*Tomasz Rak**

MODEL OF INTERNET SYSTEM CLIENT SERVICE

At present, service of different kinds of internet clients involves necessity of defining their behaviour in various situations. To determine optimal solutions it is purposeful to know and to describe objectively their behaviour by means of mathematical models. In this article there are presented existing solutions of the discussed problem and the own model of service of internet clients is proposed.

Keywords: *internet system, internet model, internet client*

MODEL OBSŁUGI KLIENTÓW SYSTEMU INTERNETOWEGO

Obecnie obsługa różnego rodzaju klientów internetowych pociąga za sobą konieczność określenia ich zachowań w różnych sytuacjach. W celu wyznaczenia optymalnych rozwiązań celowe jest poznanie oraz obiektywne opisanie tych zachowań za pomocą modeli matematycznych. W niniejszym opracowaniu przedstawiono istniejące próby rozwiązania tego problemu oraz zaproponowano własny model obsługi klientów internetowych.

Słowa kluczowe: *system internetowy, model internetu, klient internetowy*

1. Introduction

Usage of Internet Systems (IS) becomes more and more common. Clients' requirements concerning services rendered by such systems also increase. To determine optimal solutions it is necessary to know behaviour of Internet Clients (IC) and objective description of their behaviour by means of mathematical models.

Clients who received access to internet services, have certain expectations concerning those services. The transactions should be carried out consequently, without the necessity of repetition of the process. The most natural way of determining a standard of the services both for the providers and their clients should be based on mutual relations between the mentioned parties. For instance, an internet provider differentiates various kinds of services, and consequently may offer quicker purchase transaction than browsing. Then the system is clear for clients and they are not surprised by the method of service.

Currently, in most cases IS treat all their clients in the same way. They are lined and served in the First-In First-Out (FIFO) mode. When the system is overloaded, the clients are served with delay or errors occur. Service emerging is difficult to anticipate.

*Rzeszow University of Technology, Control and Computer Engineering Chair, trak@prz-rzeszow.pl, tel. (+48-17) 865 17 67

Examples of various kinds of such applications are on-line IS or real-time database service systems [12, 13]. The delay as it is seen by a client is time spent by a query in the net and the IS. At the same time, along with the increasing number of clients, internet servers become potential critical points, so called bottlenecks. Thousands of clients who try to access the resources concurrently trigger a stream of queries which the system is unable to process. The simplest solution is to limit the number of users, i.e. denying access to some of them. One could implement an algorithm deciding whether the service should be granted or not. In the case the access is not granted, the client receives short message informing them of access denial. Rejecting clients is acceptable since no real system can guarantee servicing all clients. Therefore a solution would be to limit the number of clients who can use the resources concurrently or provide the services on determined (feasible) levels of access for previously specified groups of users [1, 9, 10].

A test of mathematical performance of IC, its kinds and properties is an aim of this article. Types of clients and characteristics of IS are described in 2 section. Then there are examples of existing mathematical models in literature along with elements of modelling clients. Section 4 includes a proposal of formal model of internet system client (general applicability). Next part of this article is a simplified example of using all possibilities of the introduced model.

2. Properties of IS and types of clients

We can distinguish four most important types (groups) of IS:

1) bank,
2) stock exchange (auction),
3) shop,
4) internet search engines.

We can determine the features each of them should have. Reaction time to a client's query is very important in all cases. They all use one database common to the whole system (it is not necessary). Correctness of performed operations is significant in every case.

Analysis of the problem [1, 2, 6, 9] leads to a conclusion that the most important features of IS are: time of reaction, common database, completeness of executed operations.

2.1. Reaction time

We cannot foresee delays arising in the internet network between the database system and IC. Overrunning emerging in the communicative section is beyond our reach. However, it is necessary to inform a client of IS that the lack of service is not any fault of the system.

2.2. Database

Sometimes there is a possibility of dividing a database into independent parts. It enables concurrent service, acceleration and increased security of the system operation, which in many cases is important, and even necessary.

For IS the following bases can be enumerated:

1) clients,
2) statistics of the system,
3) other data (operational),
4) archives.

Additionally, for banking systems it appears balanced of customers' accounts base. For stock (auction) systems there is also the shares value base. For shop systems there are two additional bases: goods prices and balance of customers' accounts. For systems of internet search engines the links to WWW pages base is appeared.

The above-mentioned division suggests also similarities. As a matter of fact implementation can be different depending on numerous factors: size of the system, economy, application, etc.

2.3. Completeness of executed operations

Completeness of IS can be described by validity, integrity and updating of information. For validity and integrity the following systems can be enumerated on one level: bank, stock and shop and on lower level: internet search engine. For updating the following systems can be listed in respective order: stock, bank, shop and internet search engine.

2.4. Types of clients

IS clients can be divided into certain groups according to: performed tasks, time spent in IS and generated load.

2.4.1. Types of clients due to performed tasks:

- Classic – using the basic properties of the system (logging into the system, browsing through the offer of the service, using the service, transactions, determining the balance of their account, checking the correctness of the transaction).
- "Guest" – looking only through information on the whole problem (checking the possibilities of the system, accidental opening of a web site as a result of surfing the internet).
- Administrator – a person managing the network (logging into the system with administrator's rights, introducing necessary changes, configuration of system operation and its specific behaviour).

2.4.2. Types of clients with relation to time they spend in the system:

- sporadic (short-term) clients – downloading needed information or surfing through web sites,
- permanent (long-term) clients – using some of the functions offered by the system.

2.4.3. Types of clients concerning the loading of the system:

- surface – clients who do not load the whole system but only the first layer,
- overall – clients who load the inner layers,
- partial – clients who load some parts of the system.

3. Task performance in IS

IS should guarantee a level of services concerning traffic as declared by clients [11]. In case of congestion it is necessary to apply counteraction mechanisms [5, 8, 3]. Testing with the use of models becomes necessary before implementation of the system, when guaranteeing Quality of Service (QoS) in IS is in question. The models concern IS and IS clients. Determining behaviour of a client of such a system also becomes indispensable. There are different types of clients (introduced in section 2). Such models always take time relations into account. Clients can be divided into various classes as for different servicing times.

Ferrari classifies the clients into two types [6]:

1) static, those who are not able to modify their own queries, which in turn leads to incomplete use of IS;
2) dynamic (changeable), who can modify their own queries; in that case IS also modifies distribution of its resources.

Andersen in his work [2] defines IS model as a time:

$$T_{\text{all}} = T_{\text{redirect}} + T_{\text{data}} + T_{\text{server}} + T_{\text{net}} + T_{\text{client}} \tag{1}$$

where:

T_{redirect} – task redirection time (e.g. to another server, if it is necessary),

T_{data} – time required for transport of data to a disk of a local server or to a disk of a remote server,

T_{server} – processing time on server side,

T_{net} – transport time through the network,

T_{client} – processing time on client's side.

$$T_{\text{data}} = \begin{cases} T_{\text{lstartup}} + \dfrac{D_{\text{sd}}}{B_{\text{disk}} \times d_1} \\ T_{\text{rstartup}} + \dfrac{D_{\text{sd}}}{\min(B_{\text{disk}} \times d_1, B_{\text{lnet}} \times d_2)} \end{cases} \tag{2}$$

where D_{sd} – disk size data.

If the files exist locally then the time required for transport of data is a ratio of files size (D_{sd}) to product of the available bandwidth of disk memory channel (B_{disk}) and the load of the disk channel (d_1), plus starting load (T_{lstartup}). If a great deal of competing queries emerge, the efficiency of the transmission falls adequately. As a rule, the positioning times are ignored due to their low importance in relation to query servicing times. If the data are not found locally then values of the width of the transmission band of B_{lnet} net and band loading of the d_2 net are taken into account. T_{rstartup} is net starting load.

$$T_{\text{server}} = CPU_{\text{load}} \frac{SO}{CPU_{\text{server\_speed}}} \tag{3}$$

where SO – Server Operation (required quantity of the operations of server).

The number of serviced queries depends on the speed, processor load and the amount of queries.

$$T_{\text{server}} = \frac{CO}{CPU_{\text{client\_speed}}} \tag{4}$$

where CO – Client Operation (required quantity of the client's operations).

The number of executed client's operations depends on their amount and speed of the client's server.

$$T_{\text{server}} = T_{\text{nstart}} + \frac{C_{CS}}{NB} \tag{5}$$

where:

C_{CS} – Communication client-server (quantity of bytes sent between the client and the server),

NB – Net Bandwidth.

Coefficient T_{server} determines time required for communication between a client and a server. It depends on the width of net transmission channel. The initial value of T_{nstart} is negligibly small and mostly is used for initiation of the channel between the client and the server.

As noticed in available publications, the models are rather general and describe the phenomenon connected with the IC servicing in a fragmentary way. They can only be a basis for further insight. Therefore, in the subsequent part, the author proposes his own mathematical model.

4. Proposed mathematical model of IS clients

Creating the mathematical model of IC comes from necessity of more exact analysis of IS which working and building are mainly dependent on only clients' behaviour.

It has been assumed that every IS client can be attributed to a certain client class. Marking a set of client types (classes) as TK and assuming that the client types number is k, we obtain:

$$TK_k = \{(MK_1)_k, (MK_2)_k, \ldots (MK_i)_k\} \tag{6}$$

Model of i client type has been marked as MK_i. Model of a particular type client depends on numerous parameters, such as sort of services, time limits imposed on collective time of service execution, intensity and distribution of clients' localization. Therefore i client type model can be defined as the following:

$$(MK_i)_k = (O_i, D_i, F_i)_k \tag{7}$$

where:

O_i – client service type,

D_i – time limit, i.e. time interval, in which the client has to be served,

F_i – function describing arrivals of a particular clients' type.

Depending on the client's type we can determine different kinds of services:

$$O_i = \{(O_{WS})_i, (O_{AS})_i, (O_{DB})_i\} \tag{8}$$

where:

O_{WS} – service on a WWW server,

O_{AS} – service on an application server,

O_{DB} – service on a database server.

We can calculate time for each type of service:

$$C(O_{\text{all}})_i = C(O_{WS})_i + C(O_{AS})_i + C(O_{DB})_i \tag{9}$$

where:

$C(O_{\text{all}})_i$ – total time of servicing for clients of MK_i type,

$C(O_{WS})_i$ – time of servicing for client of MK_i type on a WWW server,

$C(O_{AS})_i$ – time of servicing for clients of MK_i type on an application server,

$C(O_{DB})_i$ – time of servicing for clients of MK_i type on a database server.

The total servicing time must be lower than the imposed time limit (time out):

$$C(O_{\text{all}})_i < D_i \tag{10}$$

Completing this condition does not mean completing time limits for a particular type client.

Individual servicing times must meet the following dependences:

$$C(O_{WS})_i < D(O_{WS})_i \tag{11}$$

$$C(O_{AS})_i + C(O_{DB})_i < D(O_{AS+DB})_i \tag{12}$$

Whereas individual time limits must meet the following conditions:

$$D(O_{WS})_i \leqslant D_i \tag{13}$$

$$D(O_{AS+DB})_i \leqslant D_i \tag{14}$$

For instance, if time limit $D_i = 4$ time units, and individual $D(O_{WS})_i = 3$ and $D(O_{AS+DB})_i = 5$, then the above dependences are not met.

A function modeling arrival of a particular type of clients' queries depends on the client type and can be described through:

$$F_i = (f_i, \lambda_i, p_i) \tag{15}$$

where:

f_i – distribution of probability of clients' arrivals (determinative, Poisson, etc.),

λ_i – average intensity of arrivals,

p_i – second parameter of distribution (if exists).

In many cases it is necessary for IC to be served in the allotted period of time. Checking if the time limits are met by IC can be carried out in certain cases with the application of time analysis [13].

The most typical distributions analyzed within the theory of mass service are:

- determinative distribution,
- Poisson distribution.

For the determinative distribution the process of request arrivals is precisely determined, and the time intervals between consecutive arrivals T_i are constant.

$$\sum_{i=1}^{m} \frac{C_i}{T_i} \tag{16}$$

where:

C_i – time of task execution,

T_i – period of task occurrence, $T_i = D_i$,

D_i – time limits of the task.

For exponential Poisson distribution [4, 7] the probability of n arrivals in any interval of t length is:

$$P_\lambda(x = n) = \frac{(\lambda t)^n}{n!} e^{-(\lambda t)} \tag{17}$$

for $n = 0, 1, 2, \ldots$
where:

x – random variable (amount of tasks in t time),

λ – intensity of task arrivals.

Average arrival intensity $\lambda_i = 1/T_i$, where T_i is an interval between consecutive arrivals.

System clients (tasks) appear in accordance with Poisson distribution and the time of awaiting them is exponential distribution (continuous).

5. Example of an application of a model of internet system clients

Figure 1 shows two typical IS clients. The first one only browses through WWW sites (superficial), therefore he is not a client using the entire system, but only its first layer. The second one – the actual client uses most of the functions provided by the whole system.

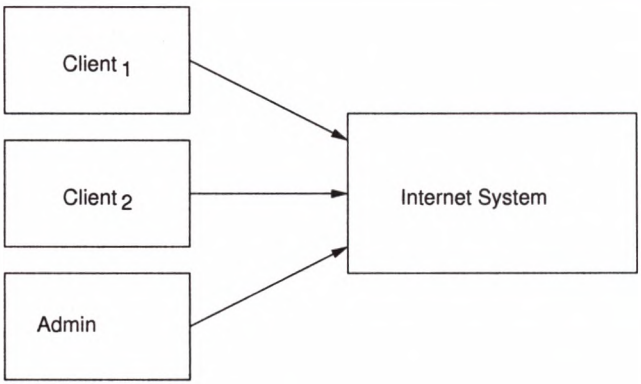

Fig. 1. Exemplary IS model

$$TK_3 = \{(MK_1)_3, (MK_2)_3, (MK_3)_3\}$$

where MK_1 – a model for a client only browsing through internet sites of the system;

$$MK_1 = (O_1, D_1, F_1),$$
$$O_1 = (O_{WS})_1,$$
$$C(O_1)_1 = C(O_{WS})_1,$$

$$C(O_{WS})_1 < D(O_{WS})_1,$$
$$D(O_{WS})_1 \leqslant D_1,$$
$$F_1 = (f_1, \lambda_1);$$

where:

f_1 – Poisson distribution;

MK_2 – a model for a client browsing the entire system;

$$MK_2 = (O_2, D_2, F_2),$$
$$O_2 = \{(O_{WS})_2, (O_{AS})_2, (O_{DB})_2\},$$
$$C(O_2)_2 = C(O_{WS})_2 + C(O_{AS})_2 + C(O_{DB})_2,$$
$$C(O_{WS})_2 < D(O_{WS})_2, C(O_{AS})_2 + C(O_{DB})_2 < D(O_{AS+DB})_2,$$
$$D(O_{WS})_2 \leqslant D_2, D(O_{AS+DB})_2 \leqslant D_2,$$
$$F_2 = (f_2, \lambda_2);$$

where f_2 – Poisson distribution.

The third client (Fig. 1) could be the system administrator. He is a "client" who is actually similar to the second type of a client presented earlier, with the difference that this user uses another part of the system. His service does not require meeting strict time limits and his behaviour in the system can be easily predicted and defined, because he is not a strange user, who could be unforeseeable. However his actions should have the highest priority due to the necessity of performing certain administration tasks.

$$MK_3 = (O_3, D_3, F_3),$$
$$O_3 = \{(O_{WS})_3, (O_{AS})_3, (O_{DB})_3\},$$
$$C(O_3)_3 = C(O_{WS})_3 + C(O_{AS})_3 + C(O_{DB})_3,$$
$$C(O_{WS})_3 < D(O_{WS})_3, C(O_{AS})_3 + C(O_{DB})_3 < D(O_{AS+DB})_3,$$
$$D(O_{WS})_3 \leqslant D_3, D(O_{AS+DB})_3 \leqslant D_3,$$
$$F_3 = (f_3, \lambda_3);$$

where f_3 – determinative distribution.

With simplified assumptions stating that the time of task arrival distribution is determinative, it is possible to check if the time limit condition has been met for all tasks. Referring to the example, let's take: 100 clients of the first type (superficial – τ_1), 30 clients of the second type (overall – τ_2) and two administrators (overall – τ_3).

If we assume that τ is a set of all m independent periodical tasks $\tau = \tau_1, \tau_2, \tau_3$, with C_i execution time and T_i occurrence period and taking the assumption that time limit $D_i = T_i$ meets the condition:

$$\sum_{i=1}^{m} \frac{C_i}{T_i} \leqslant i(2^{1/i} - 1) \tag{18}$$

then a set of those tasks meets time limits.

For the example of those three tasks τ_1, τ_2, τ_3 with the assumption that $D_i = T_i$ we obtain the following results (Tab. 1).

Table 1
Example for tree tasks τ_1, τ_2, τ_3

Tasks [i]	Task occurrence period T_i [ms]	Task execution time C_i [ms]	Time limits of task D_i [ms]	System load factor $\frac{C_i}{T_i}$	System load factor $\sum_{i=1}^{m} \frac{C_i}{T_i}$	Value of resources use factor $i(2^{1/i} - 1)$
τ_1	20	5	20	0.25	0.25	1
τ_2	50	20	50	0.4	0.65	0.828
τ_3	200	100	200	0.5	1.15	0.779

The presented example shows that the time limit condition (18) is met only for the first two tasks $\{\tau_1, \tau_2\}$. Adding another task τ_3 causes increase of load and condition (18) is not met, which means that the system does not have any further capabilities of processing next tasks.

6. Summary

Modelling IS clients service is complex. For instance we do not know how clients will behave. They may use the system for a longer (indefinite) period of time and then overload it with excessive number of queries at any time. Such a problem was analyzed in Koucheryavy's article [9] – it concerns overloading a WWW server during the 1998 International Football Championship in France. Mathematical modeling of IC enables creation of IS which will ensure a specified level of service. It is not possible to create one perfect model, however, it is possible to create it depending on customers' requirements. The problem of determining the characteristics of a particular system and setting the parameters of a model is connected with knowledge of the client, that is why it is so important to learn the clients' behaviour patterns.

As follows from the simple example in section 5, the introduced IC model can join with the general IS model in easy way. This example takes an assigenment of clients to definite classes. In special cases it is necessary to extend this model with additional parameters. Further work will focus on modelling queries with the use of exponential distribution which, despite considerable simplification, can (as it is supported by many years' experience) imitate reality in a good way.

Acknowledgments

Thanks to Prof. Jan Werewka for help to write this article.

References

[1] Abdelzaher T., Lu C.: *Modeling and Performance Control of Internet Servers.* [in:] IEE Conference on Decision and Control, Sydney, Australia, December 2000

[2] Andersen D., Yang T.: *Adaptive Scheduling with Client Resources to Improve WWW Server Scalability.* UCSB Tech Report, TRCS96-27
http://www.cs.ucsb.edu/TRs/techreports/TRCS96-27.ps.

[3] Banga G., Druschel P.: *Measuring the Capacity of a Web Server.* [in:] Proceedings of USITS, December 1997
http://www.cs.rice.edu/CS/Systems/Web-measurement/paper/paper.html

[4] Bazewicz M.: *Własności i funkcje sieci komputerowych.* Cz. II: Komunikacja. Wrocław, Politechnika Wrocławska 1980

[5] Czachórski T.: *Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych.* Gliwice, PKJS 1999

[6] Ferrari D., Remaekers J., Ventre G.: *Client-Network Interactions in Quality of Service Communication Environments.* IFIP Transactions C (Comunication Systems), vol. C-14, 1993, 221–234

[7] Filipowicz B.: *Modele stochastyczne w badaniach operacyjnych.* Warszawa, WNT 1996

[8] Firoiu V., Boudec J. Towsley D., Zhang Z.: *Theories and Models for Internet Quality of Service.* Proceedings of IEEE, May 2002

[9] Koucheryavy Y., Krendzel A.: *Analysis of Web Traffic and Users' Behaviour Modeling During Busy Hour.*
http://wwwtgs.cs.utwente.nl/Docs/eunice/summerschool/papers/paper1-2.pdf

[10] Orkisz K., Rak T.: *Port wielki jak świat (Port równoległy – bezpośredni dostęp).* CHIPSpecial, vol. October/2001, 53–55

[11] Rak T.: *QoS w systemach klastrowych.* Zeszyty Naukowe Politechniki Rzeszowskiej, Rzeszów, 2002

[12] Werewka J.: *Analiza czasowa systemów czasu rzeczywistego – Analiza i projektowanie systemów komputerowych czasu rzeczywistego o różnym stopniu rozproszenia.* Kraków, Wyd. AGH 1991

[13] Werewka J.: *Projektowanie symulacji systemów – symulacja systemów zdarzeń dyskretnych.* Kraków, Wyd. AGH 1989