One model for managing incoming requests of web based educational applications.

Authors:

Artem V. Joukov, Irina V. Aminova

Main goal:

To provide necessary quality of service of web-based educational application

Some characteristics of web-based educational application

Type of content

- Lecture materials
- Tests for students' self-control
- Vocabularies
- Tests to control educational level of students by lecturer
- Online chats and offline conferences for interactive communications between students and lecturers
- Different types of reports

Complexity type of different services

- Easy (lecture materials)
- Medium (any type of test)
- Hard (reports)

Type of service priority (Type of service priority can be defined by administrator, to show that service from list is most demanded at the moment)

- Obligatory
- Secondary

Functional model

Diagram 1 demonstrates how web based application works

- No permanent connection
- User works with web-based application via clients – "browsers". One user can use several clients.
- Server executes necessary functions then receives requests from clients.
- Server sends answers after completing requested functions

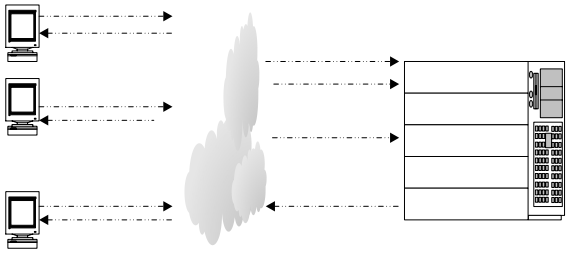

Diagram 1. Scheme of organization of web-based education application at one server

Mathematical model to provide maximum performance for specific services (QoS for web-based educational application)

There are M resources (named service channels in queuing theory). The server receives queries (customers) from Internet/Intranet clients to provide necessary resources.

There are M types of queries: $K_1,\ldots,K_M$. Queries of type K_i have service at R_i resource, $i \in 1,\ldots,M$.

Server doesn't have queue buffer. Arrived query is serviced immediately without waiting in queue. There are limits for queries service times. Namely, queries service time at resource i is limited by $T_i, i \in \overline{1,M}$. If the time limits are reached but the request isn't served successfully then the execution of query will be terminated and error message will be sent to the client.

Every resource i can be characterized as set of $R\{S_i, \Pr_i, T_i\}$, where

S_i – Type of complexity. It is a quantity of commands which are executed at system while request is i served. It can be named as "length of request";

$\Pr_i$ – Relative priority of resource i, $0 \le \Pr_i \le 1$. $\Pr_i > \Pr_j$ means, that resource i has higher priority then resource j;

λ_i – Intensity of incoming queries stream of resources i coming from the clients. We consider different streams as independent, $i = 1,\ldots,M$.

V – Productivity of server. This is number of transactions which can be completed at server per time.

The goal is to organize queries service to minimize the number of refuses.

We claim that queries of type K_i have the priority of service $\Pr_i$.

Let's denote $x_i = 1$, if the query of type K_i is admitted for service at the resource i, $x_i = 0$, otherwise, $i = 1,\ldots,M$.

Model for managing web based educational applications:

$$f(x_1,..,x_M) = \sum_{i=1..M} x_i \lambda_i \Pr_i ;$$

$$f(x_1,\ldots,x_M) \to \max$$

$$\sum_{i=1}^{M} x_i \lambda_i S_i \le V ,$$

$$x_i \in \{0,1\}, i = 1,\ldots,M$$

Extension of mathematical model to provide maximum performance for specific services for the Web clusters (QoS for web-based application)

The restriction of the incoming queries is not the best way to provide quality service. There is an opportunity of extension numbers of web servers to provide access to the web based education application. Situation with cluster system can be managed in the same way. We can redirect request from a busy server to a free one or restrict requests if there are no free servers at cluster.

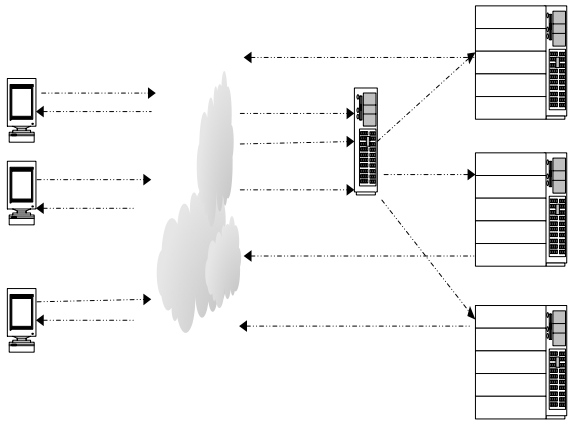

Diagram 2: Scheme of organization of web-based education application at web cluster

Let's suppose that the cluster is a set of N servers, with productivity of each server $S^j; j \in [1; N]$;

Every server has the same web based educational application. Each application is a set of the same number of resources.

Let's suppose that the type of complexity is S_i for each service.

Let's name the intensity of incoming queries stream of resources i coming from the clients as λ_i.

Priority of resource i is named as $\Pr_i$

Let's denote $x_{ij} \in \{0,1\}$

$x_{ij} = 1$ If query of **j** class is served at server I.

$$f(X) = \sum_{ij} x_{ij} \Pr_i \lambda_i$$

$$f(X) \to \max$$

$$\sum_{ji} x_{ij} \lambda_i S_i \le S^j$$