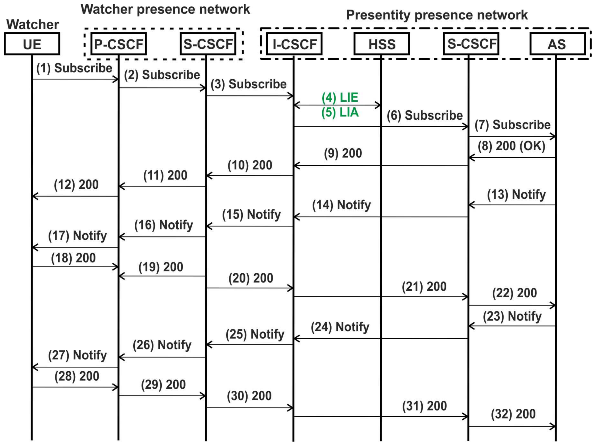

Figure 4: Process of subscribing.

ument. These documents have a large capability due to large amount of information they contain. The watcher, subscribing the information from presentities, usually obtains one of these XML documents with every change of any status. In case a smaller device with low performance and restricted memory is used, the device may be overloaded by the large amount of information and wont be able to obtain the information or utilize it in real time. To prevent the device overload, we need to find the compromise between the amount of the sent information, frequency of notify requests and information sending bandwidth [13].

6 Traffic Analysis

Studies show presence service covers around 50% (and more) of signalling traffic used by IMS core. It represents heavy load and it is necessary to solve this problem. NOTIFY messages occupy the biggest traffic load storage. Because of presence service shows the actualization of user states, the traffic load depends on user behaviour characteristics in the service. Traffic load is affected by the interval of users logging in/logging out, behaviour of users in online state and also their number of contacts. Each SIP request directed to server may initiate transmission creation. The number of transmissions processed by SIP server is an important parameter which describes the service capacity. There are several types of SIP processes with different data processing. First log in generates initial PUBLISH message r_initial_pub and the refreshing PUBLISH messages r_refresh_pub, which are generated regularly until the user log off. Then the terminal PUBLISH message r_terminal_pub is sent. Result of every initial and terminal PUBLISH message is NOTIFY message. For subscribing of presence state of user, initial SUBSCRIBE message r_initial_sub is sent to the presence server. Then refreshing SUBSCRIBE message r_refresh_sub is sent. In the end of communication, the terminal SUBSCRIBE message r_terminal_sub is sent and subscribing of presence of other users is cancelled. When the user status changes, modifying PUBLISH message r_modify_pub is sent. Each of these 3 types of messages are followed by n_online_watcher NOTIFY messages. NOTIFY messages r_notify have the biggest representation in the system. The number of users logging in/logging out is given by the following equation (1):

$$r\_notify = n\_online\_watcher.(r\_pub\_login + r\_pub\_logout + r\_pub\_modify + r\_pub\_refresh) \tag{1}$$

The number of messages depends on incoming requests and on authorized watchers. NOTIFY requests are directed to waiting queue. The server sends messages in periodical intervals to prevent the network overload. If the waiting queue is filled, next incoming messages are deleted [2], [14].

7 Traffic Modelling

Creation of messages at queuing systems could be shown as Markov chains and is shown in matrix, where users can be in 3 states represented by *s0*, *s1*, *s2*. *s0* represents the user who has not changed his or her status since coming online. *s1* represents the online user, who changed his status since he came online. *s2* represents the offline user. The probability of the status change by the matrix is shown:

$$\begin{pmatrix} 1-p_{01}-p_{02} & p_{01} & p_{02} \\ 1-p_{11}-p_{22} & p_{11} & p_{12} \\ p_{20} & p_{21} & 1-p_{20}-p_{21} \end{pmatrix} \tag{2}$$

Various states in matrix mean the following probabilities:

- Probability in which online presentity does not change your presence status over time Δt:

$$P(s_0, t \mid s_0, \Delta t) = 1 - p_{01} - p_{02} \tag{3}$$

- Probability in which online presentity changes your presence status over time Δt:

$$P(s_0, t \mid s_1, \Delta t) = p_{01} \tag{4}$$

- Probability in which online presentity goes offline over time Δt:

$$P(s_0, t \mid s_2, \Delta t) = p_{02} \tag{5}$$

- Probability in which online presentity does not change your presence status over time Δt:

$$P(s_1, t \mid s_0, \Delta t) = 1 - p_{11} - p_{12} \tag{6}$$

- Probability in which online presentity changes your presence status over time Δt:

$$P(s_1, t \mid s_1, \Delta t) = p_{11} \tag{7}$$

- Probability in which online presentity goes offline over time Δt:

$$P(s_1, t \mid s_2, \Delta t) = p_{12} \tag{8}$$

- Probability in which offline presentity goes online over time Δt:

$$P(s_2, t \mid s_0, \Delta t) = p_{20} \tag{9}$$

- Probability in which offline presentity changes your presence status over time Δt:

$$P(s_2, t \mid s_1, \Delta t) = p_{21} \tag{10}$$

- Probability in which offline presentity stays offline over time Δt:

$$P(s_2, t \mid s_2, \Delta t) = 1 - p_{20} - p_{21} \tag{11}$$

- Probability of transition from one state to another p_{ij} is given by exponential distribution (where $i = 0, 1, 2$ and $j = 0, 1, 2$):

$$p_{ij} = \int_0^{\Delta t} \lambda_{ij}.e^{-\lambda_{ij}.x}\, dx \tag{12}$$

$$\lambda_{ij} = \frac{1}{t_{ij}} \tag{13}$$

where t_{ij} represents the average time of creation message.

8 Graphic Visualisation of Message Amount

The Fig. 5 shows creation and behaviour of PUBLISH messages while the number of online users in network increases. For this example, we used a network with 825 000 users.

The number of messages is given by the following equations:

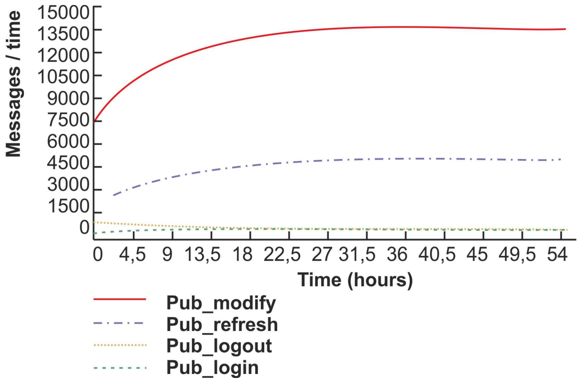

Figure 5: Creating of PUBLISH messages.

$$pub\_modify(t) = s_0(t).P(s_0, t \mid s_1, t - \Delta t) + s_1(t).P(s_1, t - \Delta t) \tag{14}$$

$$pub\_login(t) = s_2(t).P(s_2, t \mid s_0, t - \Delta t) \tag{15}$$

$$\begin{aligned} pub\_logout(t) = s_0(t).P(s_0, t \mid s_2, t - \Delta t) + \\ s_1(t).P(s_1, t \mid s_2, t - \Delta t) \end{aligned} \tag{16}$$

$$pub\_refresh(t) = s_0(t).P_{Ref} + s_1(t).P_{Ref} \tag{17}$$

$$\begin{aligned} P_{Ref} = (1 - \int_0^R \frac{1}{t_m}.e^{(-\frac{1}{t_m})x}\, dx) + \\ (1 - \int_0^R \frac{1}{t_{off}}.e^{(-\frac{1}{t_{off}})x}\, dx) \end{aligned} \tag{18}$$

where t_m represents the average time of status changes of users and t_{off} represents the average time of offline users.

The Fig. 5 shows the number of pub_modify messages exponentially increases and the number of created messages is much higher than the number of other created requests. Pub_refresh curve has a similar behaviour, but the number of messages is lower and their creation begins later. Pub_logout messages decrease because of more online users than offline. The last curve shows the number of pub_login messages. The curve increases and fluently determines with the determination of online users.

The Fig. 6 shows the amount of NOTIFY messages created in the presence service. Because of their

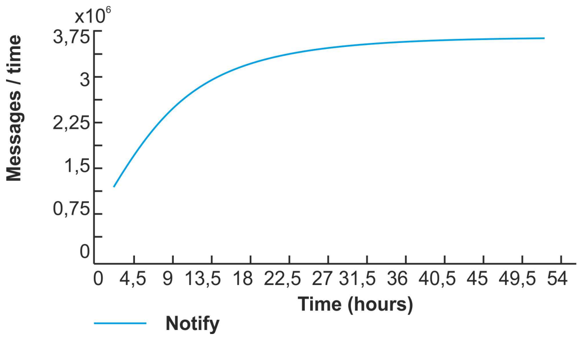

Figure 6: Creation of NOTIFY messages.

number which is much higher than in the Fig. 5, we used the time axis with lower units.

The number of NOTIFY messages is given by the following equation:

$$notify(t) = watchers(t).(pub\_refresh(t)+ pub\_modify(t) + pub\_login(t) + pub\_logout(t)) \tag{19}$$

Curve of NOTIFY messages shows the exponential growth of messages with increasing time. Creation of NOTIFY messages follows every creation of PUBLISH messages, that is why their number is significantly higher.

9 Conclusion

The presence service is being highly used nowadays. Most of the users use the applications based on presence service and the number of users is increasing. It is the reason, why servers usually confront the overload.

Due to the big number of requests waiting for processing and their deleting when the waiting queue is full, this may cause data loss and degradation of QoS in queuing systems and the presence service.

The possible solution could be the design of traffic directed to several servers. The role of the primary server is to process the incoming requests. The agent obtained in this server shows the number of incoming messages and sums up if the state of the waiting queue is fulfilled with requests. After this state, the following messages could be directed to another server located in the same cloud.

Another solution could be the system of message division due to their priority. Messages with high priority could be processed with primary server with highest performance. Messages with lower priority could be directed to server with lower performance. This solution opens the possibility of waiting queue usage much more effectively.

These solutions present wide mapping of traffic. While the applications are used by different users, these suggestions could be realized globally.

Acknowledgements: 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 - 007STU-4/2016 Progressive educational methods in the field of telecommunications multiservice networks and VEGA agency project - 1/0462/17 Modeling of qualitative parameters in IMS networks.

References:

[1] P. Bazot, R. Huber, J. Kappel, B.S. Subramanian, E. Oguejiofor, B. Georges, C. Jackson, C. Martin and A.Sur, Developing SIP and IP Multimedia Subsystem (IMS) Applications. IBM Corp. 2007, First edition, ISBN 0738489573.

[2] D. Petras, I. Baronak and E. Chromy, Presence service in IMS. The Scientific World Journal, Volume 2013 (2013), Article ID 606790, ISSN: 1537-744X (Online), http://dx.doi.org/10.1155/2013/606790.

[3] F.De Rango, M. Tropea, P. Fazio and S. Marano, Call admission control with statistical multiplexing for aggregate MPEG traffic in a DVB-RCS satellite network. GLOBECOM '05, IEEE Global Telecommunications Conference, 2005, St. Louis, MO, USA, pp. 3231-3236, ISBN: 0-7803-9414-3.

[4] R. Roka and F. Certik, Simulation and analysis of the signal transmission in the optical transmission medium. SIMULTECH 2015 - 5th International Conference on Simulation and Modeling Methodologies, Technologies and Applications, Colmar, France, 2015, pp. 219-226.

[5] F. Certik and R. Roka, Possibilities for Advanced Encoding Techniques at Signal Transmission in the Optical Transmission Medium. Journal of Engineering (United States), Vol. 2016, Article number 2385372, Hindawi Publishing Corporation, ISSN 2314-4904.

[6] A. Kovac, M. Halas and M. Orgon, E-model MOS estimate improvement through jitter buffer packet loss modelling. Advances in Electrical and Electronic Engineering, Vol. 9, No. 5, 2011,

VSB-Technical University of Ostrava, pp. 233-242, ISSN 1804-3119.

[7] J. Frnda, M. Voznak and L. Sevcik, Impact of packet loss and delay variation on the quality of real-time video streaming. Telecommunication Systems, Vol. 62, No. 2, 2016, Springer New York LLC, pp. 265-275, ISSN 1018-4864.

[8] M. Halas and S. Klucik, Modelling the probability density function of IPTV traffic packet delay variation. Advances in Electrical and Electronic Engineering, Vol. 10, No. 4, 2012, VSB-Technical University of Ostrava, pp. 259-263, ISSN 1804-3119.

[9] K. Al-Begain, Ch. Balakrishna, L. Angel Galindo and D.M. Fernandez, A Development and Deployment Perspective. John Wiley & Sons, 2009, ISBN: 978-0-470-74034-7.

[10] M. Poikselka, A. Niemi, H. Khartabil and G. Mayer, The IMS: IP Multimedia Concepts and Services. 2nd Edition, John Wiley & Sons, 2007, ISBN: 978-0-470-03183-4.

[11] N. Wilkinson, Next Generation Network Services: Technologies and Strategies. John Wiley & Sons, 2002, ISBN: 978-0-471-48667-1.

[12] M. Ilyas and S.A. Ahson, IP Multimedia Subsystem (IMS) Handbook. Taylor & Francis Group, 2009. ISBN 9781420064599.

[13] G. Camarillo and M.-A. Garcia-Martin, The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds. 3rd Edition, John Wiley & Sons, 2008, ISBN: 978-0-470-51662-1.

[14] C. Chi, R. Hao, D. Wang and Z. Cao, IMS presence server: Traffic analysis & performance modelling. IEEE International Conference on Network Protocols, 2008, Orlando, FL, USA, October, 2008, pp. 63-72.