A Self-Adaptive Information Push of Campus Micro Portal

Yuan Feng

*Modern Educational Technology Centre, Nantong University,
Nantong 226019, Jiangsu, China*

Weize Ding*

*Modern Educational Technology Centre, Nantong University,
Nantong 226019, Jiangsu, China*

Corresponding author

Abstract
Mobile micro portal is the terminal of campus mobile applications. In the traditional mobile applications, it is very difficult for the information push mechanism to balance the performance requirements in terms of the terminal diversity, electricity consumption, flow consumption and the real-time information. For these problems, a new self-adaptive information push technology is proposed in this paper. The technology analyzes the running parameters of the mobile terminal, and dynamically distributes the information push modes for the terminal. The experiment shows that the technology not only reduces the electricity consumption of the mobile terminal and saves the network flow when adapting to the terminal diversity, but also guarantees the real-time information.

Key words: MICRO PORTAL, INFORMATION PUSH, MOBILE TERMINAL

1. Introduction

The future is the era of touch, the mobile learning is attracting more and more attention from the learners due to the realization of the free learning based on “Anyone, Anytime, Anywhere and Any style” (4A) mode. As the terminal of the campus mobile application, the campus micro portal is the main platform and channel for students to get information. The traditional information push-and-pull technology has been unable to meet the existing user needs, so the corresponding information push technology becomes the new development trend for information acquisition. The information push mode is different from the information push-and-pull mode; the former is the technology based on which the server actively pushes the information to the client; its main characteristic is to actively push the information through the server and achieve the check for update without constant refresh from the client. Its advantage focuses on the initiative and timeliness of the information, so that the information can be pushed to the users at any time.

Compared with the access to the Internet based on traditional PC way, the mobile terminal is not convenient in operation and the user experience for browser is poor, it is sensitive to the flow and electricity, and characterized by the frequent network changes; moreover, in the mobile network, as a communication device, the mobile terminal is restricted by many factors. According to push technology, the active information push to the user will undoubtedly and greatly provide convenience for users and reduce the resource consumption on the terminal equipment. Therefore, push technology attracts more and more attention in terms of the superiority of the mobile terminal [1]. The push technology based on mobile terminal is relatively typical; for instance, Push Mail service of Blackberry phone from Canada RIM Company is a kind of mobile email business which transfers the traditional email to the mobile terminal [2], and provides the service of receiving and sending email at any time and any place. Push Mail service utilizes the convenience of the mobile terminal to push the information actively, so that the users can receive and handle emails for the first time at any place.

There are two information push modes based on the mobile terminal, including the polling and long connection. According to the polling [3], the client makes a request to the server on a regular basis, so as to detect the existence of the new data; the realization of the mode is relatively simple. According to the long connection, the push is achieved by a constant TCP/IP connection maintained between the client software and server.

The working principle of polling is shown in Figure 1; the client sends a request to the server, the server responses and disconnects immediately; then the client sends the request to the server again after waiting for a time interval. The working principle of the long connection is shown in Figure 2. After sending a request, the client hangs immediately until the server updates; the server actively pushes the information to the client, and the connection between the client and server is never closed until one of the parties sends the information of “closing the connection”, or causes timeout or the network error.

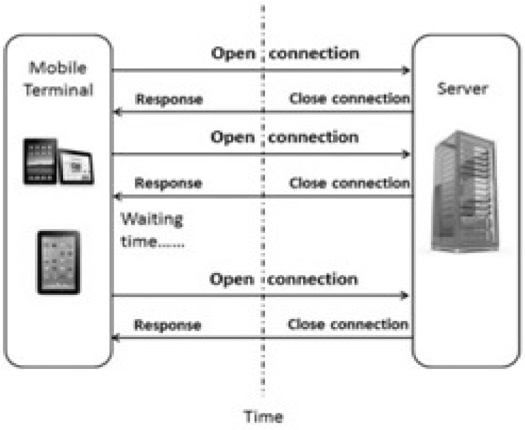

Figure 1. Principle of the polling

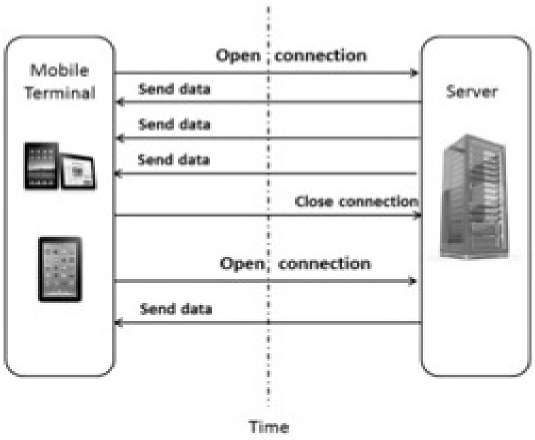

Figure 2. Principle of long connection

In the polling mode, the greater of the interval value set at the client, the slower of the polling frequency. If the polling frequency is too slow, the acquired information could have been out of date or the data may be lost [4]; if the polling frequency is too fast, too frequent establishment and disconnection, distribution of system resources will consume a lot of CPU time and memory resources on the server; meanwhile, the phone electricity and flow will be consumed accordingly; in particular, when there is no updating data on the server, the polling may consume more meaningless flow and electricity. According to the long connection mode, a persistent long connection is kept with the server, and the client keeps this IP connection in heartbeat way, which can solve the performance problem arising from polling.

Polling is a regular connection established between the client and server; the connection is closed after the response is received, so it occupies less server resources; when the non-resonance phenomenon occurs (all the clients request the server at the same time), the polling has lower request for the server memory, so the server resources are saved. Polling is suitable for the scene with fixed-frequency accidents and low real-time requirements. In the long connection mode, the server keeps a long-term connection with the client, and will occupy the server resources; with the increasing connections, the increase of server pressure may cause the server crash. Keeping a long-term connection with the server will consume a lot of phone power; meanwhile, the long connection service is related to the types of the phone operating system; so a kind of service needs to be developed for many times. Now, the long-connection push modes based on the mobile terminal include Apple Push Notification Service, Windows Phone Push Notifications, RIM's Push Mail, Google's second generation G2DM [5] and IBM MQTT protocol [6], etc.

Compared with the two push notifications, in terms of the real-time information, the long connection is superior to the polling; in terms of the mobile terminal power consumption, the long connection is higher than the polling; in terms of the network flow consumption, the long connection is lower than the polling; in terms of mobile terminal operating system diversity, the polling supports the mobile terminal operating system diversity, but a service of long connection needs to be developed for many times. The long connection and polling have their own advantages and disadvantages; in this paper, the long connection is combined with the polling to analyze the running parameters of the mobile terminal; the self-adaptive scheduling algorithm designed in this paper is the dynamically distributed push mode of the mobile terminal; when adapting to the mobile terminal diversity, the algorithm not only reduces the electricity consumption of the mobile terminal and saves the network traffic, but also ensures the real-time information and improves the information push service quality.

2. Self-adaptive information push strategy

The core of self-adaptive information push strategy is ASA (Adaptive Scheduling Algorithm) [7]. According to the algorithm, when the user sends the connection request to the server, the server acquires the running parameters of the mobile terminal, analyzes the parameters and distributes the push mode to the mobile terminal in accordance with the analysis results; when the user refreshes, the server retrieves the running parameters, analyzes the parameters and redistributes the push mode to the mobile terminal in accordance with the analysis results. On the premise that the electricity can guarantee the normal operation of the mobile terminal, the push strategy based on the dynamic distribution not only guarantees the users to acquire the push information more rapidly, but also guarantees the real-time information.

2.1 Parameter definition

For the convenience of describing ASA algorithm, the definition is listed as below:

Definition 1: The supported operation system set L-S: the set is specially designed for the long connection, and used to record the supported operating system types with long-connection provided by the server, namely, $L_{-S}=\{L_{-S1}, L_{-S2}, L_{-S3}, \ldots, L_{-Si}, \ldots, L_{-Sn}\}$; where $\{L_{-Si}|L_{-Si}=0$ or $L_{-Si}=1$, $L_{-Si}=1$ represents that the long-connection service supports the operating system of this type.

Definition 2: The long connection set *Conn*: when the client requests the connection, the server creates the long connection set, namely $Conn = \{c_1, c_2, c_3, ..., c_i, ...C_n\}$, $\{c_i \mid c_i = 0$ or $c_i = 1\}$.

Definition 3: The maximum connections Nmax: the maximum of long connections supported by the server.

Definition 4: The number of used connections Nu: the number of the long connections that the server are keeping, namely, the number of the elements with 1 as the element value in the set *Conn*.

Definition 5: Polling request cycle *T*: the time from the ending of the first polling to the beginning of the second polling.

Definition 6 : Push the scheduling threshold P_t: when the number of the long connections kept by the server is up to P_t, the following connection requests create the polling mode; P_t is the number of long connections; P_t value in the algorithm is ⌈0.66N⌉, namely, round up to an integer [8].

Definition 7: Non-empty polling: create the polling between the cline and the server; in one polling, the acquired data at the request of the client can’t be empty.

Definition 8: The non-empty polling threshold Q_t: When the flow of the non-empty polling in consecutive x times during the recent t time is not less than the consumed flow of the long connection, take the minimum x and record it as Q_t, where t=x*T; according to the experimental data, Q_t value is set as 5 in the algorithm.

Definition 9: The non-empty polling set Q_1 (t): the number of times for the client continuous non-empty polling in t time.

Definition 10: The empty polling set Q_0 (t): the number of times for the client continuous non-empty polling in t time.

Definition 10: Electricity threshold E_t: the electricity that can guarantee a push mode to operate normally in an hour. The electricity threshold based on the long connection mode can guarantee the long connection to operate in an hour normally, and recorded as E_t (L). The electricity threshold based on polling mode can guarantee the polling to operate in an hour normally, and recorded as E_t (P). According to the experimental data, the value of E_t (L) in the algorithm is set as 20%, and the value of E_t (P) is set as 10%.

Definition 12: Battery electricity E_r: the residual electricity of the battery for the mobile terminal and calculated in percentage.

2.2 ASA Adaptive Scheduling Algorithm

ASA is considered from four aspects, including the real-time information, the flow consumption, the electricity consumption and the supported operating system, so as to make the push service provided by self-adaptive information strategy become higher than the polling mode in terms of the real-time information, lower than the long connection in terms of the flow consumption, lower than the polling mode in terms of the electricity consumption and support the mobile terminal operating system diversity.

In ASA, the server distribute the push modes when the users log in for the first time or refresh; if the running parameters of the mobile terminal conforms to the long connection mode, then the long connection is the preferred way to push, so as to ensure the real-time information. The detailed process of the algorithm is shown in Figure 3.

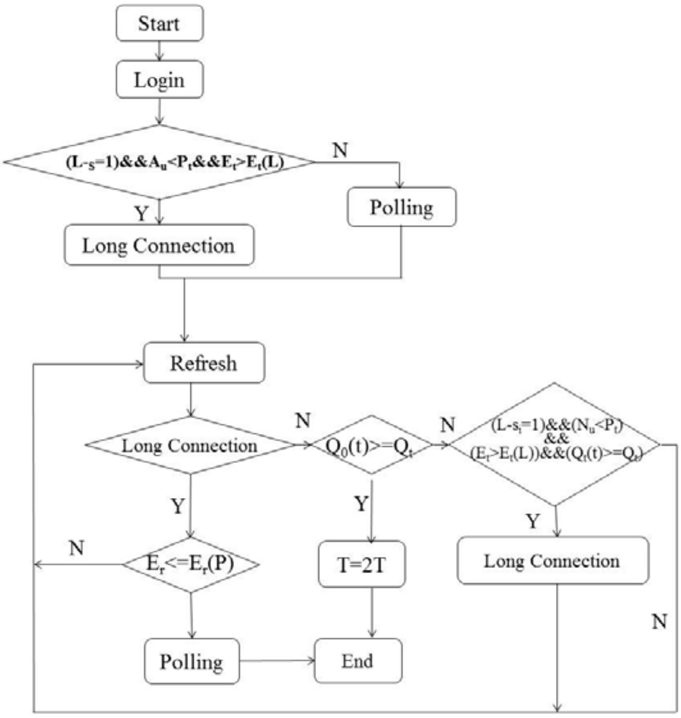

Figure 3. ASA algorithm flowchart

When the user logins for the first time, the client sends a connection request to the server; if the following three conditions are met at the same time: $L_{-Si}=1$, $N_u<P_t$, and $E_r>E_t(L)$, then the long connection is distributed to the user, otherwise the polling is distributed. When the user refreshes, if the long connection is used and $E_r<E_t$ (P), then the long connection is scheduled into the polling; if the polling mode is used and the following four conditions are met: $L_{-Si}=1, N_u<P_t, E_r>E_t(L), Q_1(t)\geq Q_t$, then the polling is scheduled into the long connection; if the polling is used and $Q_0(t)\geq Q_t$, then T is adjusted to 2T.

3. Model of performance evaluation parameters

The performance evaluation function F(R,F, E,S) is designed from four aspects, including the real-time information received on the basis of the mobile terminal (expressed as R), the consumed flow (expressed as F), the consumed electricity (expressed as E) and the operating system types of the mobile terminal (expressed as S).

3.1 Real-time information

Real-time information is judged by the average delay time of the information; the shorter the average delay time, the higher the real-time information. The real-time information R can be expressed as:

$$R=\frac{1}{\bar{t}}, \bar{t}=\frac{\sum_{i=1}^{n} t_i}{n} \qquad (1)$$

Where, n represents the number of the information received in N hours; t_i represents the delay time of the i-th information (the unit is milliseconds). Calculate the average delay time according to the acquired data. The polling delay time is related to the polling request cycle T; the greater the T valve, the bigger the t; the smaller the R value, the worse the real-time information.

3.2 Flow consumption

The flow consumption is judged by testing the average consumed flow per unit time. In this paper, the minute is set as the unit time; through testing the consumed flow in N hours, the flow consumption value is recorded every 10 minutes; then average consumed flow per minute is calculated accordingly. F evaluation model can be expressed as:

$$\mathrm{F}=\frac{\sum_{i=1}^{n} f_i}{\sum_{i=1}^{n} t_i} \qquad (2)$$

Where, f_i represents the consumed flow of the i-th sample in ti time, and the unit is KB.

Due to the difference in the flow consumption between polling and long connection, the self-adaptive push mode proposed in this paper is used for the client; during the same time interval, the flow consumed by the information for receiving the same amount of information will differ. In order to guarantee the accurate performance evaluation, when testing the flow, the multiple testing of one phone from the fixed electricity to the electricity depletion is used as the testing mode in this paper to record the consumed flow every 10 minutes, calculate the average consumed flow per minute and accurate to the second decimal place.

3.3 Electricity consumption

Electricity consumption is tested and judged by the time t_i (in minute) consumed by a cell phone from the fixed electricity to the electricity depletion. The greater the t_i, the less the consumed electricity. The evaluation model of E can be expressed as:

$$E=100\times A/t_t \qquad (3)$$

where, A is constant.

In ASA, one of the factors to trigger the scheduling conversion of the push mode is the mobile terminal electricity; for the client with long connection, when the mobile terminal electricity is less than 10%, it will convert into the polling mode. In order to guarantee the accurate performance evaluation, when the electricity is tested, the consumed time (in minute) from some fixed electricity (i.e. A value) to the 0% electricity of testing one cell phone for many times is used in this paper, and then the calculated average value is the value of t_i.

3.4 Mobile terminal diversity

Mobile terminal diversity refers to the diversity of mobile terminal operating system; it is judged by testing the types of the mobile terminal operating system can be supported by the server. The long connection is related to the mobile terminal operating type; one service needs to be developed for many times. If the server only offers the long connection service, then it doesn't support the new mobile terminal operating system. However, the self-adaptive push mode proposed in this paper combines the polling with the long connection, so that the mobile terminal with new operating system can receive the push information in the way of polling. The evaluation mode of S can be expressed as:

$$S=\sum_{i=1}^{n} s_i \qquad (4)$$

$s_i=0$ the push service doesn’t support the operating system

$s_i=0$ the push service supports the operating system

s_i represents the *i*-th kind of mobile terminal operating system. When $s_i=1$, it represents that the push service supports the operating system; when $s_i=0$, it represents that the push service doesn’t support the operating system. S value is the number of the types of the mobile terminal operating systems supported by the server; the greater the S value, the more the types of the supported mobile terminal operating system.

3.5 Analysis of evaluation parameters

Among the performance evaluation parameters R, F, E, S, the smaller R value, the better the real-time information received from the client; the smaller F value, the less flow consumed at the mobile terminal, so the flow is saved; the smaller E value, the less electricity consumed at the mobile terminal, so the longer time can be guaranteed for the mobile terminal in normal operation; the greater S value, the more types of the mobile terminals supported by the server.

In theory, the real-time information based on the self-adaptive push mode designed in this paper is close to that based on the long connection mode, the consumed mobile terminal flow is not higher than that based on the polling mode, so as to guarantee that the normal operation time of the mobile terminal is longer than that based on the long connection, and the supported mobile terminal types are consistent with the polling mode.

4. Realization of self-adaptive push technology

Self-adaptive information strategy dynamically distributes the push mode (polling or long connection) to the terminal in accordance with the terminal parameters. The working principle is shown in figure.

After the user starts the micro portal client, the client sends the connection request to the server, and delivers terminal parameters (such as IP address, battery remaining capacity, operating system types) to the server, and the server provides the suitable push mode according to above parameters. After the user’s successful login, the server acquires the terminal parameters, and instantiates Push Control category, creates the connection with server in accordance with the dynamically instantiated Polling Push or Long Conn Push of the terminal parameters, and delivers the information. The client sends the connection request to the server, and the server acquires the terminal OS type and residual capacity; if the server supports the long connections service of the terminal OS; meanwhile, the number of the long connections supported by the server is no more than the threshold, and the terminal residual capacity can keep the terminal to normally operate in an hour, then the server provides the long connection for the terminal; otherwise, the server distributes the polling mode to the terminal. The user refreshes, according to the current push mode, the server, if in polling mode, will acquire the number of times for the terminal continuous empty polling; when the number of times exceeds the empty polling threshold, the sever will duplicate the polling cycle; if the number of empty polling times is significantly decreased compared with last time (less than one half of the number of empty polling times before the adjustment of previous polling cycle), then the server will adjust the polling cycle to 0.5T; if the number of empty polling times is 0, meanwhile the server supports the long connection service of the terminal OS, then the number of long connections supported by the server is no more than the threshold, the residual electricity of the terminal can keep the terminal to operate normally in an hour, it distributes the terminal to the long connection. If it is the long connection mode, the server will acquire the terminal residual electricity; if it is less than 10% of the total electricity, then the push mode is transferred to the polling from the long connection.

5. Experimental comparison

In order to verify the self-adaptive information push strategy proposed in this paper, the strategy feasibility is verified by the information push platform; on one hand, the comparative analysis is made with the information push platform based on polling; on the other hand, the comparative analysis is made with the information push platform based on long connection.

5.1 Prototyping system design

In this paper, Samsung 9038 (system version Android 4.0), iPhone5S and HTC X310e (Windows Phone 7.5) are selected as the experimental mobile terminal platform to provide the self-adaptive information push service on the server. In order to ensure that the experiment is close to the real scene, the push information time of the server is randomly generated. The user can complete three basic operations on the mobile terminal: create connections with the server, receive information and disconnect.

After the user starts the client, the client sends a connection request to the server, and delivers the parameters including the cell phone IP address, residual electricity, operating system type to the server, and the server will provide the suitable push mode according to the above parameters.

5.2 Comparative analysis of the experimental results

In the experiment of the prototyping system, the polling, long connection and ASA modes are compared in terms of four aspects, including the real-time information, the flow consumption, electricity consumption and mobile terminal. In the polling mode, T value is set as 5s; when testing the electricity consumption, A=30%;

when testing the delay and flow consumption, N=1. Comparison of real-time information: According to Table 1, $\bar{t}$ (polling)> $\bar{t}$ (ASA)> $\bar{t}$ (long connection); ASA real-time information is better than polling-based real-time information.

Table 1. Average delay schedule

	polling	long connection	ASA
0-10min	3098	41	3040
10-20min	3038	7	3038
20-30min	3185	6	3143
30-40min	3208	7	7
40-50min	3093	7	2
50-60min	3233	41	14
$\bar{t}$	3237	14	1588

Comparison of flow consumption: According to Table 2, F (polling) > F (ASA) > F (long connection), the consumed electricity based on ASA is lower than that based on polling.

Table 2. Flow consumption

	polling	long connection	ASA
0-10min	41.3	5.3	24.4
10-20min	44.0	3.0	22.3
20-30min	46.7	3.4	24
30-40min	39.3	3.4	8.4
40-50min	43.7	3.1	4.7
50-60min	43.7	4.1	5.9
F	4.393	0.372	1.493

Comparison of electricity consumption: According to Table 2, E (long connection) > E (ASA) > E (polling), the consumed electricity based on ASA is lower than that based on long connection.

Table 3. Power consumption

algorithm	time	E
polling	52	0.535
long connection	49	0.638
ASA	53	0.588

Comparison of the mobile terminal diversity: ASA and polling support a variety of operating types; but the long connection only supports the operating system with corresponding service.

6. Conclusions

A self-adaptive information push strategy of the mobile campus micro portal terminal is proposed in this paper; the strategy analyzes the running parameters (the types of operating systems, electricity, push modes, etc) of the mobile terminal, and dynamically distributes the information push modes for the terminal (long connection or polling). The prototyping system experiment shows that the strategy reduces the electricity consumption of the mobile terminal in comparison with the long connection in addition to adapting to the terminal diversity, and saves the network flow in comparison with the polling, and ensures the real-time information. The consumed flow based on ASA is more than that based on long connection; in the future, ASA will be improved, so as to save more network flow.

Acknowledgements

This work was supported by:

Natural science research project of Nantong University: Research on the key technology of smart campus based on Android mobile terminal (Project Number:13Z038)

The development of University Digital Campus Based on Android: The development of University Digital Campus Based on Android (Project Number: 2013-R-26238)

National education and information technology research "Twelfth Five Year Plan" Youth Project: Research on the construction and application of digital campus micro portal in Colleges and Universities (Project Number: 136241013)

Nantong University lateral issues: Micro portal of Nantong University

References

1. Kim Y.S., Lee J.W., Park S.R., et al. (2009) Mobile advertisement system using data push scheduling based on user preference. *Proc of Wireless Telecommunications Symposium*, p.p.1-5.
2. Ben Letaifa A., Tabbane S. (2009) Push mail: A real case of IP con-vergence in NGN networks. *Proc of the 70th IEEE Vehicular Technology Conference Fall*, p.p.1-5.
3. Boon M.A.A., Van der Mei R.D., Winands EMM. (2011) Surveys in Operations Research and Management Science. *Applications of polling systems*, 16(2), p.p.67-82.
4. Van der Mei R.D., Winands E.M.M. (2008) A note on polling models with renewal arrivals and nonzero switch-over times. *Operations Research Letters*, 36(4), p.p.500-505.
5. Hansen J., Grnli T.M., Ghinea G. (2012) Towards cloud to device push messaging on Android: Technologies, possibilities and challenges. *International Journal of Communications, Network and System Sciences*,5(12), p.p.839-849.

6. Stanford-Clark A.J., Wightwick G.R. (2010) The application of publish/subscribe messaging to environmental, monitoring, and control systems. *IBM Journal of Research and Development*, 54(4), p.p.1-7.
7. Guan He-qing, Zhang Wen-Bo, Wei Jun, et al (2006) An application-aware web service requests scheduling strategy. *Chinese Journal of Compute*r, 29(7), p.p. 1189-1198.
8. Mei Rui (2011) *The research on cross-sever messages publish and push mechanism*, Wuhan: Huazhong University of Science and Technology.