Designing And Analysing Of Operational Load Of The Presence Service In IMS

Ivan BAROŇÁK[1], *Martin DRIENOVSKÝ*[1], *Lucia DEMETEROVÁ*[1]

[1]Institute of Multimedia Information and Communication Technologies, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, Vazovova 5 Bratislava 812 43, Slovakia

ivan.baronak@stuba.sk, martin.drienovsky1@gmail.com, luc.demeterova@gmail.com

DOI: 10.15598/aeee.v21i3.4660

Article history: Received Aug 07, 2022; Revised Nov 01, 2022; Accepted Jun 27, 2023; Published Sep 30, 2023.

Abstract. *The presence in contrast to instant messaging (IM) is a dynamic user profile that is visible to others and shares information about itself. It can be represented through statuses that are perceived by other users of the service. Status includes personal information, information about the device, location, type of terminal device or preferred way of contacting the user (messages / video / call). This information is always linked to the person concerned. The presence service uses the SIP (Session Initiation Protocol) protocol, which manages presence information and determines how it is used (for example, who can see the presence information and to what extent) and is designed for a wide range of applications. The article describes a new way of analysing operational load in the presence service based on mathematical expressions. The whole concept is derived from the Markov chain. The traffic load simulations described in this paper have shown that NOTIFY messages have a higher quantity than other system messages and increase over time. This can lead to server overload and data loss [1] [2].*

Keywords

Presence service, IMS, messages.

1. Introduction

Users of the presence service are divided into two groups:

1. Users who are providers of information about their presence, called presence entities. These users can use several devices (e. g. mobile, laptop, computer, ...) for communication, which are collectively called presence user agents (PUAs) and provide the necessary presence information. These devices send their pieces of information to a presence agent (PA), which creates a complete picture of the user's presence. The PA may be part of a presence server (PS), which may also work as a proxy server for SUBSCRIBE requests.

2. Watchers who ask the PA for information about the presence of a presence entity. They are divided into several types. The first type are watchers (so-called fetchers), who obtain from the PA information about the current state (presence) of the entity of presence. Other watchers (so-called pollers) are derived from them, who obtain information about the user's presence at regular intervals. The last type are the presence subscribers who request notifications in the event that the status of the presence entity changes [3].

2. The presence service architecture

User presence information can be obtained from many entities in the IMS (IP Multimedia Subsystem) network, no matter where they are located. For watchers, it does not matter whether they are in the same area as the presence entity or in a different network. The IMS presence service architecture is defined by 3GPP (The 3rd Generation Partnership Project) in the 3GPP TS 23.141 specification (2007m). The architecture is shown in Fig. 2.

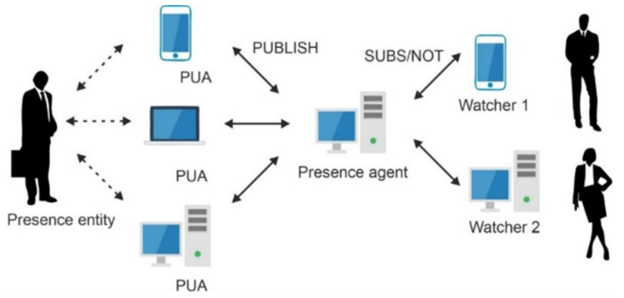

Fig. 1: The users of the presence service.

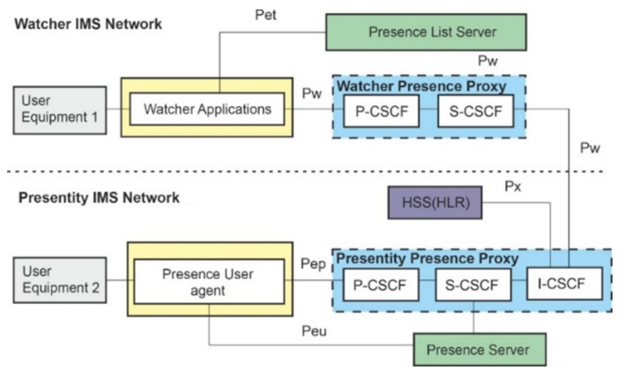

Fig. 2: Architecture of the presence service.

2.1. The Servers

The servers are responsible for sending information about the users [4].

The presence server which is located in the home network is the IMS application server. Manages presence information added from the sources that retrieve that information. It stores all the information and, when it is requested to remove the presence, selects the necessary data from the database and checks if the presence information is still valid. If the presence information is not expired the server does forward all the selected information to the watcher. It is also in charge of authorizing watchers and deciding whether or not to provide the requested collection of information.

The presence list server is an IMS application server that receives and manages information contained in presence lists. It includes several lists (for example a "buddy list"), which it obtains from presence lists from user equipment (UE). It also allows the watcher application to retrieve information from multiple presence entities.

The proxy server of watcher is in charge of watcher authentication. Identifies the destination address for presence entities and decrypts its address. The proxy server of presence entities identifies the presence server assigned to a particular presence entity. Together with the watcher's proxy server, they consist of CSCF servers. The I-CSCF serves as an access point from an external network that protects information about the IMS network, such as its configuration, capacity, or topology. The P-CSCF has limited address translation capabilities and forwards requests to the I-CSCF. P-CSCF is in charge of authorization. The role of the S-CSCF is to serve user equipment. It supports signalling interactions with the UE for call and communication settings. P-CSCF also stores UE contact addresses.

2.2. The Agents

The Agents collect presence information and provide it to the server [4, 5].

The network PA (presence entity) obtains information from network elements. It assigns the presence information to the appropriate presence entity and subscriber.

The external PA (watcher entity) provides presence information from external networks. It has the ability to determine the location of a presence server associated with a presence user.

Both agents are applications that run on user devices (for example, a mobile phone, a computer, etc.). They are responsible for asking and answering about presence status. Agents are used by users of the presence service, and based on their behavior, they may be watchers, presence entities, or both.

2.3. Interfaces

The interfaces used in the presence service have been defined in the 3GPP specification TS 23.002. The Pw, Pep and Pwp interfaces are based on the SIP protocol and are used to deliver SUBSCRIBE, NOTIFY and PUBLISH messages. Pet and Peu are based on the Hypertext Transfer Protocol (HTTP). These are used to configure the resource list server and the presence server. The Px interface is based on the DIAMETER protocol. It is used to obtain user information from the HSS database [3, 4, 5].

3. Messages

The flow of messages in the presence service differs from that in SIP. In addition to the authentication procedures that are defined in IMS, there are two elements in the service - a resource list server and a presence server. The resource list server assists the watcher entity in retrieving information, and the presence server assists the presence entity in responding to the presence retrieval request.

The body of each message in the presence service contains an XML (Extensible Markup Language) document with the necessary information. The protocol to which an XML document belongs is called The XML Configuration Access Protocol (XCAP). It is encoded in an XML document. The types of messages sent in the service are as follows:

- PUBLISH - a message through which presence information is shared and updated. The body of the message contains the current presence information encoded in the XML document. It is sent at login (pub_login), when the XML body reveals that the user is registered, present and wants to communicate. XML body additionally contains information when the request was shared. However, the request is also sent when the user logs off (pub_logout) or when the user refreshes (pub_refresh) or changes his (pub_modify) status.
- SUBSCRIBE - a message is sent to obtain presence information. It can be sent to one presence entity or to an entire group on the observer's presence list, which contains a list of users. When the user starts subscribing to the presence, a sub_initial request is sent. If the user ends with subscribing then sub_terminal is sent and if the subscription needs to be renewed, a sub_refresh request will be sent.
- NOTIFY - a message sent by the server to all watchers during active subscription. Informs the watcher about the presence of the requested user. It forms the largest set of sent messages.

If the user's status changes, a REGISTER message is sent at the beginning of the session. These messages provide information about the user's location (IP address). If users are not registered, the PA sets their presence status to offline. If they are registered, their status will be online [5, 6, 7].

3.1. Subscription

In the first step, the watcher begins to subscribe the presence of a particular user through the SIP request SUBSCRIBE. The presence server verifies that the watcher is authorized for the requested subscription and sends a 200 OK message and the current status of the presence entity via a NOTIFY message. If the status of the presence entity is changed, the presence server is notified via a PUBLISH request. Based on this, the server informs all subscribers of the presence entity about the change of status. Obtaining the presence information is also possible via the HSS server [8].

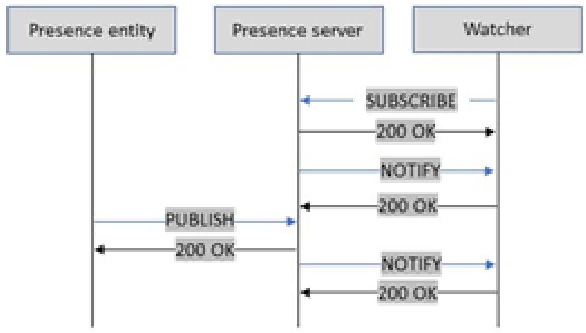

Fig. 3: Subscription of the presence entity.

3.2. User login / logout

A user login means the generation of an initial PUBLISH message, which is sent to the presence server. These are the trigger for pub_refresh messages to be transmitted at regular intervals. They end when the user logs off, when a pub_logout message is sent to the presence server.

The frequency of sending pub_refresh messages depends on the user's software settings and should not exceed an interval of less than once an hour according to the SIP protocol specification. All PUBLISH messages sent at the beginning and end of the session result in NOTIFY messages sent to each online watcher subscribed. We can express the number of online watchers by the equation n_online_watcher = n_watcher * p_online, where n_watcher is the total number of watchers and p_online is the probability of their state to be online [9, 10].

3.3. Communication

The communication in the presence service is shown in Fig. 4. UE1 is a watcher of a presence entity and UE2 is the entity of presence. The condition for subscribing the presence is that UE2 must be in UE1's 'buddy list' and UE2 must allow the subscription of presence statuses. These two users are on different networks. At the beginning of the connection, UE2 sent information about its presence to the presence server.

Steps 1 and 2: UE1 sends a SUBSCRIBE (sub_initial) request to the observer proxy server via Pw interface. After acknowledging the message, the proxy server forwards it to the presence list server. It retains the request and responds to UE1 with a 200 OK message via the proxy server (P-CSCF and S-CSCF)

Step 3: The watcher proxy server then retrieves the watcher's buddy list of UE1 from its database and sends the SUBSCRIBE request to the resource identifiers of the URIs buddy lists. Because both users are on different networks, the message is first delivered to the

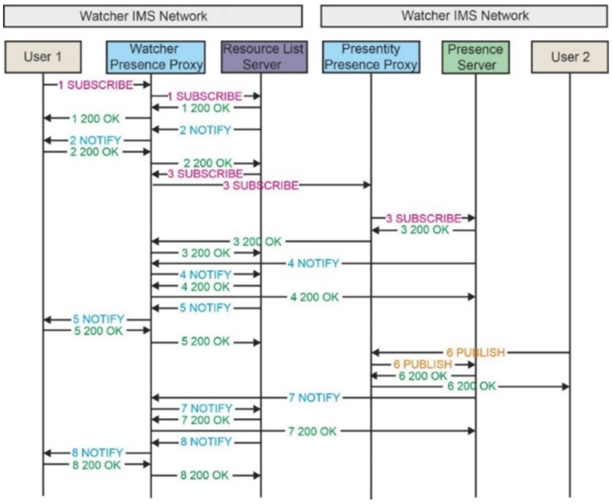

Fig. 4: Communication in the presence service.

presence entity's proxy server (I-CSCF, S-CSCF, P-CSCF). It then forwards it to the presence server. The server then responds with a 200 OK message, which is sent to the presence list server as an acknowledgment of receiving of the SUBSCRIBE request.

Step 4: The presence server looks up UE2 presence information in its database to see if the information has expired. If not, it sends the searched presence information via a NOTIFY message to the presence resource list server. The presence resource list server then responds with a 200 OK message as s response to receiving the request.

Step 5: To reduce power consumption, the presence resource list server combines multiple NOTIFY requests into one message which is sent to UE1. UE1 responds with a 200 OK message after receiving the message.

Step 6: In this step, UE2 changed its presence status and sent a PUBLISH (pub_modify) request to the presence server. The presence server stores this information in its database and responds with a 200 OK message.

Steps 7 and 8: The presence server has already detected that the presence status of UE2 has changed. It therefore sends a NOTIFY message to the watcher UE1. It is firstly delivered to the resource list server, which forwards it to UE1. UE1 responds then with a 200 OK message after being notified of the change.

In this case, the presence servers and presence resource list servers were stateful servers that processed information for both users separately. Such servers allow users to extend battery life and use less bandwidth by helping them process most messages instead of forwarding them directly to users' end devices [10, 11, 12].

4. Mathematics

The best solution to illustrate message generation in the presence service is to use discrete Markov chain. When using this type of Markov chain, the probability of transition to the next state depends only on the current state and not on previous states [13, 14].

In the case of our model, the user can be in 3 states, which are:

$s0$ - user status has remained unchanged since login

$s1$ - user status has changed since login

$s2$ - logged off user

The probability of the transition of these states can be represented by the following matrix:

$$\begin{array}{c} \\ s0 \\ s1 \\ s2 \end{array} \begin{array}{c} \quad s0 \quad s1 \quad s2 \\ \begin{pmatrix} 1-p_{01}-p_{20} & p_{01} & p_{02} \\ 1-p_{01}-p_{12} & p_{11} & p_{12} \\ p_{20} & p_{21} & 1-p_{20}-p_{21} \end{pmatrix} \end{array} \tag{1}$$

The individual states in the matrix can be expressed by probabilities:

$$P\left(s0,t|s0,\Delta t\right) = 1 - p01 - p02. \tag{2}$$

The probability that a user who is logged in will not change their presence status over time Δt:

$$P\left(s0,t|s1,\Delta t\right) = p01. \tag{3}$$

The probability that a user who is logged in will log out over time Δt:

$$P\left(s0,t|s2,\Delta t\right) = p02. \tag{4}$$

The probability that a user who is logged in will not change their status over time Δt:

$$P\left(s1,t|s0,\Delta t\right) = 1 - p11 - p12. \tag{5}$$

The probability that the online user will change their presence status over time Δt:

$$P\left(s1,t|s1,\Delta t\right) = p11. \tag{6}$$

The probability that the online user will log out over time Δt:

$$P\left(s1,t|s2,\Delta t\right) = p12. \tag{7}$$

The probability that a user who is offline will log in over time Δt:

$$P\left(s2,t|s0,\Delta t\right) = p20. \tag{8}$$

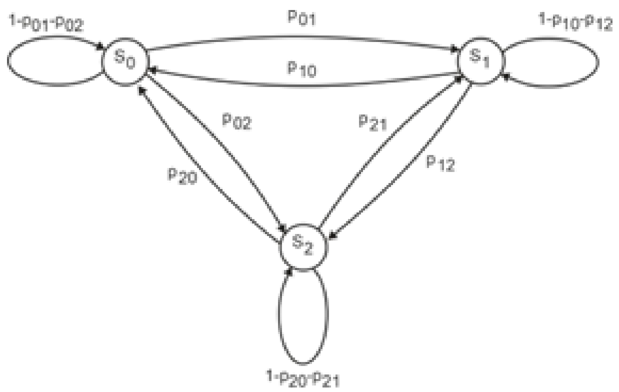

Fig. 5: Discrete Markov chain.

The probability that the offline user will change their status over time Δt:

$$P\left(s2,t|s1,\Delta t\right)=p21. \quad (9)$$

The probability that the logged out user will remain offline over time Δt:

$$P\left(s2,t|s2,\Delta t\right)=1-p20-p21. \quad (10)$$

The probability of the user's transition from one state to another can be expressed by an exponential distribution (where p_{ij} (i = 0, 1, 2 and j = 0, 1, 2)):

$$p_{ij}=\int_0^{\Delta t}\lambda_{ij}.e^{-\lambda_{ij}x}dx, \quad (11)$$

$$\lambda_{ij}=\frac{1}{t_{ij}}. \quad (12)$$

where t_{ij} - average time of message creation

According to the basic Kendall classification of the queuing system, we can define this QS through the classification A / B / n, where

A - describes the intervals between the arrival of requests to the system (to the server)

B - describes the duration of service by the server

n - represents the number of service lines (servers)

The number of messages that are transmitted is determined by the Markov chain in Fig. 5.

4.1. Number of system requirements

1) Subscribe messages

The following formulas are used to calculate subscribe messages [13, 14, 15]:

$$sub\_initial\left(t\right)=s2\left(t\right).P\left(s2,t|s0,t-\Delta t\right), \quad (13)$$

$$sub\_terminal\left(t\right)=s0\left(t\right).P\left(s0,t|s2,t-\Delta t\right), \quad (14)$$

$$sub\_refresh\left(t\right)=s0\left(t\right).Psr, \quad (15)$$

where

$$Psr=1-\int_0^{t_{sr}}\frac{1}{t_{sof}}.e^{-\frac{1}{t_{sof}}x}dx. \quad (16)$$

The total number of subscribe requests is calculated using the formula:

$$sub\left(t\right)=sub\_initial\left(t\right)+sub\_ter\min al\left(t\right)+sub\_refresh\left(t\right). \quad (17)$$

Sub_initial requests are sent each time you subscribe to the current status of a presence service user. sub_terminal is sent each time a subscription is canceled, and sub_refresh is sent periodically and monitors the user's current status. Psr represents the probability of a message creation.

2) Publish messages

The following formulas are used to calculate individual publish messages [13, 14, 15]:

$$pub\_modify\left(t\right)=s0\left(t\right).P\left(s0,t|s1,t-\Delta t\right)+s1\left(t\right).P\left(s1,t-\Delta t\right), \quad (18)$$

$$pub\_login\left(t\right)=s2\left(t\right).P\left(s2,t|s0,t-\Delta t\right), \quad (19)$$

$$pub\_logout\left(t\right)=s0\left(t\right).P\left(s0,t|s2,t-\Delta t\right)+s1\left(t\right).P\left(s1,t|s2,t-\Delta t\right), \quad (20)$$

$$pub\_refresh\left(t\right)=s0\left(t\right).Pref+s1\left(t\right)Pref, \quad (21)$$

where

$$Pref=\left(1-\int_0^{R}\frac{1}{t_m}.e^{\left(-\frac{1}{t_m}\right)x}dx\right) \quad (22)$$

$$+\left(1-\int_0^{R}\frac{1}{t_{off}}.e^{\left(-\frac{1}{t_{off}}\right)x}dx\right). \quad (23)$$

t_m - expresses the average time of change of users' states,

t_{off} - expresses the average time of users who are logged out.

The total number of publish requests is calculated as:

$$pub\left(t\right)=pub\_modify\left(t\right)+pub\_login\left(t\right)+pub\_logout\left(t\right)+pub\_refresh, \quad (24)$$

$$sum\_pub=\sum_{i=T_1}^{T_2}pub\left(t\Delta t\right). \quad (25)$$

It is given that PUBLISH requests are generated throughout the session, with pub_login being sent at login, pub_logout at logout, and pub_refresh being sent throughout the session at regular intervals (at least once an hour) to check the user's current status. pub_modify is sent when a user changes their status.

3) Notify messages

NOTIFY requests are an accompanying phenomenon of each of the sent messages, which is why they have the largest share in the number of requests in the system. They are routed by the system to the queue so that the network is not overloaded, and the ideal type for request processing is QS without losses, which means that none of the requests is rejected. The FIFO queue mode is most often used - means first in / first out, when requests are served chronologically, depending on the order in which they were routed to the queue.

Accompanying notify requests are calculated using the equation [14, 15, 16]:

$$notify\,(t) = notify\_pub\,(t)\,.watching + notify\_sub\,(t)\,, \tag{26}$$

where watching represents the time ratio during which the subscriber watches the entities of presence to the total time:

$$watching = \frac{t_t}{t_t + t_i}. \tag{27}$$

notify_sub(t) is calculated using formula (17) as:

$$notify\_sub(t) = sub(t). \tag{28}$$

notify_pub(t) is calculated using formula (24) supplemented by a watchers factor representing the number of watchers in the presence service:

$$notify\_pub(t) = watchers.pub(t), \tag{29}$$

$$watchers = \frac{watchers_{on}}{N_s}.subscribe\_list. \tag{30}$$

The divisor N_s in this formula represents the total number of all watchers and subscribe_list represents the number of watchers per user [14, 15, 16].

5. Service design

When designing our service, we work with a model of a corporate company that uses the IMS presence service in the voice area for employee IP (Internet Protocol) phones

We configure the presence service via the Communications Manager Express (CME) system with a SIP WAN (Session Initiation Protocol for communication across a Wide Area Network) connection, which allows employees to monitor the availability of other colleagues based on their phone numbers in the list. The presence service thus facilitates communication and avoids unnecessary attempts to establish a connection with a user who is currently busy and unable to answer the call.

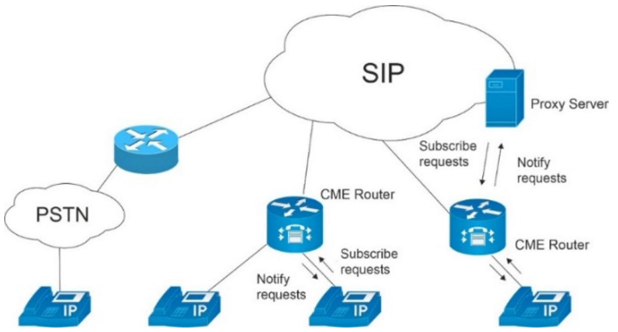

Fig. 6: Design of a corporate telephone network using the presence service.

Each time the status of one of the users changes, a notification message is sent - in the case of SIP phones and SIP trunks, these will be SIP messages. In the case of SCCP phones, these would be SCCP messages.

The presence service also supports Busy Lamp Field (BLF) functions, through which it is possible to quickly dial users and display telephone directories of missed calls, dialled numbers and numbers of received calls. Due to this function, it is also possible to receive notifications when the presence status of company extensions and other telephone numbers is changed.

If the watcher or presence entity is not on the corporate network, the subscribe request is processed by a proxy server [17, 18].

Through the BLF function, several presence indicators are displayed to employees on the phone:

1. Free line - in case the line is not used and is idle.
2. Busy line - in case the line rings or the user calls.
3. Unknown number indicator - in case the number is not registered or has access denied for monitoring its presence.

5.1. Operational load analysis

The analysis of the operational load is based on the assumption that the messages connected to watching of presence entity are performed in minutes. In our model, we determined the values as follows:

$$pub\_initial = pub\_login. \tag{31}$$

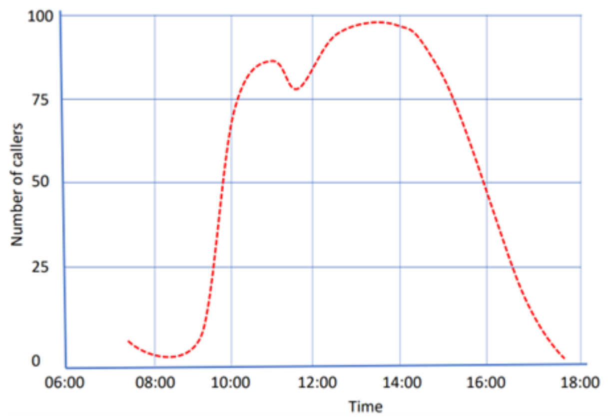

Fig. 7: Classic representation of the course of operation during the day.

which means that every time a user logs on to the network, a pub_initial request about their presence is sent

$$pub\_refresh = 2. \tag{32}$$

which means that during the user's online presence, information about their presence is sent to the server twice in 1 minute

$$pub\_terminal = pub\_login = pub\_logout. \tag{33}$$

which means that if the user logs off, a pub_terminal message is sent. Number of requests informing the server about user login and logout

$$pub\_modify = 0.5. \tag{34}$$

which means the assumption that the user would change their state once every 2 minutes in the case of a busy line

$$notify = online\_watcher.(pub\_modify + pub\_initial + pub\_terminal). \tag{35}$$

which means that all publish messages except pub_refresh are accompanied by notify requests

$$sub\_initial = 0.25. \tag{36}$$

which means the assumption that the user subscribes the presence of other users saved in the user's address book every 4 minutes

$$sub\_refresh = 2. \tag{37}$$

which means refresh of subscribing user's presence 2 times per minute

$$sub\_terminal = 0.25. \tag{38}$$

which means that the user stops subscribing once every 4 minutes.

The time for processing requests by the presence server is given as follows:

$$pub\_initial = 6ms, \tag{39}$$

$$pub\_refresh = 3ms, \tag{40}$$

$$pub\_terminal = 2ms, \tag{41}$$

$$pub\_modify = 5ms, \tag{42}$$

$$notify = 5ms, \tag{43}$$

$$sub\_initial = 10ms, \tag{44}$$

$$sub\_refresh = 8ms, \tag{45}$$

$$sub\_terminal = 3ms. \tag{46}$$

We assume the ideal case that the company has 100 employees and all 100 employees are present and working in one day. Expected working hours are from 8.00 am to 4.30 pm. In this time range, the number of calls during the day from the morning begins to increase until it reaches its peak around 11 - 12 o'clock. Subsequently, at lunch time, the curve begins to decrease until between 13 and 14 o'clock it acquires a growing character again. In the afternoon, the number of calls reaches a higher number than in the morning, and thus the server is busy. We can assume that in the time period from 2.00 pm to 3.30pm, the most frequent transmission of system requests takes place, as users most often change their presence within the presence service [14, 15, 16, 17].

We have shown the operating load using the following graphs. This is given by the formula:

$$v_i = \frac{\rho_i}{\sum_{i=1}^{8} \rho_i}, \tag{47}$$

where $\rho_i = r_i x t_i$, $i = 1$, and t_i is the processing time of the operation.

In the graphical representation, we assume that the number of logged-in users is growing. The graph shows the result of the ratio of traffic load and user login.

We can see from the graph that while some requests like pub_refresh or sub_refresh are gradually decreasing, notify messages are growing and their number is several times higher than the number of other requests in the system.

The next graph represents the ratio between the number of registered watchers and the operational load.

requests are gradually stabilizing, notify messages are seeing a significant increase in requests as the number of users increases, causing a high server load. Figures 8 and 9 show the difference in the load of notify messages compared to other messages. All the simulations can help to design the real presence service in IMS [16, 17, 18, 19].

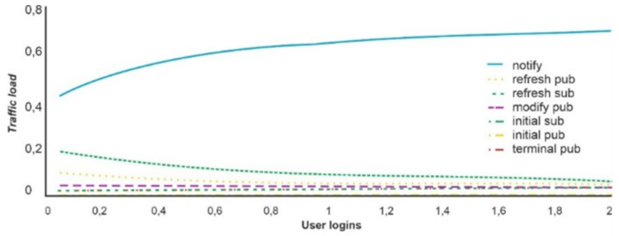

Fig. 8: Traffic load depending on user logins.

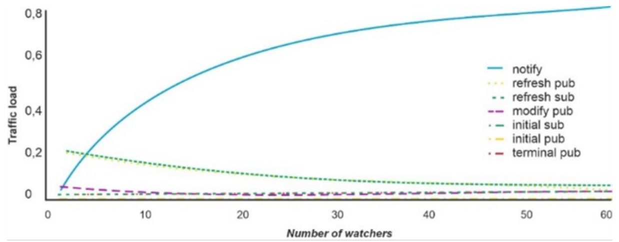

Fig. 9: Traffic load depending on the number of watchers.

It is the sharp increase in notify requests that can cause congestion of the server and the waiting queue, when incoming requests are lost due to its fulfillment and thus the quality of service is reduced. In the professional environment these collisions are a significant problem.

With the text implementation of the presence service, the problem of congestion did not occur as often as with voice, but nevertheless there were problems with notify requests in this area as well. At present, this congestion is no longer a major problem, and the presence service is receding and being replaced by newer technologies [20].

6. Conclusion

Despite the fact that IMS came to the telecommunications market in 1999, it is still a very welcome platform for telecom operators today. This is mainly because it interconnects packet-oriented networks with circuit-switched networks and guarantees the quality of calls throughout their duration and provides a wide range of services. It focuses mainly on audio and video communication. Its architecture is designed to provide users with a wide range of communication options while providing secure data transmission and error-free delivery at the expected time.

The main protocol in the IP Multimedia Subsystem is the SIP protocol, which is in charge of all multimedia sessions taking place in the system and establishing connections between endpoints. An important part of this protocol is the text requests that are sent by the system during the connection, they are also in charge of their compilation and cancellation. In addition, in the presence service, they are an important element of the whole service, as they carry information about the presence of network users and changes in their presence status.

The presence service is part of IMS, which is a dynamic user profile visible to other users on the network. These are divided into users who provide information about their presence - the presence entities and users who subscribe to this presence - the watchers. Presence status can include various information, including personal information, information about the device, location, type of terminal device, or the user's preferred method of communication. However, the main benefit of the service is information about the presence of individual entities in the network, which has found its great use [20, 21].

This presence service feature is possible due to system messages that are sent between devices throughout the session. These are PUBLISH requests - through which presence information is shared and updated, SUBSCRIBE-through which it is possible to obtain presence information and NOTIFY, which is an accompanying phenomenon of each sent request during presence subscription and is sent by the server.

The presence service has found its use in text and voice areas and especially in communication in larger companies. An example of textual use is e.g. Skype for Business application, in the case of voice use, we are talking about IP phones, which are usually enhanced by the BLF (Busy Lamp Field) function. This has brought with it the improvement of phones with light signalling the presence and availability of users. It also enabled speed dialling, displaying phone books of missed calls, dialed and numbers of received calls, and subscribing to company extensions or numbers outside the corporate network.

However, the presence service brought with it not only benefits but also complications. These occurred precisely as a result of the large number of system requests NOTIFY, which are an accompanying phenomenon of all messages sent during presence subscription. Therefore, their number was several times higher than the numbers of other system messages, and there was a risk of server congestion and reduced QoS, as there was a risk of data loss and delay.

At present, the majority of the problems of the presence service are already treated and, especially in the text area, represents a minimal load on the server. With the development of technology, it began to recede and new services based on the IMS concept began to take its place. One of them is RCS - Rich Communication Services, on which operators, manufacturers of network components and devices from all over the world work together and thus represents a symbol of their interoperability. The goal of RCS is to master mobile applications and services providing enriched communication options. Current services that RCS is able to provide include Standalone Messaging, group chat, file transfer (even during phone calls), content sharing (also during calls with another user), user presence information, IP Voice Call, video calls based on Best Effort principle, sharing the current location, Black list - list of blocked contacts and exchange of information about the possibilities of communication of individual communication devices. It is currently on the market under the joyn standard.

Despite the fact that the technology has come a long way since the beginning of the introduction of presence services to Rich Communication Services and has successfully coped with countless complications, the greatest discoveries and technological advances still await us.

Acknowledgment

This article was created with the support of the Ministry of Education, Science, Research and Sport of the Slovak Republic within the KEGA agency project - Projekt č. 034STU-4/2021 Utilization of Web-based Training and Learning Systems at the Development of New Educational Programs in the Area of Optical Wireless Technologies.

Author Contributions

Lucia Demeterová and Ivan Baronak developed the theoretical formalism, performed the analytic calculations and performed the numerical simulations. Lucia Demeterova, Ivan Baronak and Martin Drienovsky contributed to the final version of the manuscript as authors. Ivan Baronak and Martin Drienovský supervised the project.

References

[1] A.Syed, A.M.Ilyas. *IP Multimedia Subsystem (IMS)*, Handbook. Taylor & Francis Group, LLC, 2009. ISBN -13: 978-1-4200-6459-9 (Hardcover).

[2] M.Poikselkä, G.Mayer, H.Khartabil and A.Niem. *The IMS IP Multimedia Concepts and Services in the Mobile Domain.* John Wiley & Sons Ltd, 2004. ISBN 0-470-87113-X.

[3] F.Chamraz and I.Baronak, Impact of Admission Control Methods to the Traffic Management. *Advances in Electrical and Electronic Engineering*, Vol. 13, No.4, 2015, VSB-Technical University of Ostrava, pp. 280-288, ISSN 1804-3119.

[4] B.V.Mendiratta, H.Pant, Research Article, Reliability of IMS Architecture. *Australasian Telecommunication Networks and Applications Conference, 2007.*

[5] M.Kellovský, I.Baroňák, M.Kavacký, E.Chromý, The optimal Sizing of HSS Database in IMS. *In: Wireless Personal Communications, SPRINGER*, New York 2016, ISSN: 0929-6212, DOI: 10.1007/s11277-016-3750-6.

[6] L.Demeterová, Bakalárska práca, IMS – služba prítomnosti. Fakulta Elektrotechniky a Informatiky, *Slovenská Technická Univerzita v Bratislave*, 2017.

[7] M.Poikselka, G.Mayer, H.Khartabil, A.Niemi, *The IMS IP Multimedia Concepts and Services. Second edition.* John Wiley & Sons Ltd, 2006. ISBN-13: 978-0-470-01906-1.

[8] D.Petráš, Diplomová práca, IMS – služba prítomnosti. Fakulta Elektrotechniky a Informatiky, *Slovenská Technická Univerzita v Bratislave*, 2014.

[9] A.B.Johnston, SIP Understanding the Session Initiation Protocol. Second edition. *ARTECH HOUSE, INC.*, 2004. ISBN 1-58053-655-7.

[10] A.B.Johnston, SIP Understanding the Session Initiation Protocol. Third edition. *ARTECH HOUSE*, 2009. ISBN 13: 978-1-60783-995-8.

[11] SIP (Session Initiation Protocol) [online]. [cit. 2018-12-20], Available at: http://itbase.trip.sk/nw/data/protokol/sip_index.htm.

[12] N.Wilkinson, Next Generation Network Services. *John Wiley & Sons, Ltd*, 2002. ISBNs: 0-471-48667-1 (Hardback); 0-470-84603-8 (Electronic).

[13] M.Poikselka, G.Mayer: T*he IMS IP Multimedia Concepts and Services. Third edition.* John Wiley & Sons Ltd, 2009. ISBN 978-0-470-72196-4.

[14] G.Camarillo, M.A.Garcia-Martin, *The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds.* Second edition. England: John Wiley & Sons, 2006. ISBN 0-470-01818-6.

[15] K.Al-Begain, Ch.Balakrishna, L.A.Galindo, D.Moro, *IMS: A Development and Deployment Perspective.* John Wiley & Sons Ltd, 2009. ISBN 978-0-470-74034-7 (H/B).

[16] L.Demeterova, I.Baronak, E.Chromy, Research Article, IP Multimedia Subsystem - Presence Service. *WSEAS TRANSACTIONS on SYSTEMS and CONTROL*, Volume 13, 2018, ISSN: 2224-2856 (Online), Available at: http://www.wseas.org/multimedia/journals/control/2018/a525903-034.php.

[17] C. Chi, R. Hao, D. Wang, Z. Cao, Research Article, *IMS Presence Server: Traffic Analysis & Performance Modelling [online]*. 2008 [cit. 2018-12-11], Available at: http://www.ieee-icnp.org/2008/papers/Index07.pdf.

[18] M.Kvet, Prednáška 5, 20.10.2015: *Uvod do teosrie hromadnej obsluhy (THO) s praktickymi prikladmi [online]*.2015 [cit. 2019-03-30], Available at: http://www.fsi.uniza.sk/ktvi/leitner/2_predmety/SMOA/05_THO/Prednáška%205.pdf.

[19] A.Napolitano, *Rich Communication Suite [online]*. 2010 [cit. 2019-04-13], Available at: https://www.gsma.com/futurenetworks/wp-ntent/uploads/2012/10/TIMRCSTrialAantonellaNapolitano.pdf.

[20] *All About Rich Communication Services and RCS Messaging (Updated) [online]*. [cit. 2019-04-14], Available at: https://simpletexting.com/rich-communication-services-and-rcs-messaging/.

[21] F.Certik & R.Roka, Possibilities for Advanced Encoding Techniques at Signal Transmission in the Optical Transmission Medium. *Journal of Engineering (United States)*, 2016.

About Authors

Ivan BARONAK was born in Zilina, Slovakia, on July 1955. He received Master of Science degree (electrical engineering) from Slovak Technical University Bratislava in 1980. Since 1981 he has been a lecturer at Department of Telecommunications. Nowadays he works as a professor at Institute of Telecommunications of Faculty of Electrical Engineering and Information Technology, Slovak University of Technology, Bratislava. Scientifically, professionally and pedagogically he focuses on problems of digital switching systems, Telecommunication Networks, Telecommunication management (TMN), IMS, VoIP, QoS, problem of optimal modeling of private telecommunication networks and services.

Lucia DEMETEROVA was born in Bratislava, Slovakia, in November 1995. She received Engineer's degree (Telecommunications) from Slovak University of Technology in Bratislava in 2019. Nowadays she works as an IT Application Support/Analyst focusing on L2 BSS & OSS support and database management.

Martin DRIENOVSKY was born in Levice in 1999. He received a bachelor's degree (Telecommunications) from the Slovak University of Technology in Bratislava. Nowadays he is a student of master's degree in the department of Multimedia Information and Communication Technologies. In addition to studying, he gains practical experience in the IT area as a Test Specialist.