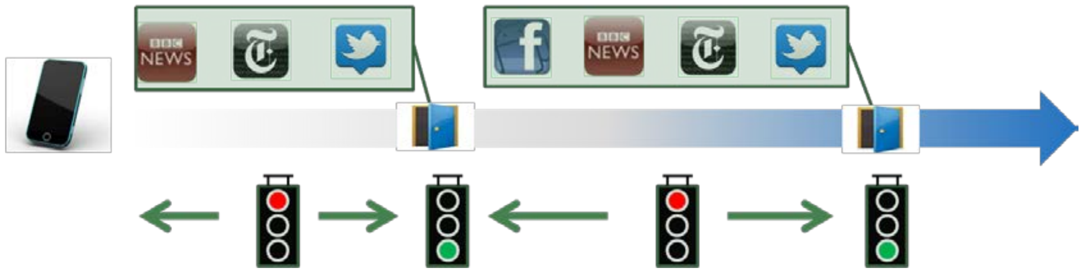


Figure 4 NSRM Background request gating

To avoid such issues, NSRM holds the application requests to connect to the network in a queue and allows them to go through when the gate is opened as shown in Figure 4. This ensures that the applications remain transparent to the network behavior.

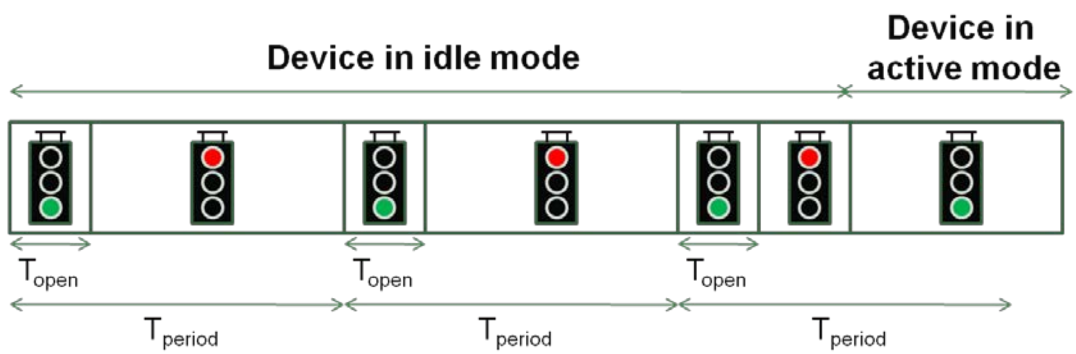


Figure 5 NSRM Timers T_{open} and T_{period}

NSRM delays asynchronous requests from multiple applications and bundles them together to improve the efficiency of such requests. This is accomplished using timers and counters that record the number of intercepted requests. In addition, if the device goes into active mode, the pending requests can be executed. When such an event triggers the opening of the gate, the radio can be turned on and all the requests are allowed to go through in a single connection. Thus, this intelligent gating reduces the number of times the device connects to the network which reduces signaling and phone power consumption.

4.1 Gate Closed

The gating mechanism is implemented using a wrapper in the device software. This wrapper intercepts application requests and accumulates them over a period of time T_{period} shown in Figure 5. During T_{period} the gate is closed from a network connectivity perspective.

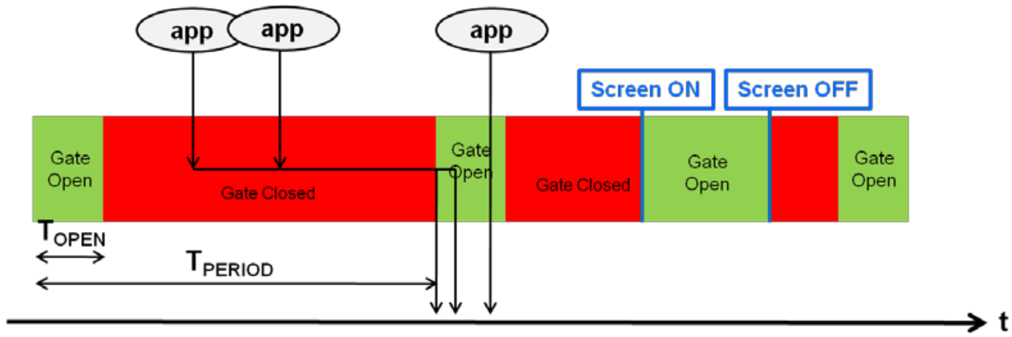


Figure 6 NSRM Gating Operation

4.2 Gate Open

In order to avoid a large accumulation of application requests in the NSRM queue, the gate is periodically opened. This allows applications to update their data periodically. As shown in Figure 5, the gate is kept open for duration of time governed by another timer called T_{open}. When T_{open} starts, requests are processed and data and signaling that has accumulated is aggregated and transmitted by the device.

At any given time, if the device is paged by the network, or user interacts with the phone (screen turns ON) as shown in Figure 6, or the phone goes to active mode due to any other reason, the gate is opened if it is currently closed. The gate is also kept open longer in case it is already open due to T_{period} expiry.

4.3 Inclusion and Exclusion Lists

To provide control to the operators, there has to be a mechanism to manage which applications' requests shall be gated by NSRM. There are two ways this can be accomplished. One way is to include an inclusion list where gating is only applied to the applications specified in an inclusion list. This method is less aggressive, and applies selective gating. Another method is an exclusion list where gating is applied to all applications except for those listed in the exclusion list. This method is more aggressive, and some specific applications such as emergency applications can be part of the exclusion list.

Operators can pre provision their smart phones with inclusion or exclusion lists and then update them over the air via an update mechanism based on standard methods such as OMA DM device management.

[5] NSRM Performance Results

In this section we present the performance results of a NSRM solution implemented on an Android smart phone prototype device on a commercial 3GPP network.

We consider an application mix with twelve commonly used applications. It is important to note that the performance results could vary significantly depending on application mix.

We show results for baseline 3GPP Release 6 commercial network currently deployed. We also show results with some of the enhanced features of HSPA such as fast dormancy with three second inactivity timer for DCH state and three and a half second inactivity timer for FACH state, U-PCH, Enhanced Cell_FACH feature with sixty second inactivity timer. T_{period} is set to ten minutes, and T_{open} is set to one minute. We also show sensitivity to application mix and push notifications. We then present simulation results for the reduction in standby power consumption.

5.1 Signaling Reduction

Figure 7 shows the results for 10 Minute T_{period} and 1 Minute T_{open}. The base application mix includes: Facebook, Facebook Messenger, Netflix, Spotify, Twitter, Instagram, Evernote, E-Mail, YouTube, Yelp, Amazon, Movies by Flixster.

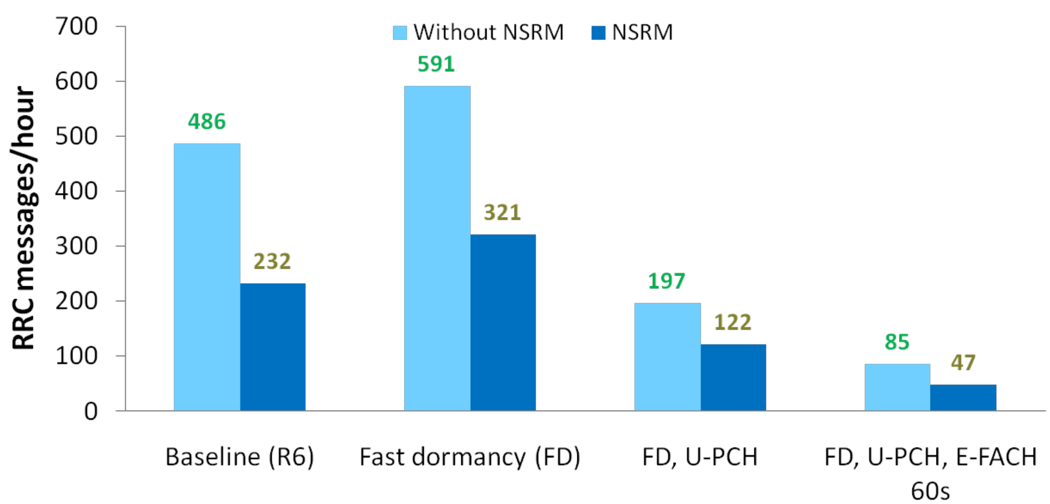


Figure 7 Signaling Reduction

NSRM achieves 52% reduction in the number of connections per hour in the device for Release-6 and 45% reduction in fast dormancy scenarios. For the cases with U-PCH and Enhanced Cell-FACH the signaling reduction is 38% and 24% respectively which indicates that NSRM still gives additional signaling reduction on top of the signaling reduction achieved by HSPA features. This means that NSRM can be applied to networks today for maximum signaling reduction, and it will continue to improve the efficiency of the network in future when new network features are introduced.

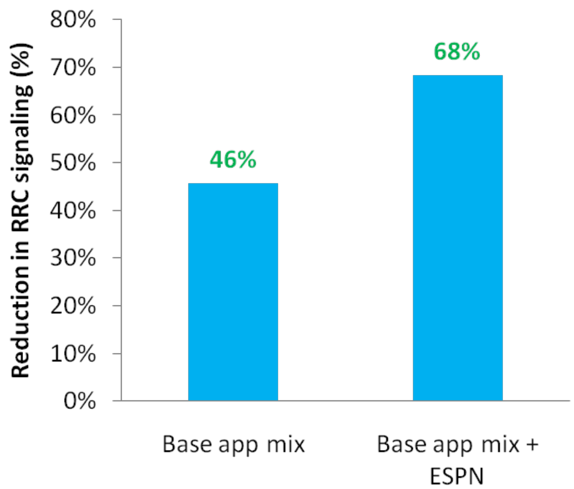


Figure 8 Sensitivity to Application Mix (Release 6)

Figure 8 shows NSRM signaling reduction depends on the application behavior and mix. Signaling reduction gains go from 46% up to 68% just by adding one application (ESPN Score Center) to the base application mix (mentioned above). ESPN application generates additional background traffic (i.e. chatty application) due to frequent score updates. NSRM gates these events in the background to achieve the high signaling reduction gains.

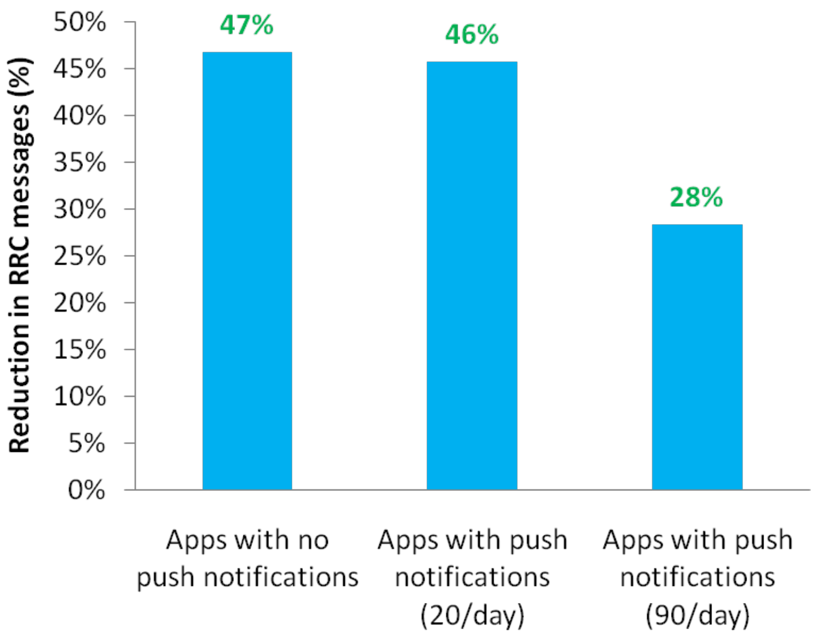


Figure 9 Sensitivity to Push Notifications

Figure 9 shows the sensitivity results with push notifications. NSRM gates pull notifications generated by applications in background mode. However, some applications use push notifications such as some email clients. Signaling reduction gains range between 28% (90 push emails per day) to 47% (no push emails).

5.2 Power Consumption Results

NSRM gates activity in background mode therefore it effectively increases the time spent in idle state and reduces the time spent in connected state. Figure 10 shows the simulated results of the reduction in standby power consumption. These results are normalized to the baseline case.

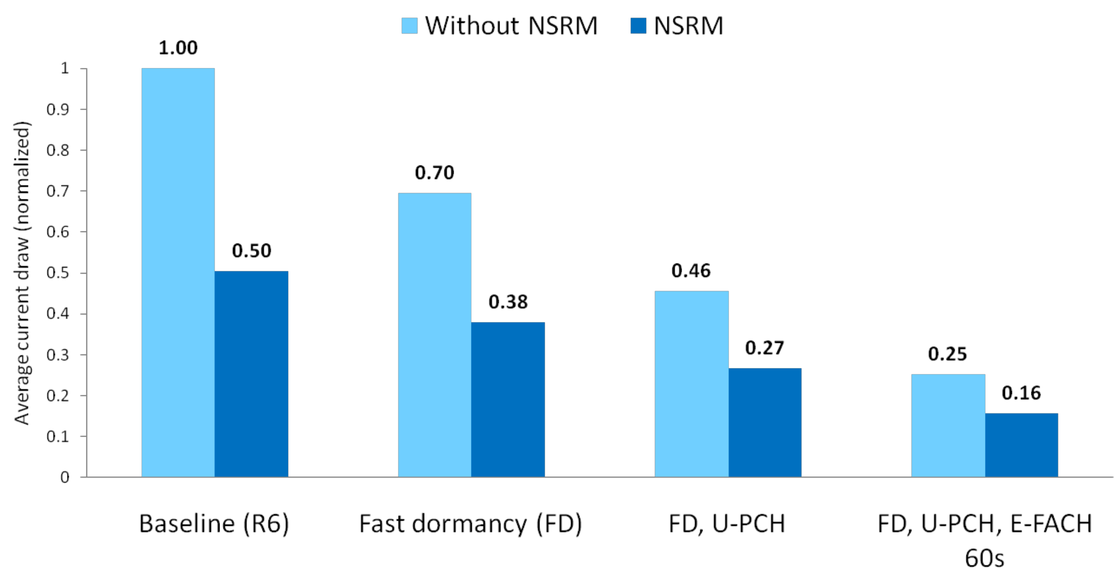


Figure 10 Normalized Standby Power Reduction

NSRM reduces the standby power consumption by 50% in both baseline and fast dormancy cases. The power reduction becomes 42% and 40% respectively when enhanced HSPA+ features are added.

[6] Future Enhancements

NSRM gating can be further enhanced by using opportunistic network load detection. The device can detect transmission opportunities based on network load fluctuations. NSRM detects when the network load is low, and opens the gate during these opportunities. This method can provide some additional capacity gains in addition to the signaling reduction gains.

NSRM can also scan for Wi-Fi before opening the gate. This can further reduce the network strain if the smart phone is in the vicinity of Wi-Fi then no connection to the cellular network is required if all the application requests waiting in the NSRM queue can be sent over Wi-Fi.

[7] Conclusion

NSRM is a complementary solution to improving network capacity that reduces signaling generated by applications on smart phones and saves battery life. NSRM is designed to not impact user experience or normal application behavior as it applies gating to requests only in background mode. Based on the tests we have performed for a smart phone prototype with twelve applications, NSRM achieves 52% reduction in number of background requests per hour which translates into reduced signaling and higher network capacity for operators. Results vary depending on application mix. Sensitivity analysis shows that the reduction in signaling can be as high as 68% depending on the application chattiness and mix. In addition to reduced signaling, NSRM simulations indicate a 50% reduction in standby power consumption due to background requests. This translates into longer battery life in smart phones and better user experience.

NSRM is designed to be operator configurable using inclusion and exclusion lists that can be updated over the air.

[8] Glossary

3GPP	Third-Generation Partnership Project
DCH	Dedicated Channel
FACH	Forward Access Channel
GPS	Global Positioning Satellite
HSPA+	High Speed Packet Access+
LTE	Long Term Evolution
NSRM	Network Socket Request Manager
OMA DM	Open Mobile Alliance Device Management
USB	Universal Serial Bus

[9] References

[1] ABI Research Report “Smartphone Technologies and Markets” May 2011

[2] ABI Research Report “Smartphone Market Data” August 2011

[3] IDC Market Analysis “Worldwide and U.S. Mobile Applications, Storefronts, and Developer 2010—2014 Forecast and Year End 2010 Vendor Shares: The “Application” of Everything. December 2010

[4] Portio Research Report “Mobile Data Usage Trends” August 2011

[5] Distimo Report January 2011

[6] Signals Research Group Report “Smartphones and a 3G Network” May 2010

[7] Flurry Analytics Research August 2011