

МЕТОД ЗГЛАДЖУВАННЯ ВХІДНОГО НАВАНТАЖЕННЯ НА СЕРВЕР ON-LINE ТАРИФІКАЦІЇ

© Скулиш М. А., 2014

Запропоновано метод згладжування вхідного потоку заявок на сервер тарифікації, що враховує вимоги до ресурсів кожного з етапів обслуговування заявок, що дасть змогу рівномірно навантажити ресурси системи протягом часу обслуговування заявки.

Ключові слова: система керування потоками, ресурси сервера, процес обслуговування, система тарифікації.

М.А. Skulysh

National Technical University of Ukraine “Kiev Polytechnic Institute”

METHOD OF SMOOTHING INPUT LOAD ON THE ONLINE CHARGING SYSTEM

© Skulysh M.A., 2014

Charging users for a server operator billing involves performing a number of operations. Each service provided by the operator or service provider customers, charged at a separate scheme, as the bottleneck in the process of providing services are bilingual system. Operation performed is limited. In case of exceeding its application is considered lost and the user is notified of the inability to obtain services. In this situation, the operator has monetary losses. In systematic failures in service deteriorates reputation. Therefore, it is important to make a control input streams so as to reduce the possibility of exceeding the service time of the user's requests. The process of managing incoming applications to the server should consider charging the features of charging large companies:

- charging process involves a sequence of standard operations, speed and resources for the implementation of which depends on the type of service that serviced
- a variety of services that are billed differently,
- extremely high number of requests for billing,
- irregularity of the input stream, applications received according to Poisson.

Scheme smoothing the input load is a set of values of the maximum number of requests (the sequence $\{k_i\}$), arriving at the input of the system in a short time interval Δt_i in a given sequence. Number of elements in a sequence of n is chosen so that the equation was carried out $t = \sum_{i=1}^n \Delta t_i$, where t - the average time of the system. It's necessary to choose a sequence $\{k_i\}$ to satisfy conditions:

1. Applications that are both served by the system must use the volume V of resources close to the total maximum number of resource V_{max} . Dispersion sequences such amounts shall be minimized.

2. Dispersion elements of the sequence $\{k_i\}$ should be minimal.

The method of solution of choice sequence $\{k_i\}$ by using genetic algorithm. When using the proposed scheme smoothing the input load is provided by the maximum allowable flow uniformity requests. Since overlooked the sequence of operations performed on the application server to the mobile operator, the amount of resource that is required for these operations. Selected sequence of maximum permissible values of number of applications coming into the system in a short period of time.

Key words: flow management system, server's resources, service processing, charging system.

Обслуговування абонентів на сервері тарифікації є необхідною складовою частиною процесу обслуговування абонентів та надання їм телекомунікаційних послуг. Оскільки кожна послуга є платною, оператор зв'язку залежно від типу тарифного плану здійснює тарифікацію заявки або перед наданням послуги в режимі реального часу, або в режимі offline має перевірити наявність коштів на рахунку абонента, зробити перерахунок, тобто виконати декілька стандартних операцій.

Тарифікація абонентів на стороні оператора зв'язку передбачає виконання ряду операцій. Кожна послуга, яку надає оператор або провайдер послуг абонентам, тарифікується за окремою схемою, тому вузьким місцем у процесі надання послуг є білінгва-система.

Час виконання операцій є обмеженим, у разі його перевищення заявка вважається втраченою і абонент отримує повідомлення про неможливість отримання послуги. У такій ситуації оператор зазнає збитків, а у разі систематичних відмов у наданні послуг погіршується репутація компанії. Тому важливо здійснити керування вхідним потоком так, щоб зменшити ймовірність перевищення часу обслуговування абонентських заявок.

Процес керування вхідним потоком заявок на сервер тарифікації має враховувати особливості систем тарифікації великих компаній:

- процес тарифікації передбачає послідовність стандартних операцій, швидкість та ресурси для виконання яких залежать від типу сервісу, який обслуговується;

- різноманітність сервісів, що по-різному тарифікуються;

- надзвичайно велика кількість запитів на тарифікацію;

- нерівномірність вхідного потоку, заявки надходять за законом Пуассона.

Розглянемо детальніше кожну особливість.

Схему обслуговування заявок на сервері тарифікації показано на рис. 1. Заявки надходять на сервер за різними протоколами (Diameter, CAP2, MAP), у відповідних модулях EDP та RES (Enhanced Diameter Proxy) розподіляються між пулом адаптерів відповідних протоколів, після чого декодуються у відповідних адаптерах CPA, DPA (CAP Protocol Adapter та Diameter Protocol Adapter) та зводяться до єдиного вигляду.

Розкодовану справу модуль передає на сервер бізнес-логіки SEE (Service Execution Environment) через модуль маршрутизації BUS. Сервер бізнес-логіки SEE є ядром системи тарифікації та забезпечує середовище для виконання послідовності операцій, які передбачає процес обслуговування заявок.

У послідовність операцій, які потрібно виконати для успішного обслуговування заявки, входять такі дії:

1. Вилучення інформації про абонента. Для цього модуль SEE звертається до бази даних керування абонентами (CM DB Customer Management data base).

2. Вилучення інформації про місце розташування абонента. Для цього модуль SEE звертається до бази даних, що зберігає структуру мережі (RE DB Resource Inventory data base).

3. Вилучення інформації про стан рахунку абонента. Для цього модуль SEE звертається до бази даних рахунків абонентів (Balance Storage).

4. Здійснюється розрахунок вартості послуги на основі тарифної книги та поточної тарифної моделі абонента. Для цього модуль SEE звертається до модуля розрахунків (RE – Rating Engine).

5. З рахунку абонента знімається плата за послугу, залежно від типу сервісу паралельно може здійснюватися резервування заданої суми, тільки для сервісів з резервацією (SCUR – Session Charging with Unit Reservation та ECUR – Event Charging with Unit Reservation), коли тривалість надання послуги невідома та не можна провести остаточний розрахунок. Для здійснення операцій дебету та резервації коштів модуль SEE звертається до бази даних рахунків абонентів (Balance Storage).

6. Формується звіт про надання послуги CDR (call data record). Для цього модуль SEE звертається до програмного модуля формування CDR (TTS – Toll Ticket Server).

7. Якщо це передбачено послугою, відправляється повідомлення абоненту про результати наданих послуг. Для цього модуль SEE звертається до модуля для надсилання повідомлень абоненту (MG-message GateWay).

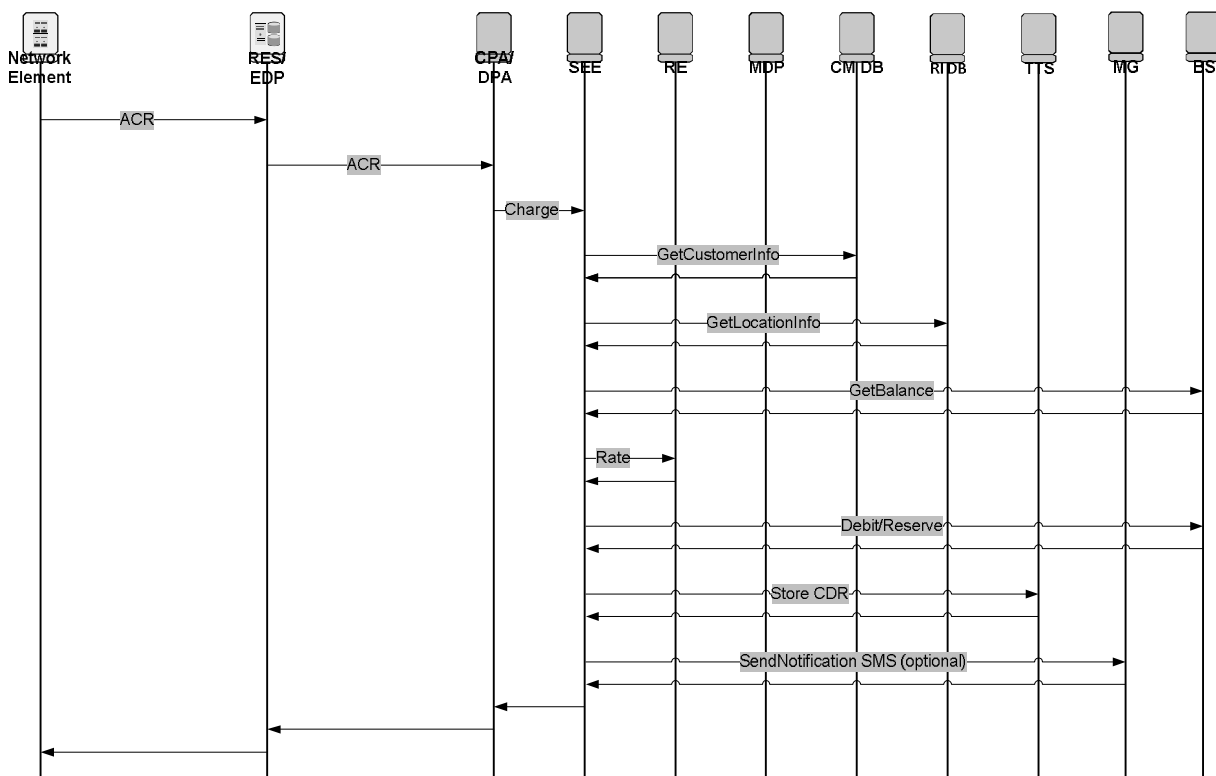


Рис. 1. Загальна схема online тарифікації послуг на сервері тарифікації

Процес тарифікації завершено.

Як видно, процес тарифікації є багатоступінним, а операції, які послідовно виконуються у ядрі бізнес-логіки SEE із залученням різних підсистем, різноманітні, відповідно потребують різної кількості оперативної пам'яті, процесорного часу та дискового простору. Розв'язуючи задачу керування вхідним потоком, необхідно звернути увагу на час використання ресурсів. Необхідно врахувати як загальну кількість ресурсів, що обслуговує увесь сервер, так і розділення ресурсів (методами віртуалізації), які, з одного боку, забезпечують ефективне обслуговування кожного етапу обробки, а з іншого – є обмеженням, оскільки підсистема використовує лише ресурси, їй відведені, та не має доступу до інших ресурсів. Також необхідно врахувати розподіл середнього часу виконання операцій.

Друга особливість полягає у тому, що кожний тип послуг, незважаючи на стандартність операцій, які виконуються в процесі тарифікації, потребує для здійснення розрахунків різної кількості ресурсів.

Стосовно необхідної процедури обслуговування всі сервіси можна розділити на три групи:

- Сесія тарифікації з резервацією (SCUR – Session Charging with Unit Reservation) оперативна пам'ять зайнята на всю тривалість сесії (може тривати до доби – наприклад, GPRS).
- Моментальна тарифікація події (IEC Immediate Event Charging) не зберігає стан свого виконання в пам'яті – виконується оцінка та списання грошових коштів у один момент (SMS).
- Тарифікація події з резервацією (ECUR – Event Charging with Unit Reservation) – оперативна пам'ять зайнята на період резервації (наприклад, час доставки контенту абоненту: відео, музика, MMS).

Отже, сервіси SCUR та ECUR виконуються у декілька етапів. Стан заявки або стан виклику зберігається у підсистемі MDP (Memory DataBase Provider). MDP – це модуль для збереження поточного стану, що являє собою програмно-апаратний комплекс, який забезпечує швидкий доступ до оперативної пам'яті (запис, зчитування, пошук).

На рис. 1 наведено загальну схему online-тарифікації. Всі сім етапів одноразово проходять заявки групи сервісів IEC. Під час обслуговування сервісів SCUR та ECUR перший та другий етапи, що передбачають вилучення інформації про абонента та про місце його розташування, виконуються

один раз, після чого вся інформація про абонента та стан заявки-виклику зберігається у підсистемі MDP.

Ресурси, яких потребує система для обслуговування семи етапів, залежать не тільки від групи сервісів, але й від їх типу. Відмінність у швидкості та ресурсозатратності операцій виникає під час розрахунку вартості послуги та вилучення інформації про стан заявки-виклику із системи MDP. Ці умови мають бути враховані під час розрахунку плану керування вхідним потоком на сервер тарифікації.

Третя особливість полягає в тому, що одночасно надходить велика кількість заявок на тарифікацію різних типів ресурсів.

Сьогодні оператори мобільного зв'язку надають послуги мільйонам абонентів, наприклад, компанія "Київстар" обслуговує до 26 мільйонів абонентів. При цьому за рахунок привабливих пакетних умов все більше абонентів користуються послугами мобільного Інтернету та дзвінків. За умови централізованого обслуговування система тарифікації одночасно обслуговує до одного мільйона абонентів, які замовляють послуги або продовжують користуватися ними. Для кожної заявки абонента ініціюється ланцюг операцій, описаних вище. У періоди найбільшого навантаження кількість абонентських заявок збільшується у декілька разів.

Четвертою особливістю є неоднорідність вхідного потоку заявок. Системи обслуговування абонентів прийнято розглядати як системи із пуассонівським вхідним потоком заявок, основними особливостями якого є значна дисперсія кількості заявок, що надходять на тарифікацію. Для розподілу Пуассона дисперсія дорівнює математичному очікуванню. Тобто можливі сплески навантаження на короткий період часу, що є меншим за час обслуговування заявки на сервері тарифікації. Такі сплески призводять до тимчасового перевантаження сервера навіть не у час пік.

Стосовно реалізації логіки процесу обслуговування роботу підсистем сервера оператора мобільного зв'язку можна представити як багаторівневу систему масового обслуговування, де керування потоком заявок здійснюється на двох рівнях.

Перший – логічний рівень прикладних програм. Тут заявки, які надійшли у систему, розділяються за типом сервісу, який вони представляють, обслуговування черг ведеться відповідно до схеми обслуговування, розробленої для відповідного типу сервісу. Процес керування передбачає формування черг за типом сервісу, застосування методів групи WRAD, а також інших схем керування, які враховують специфіку обслуговування сервісів. Отже, система масового обслуговування першого рівня являє собою заявки різних типів сервісів, які надходять на обслуговування до системи. Називатимемо їх заявками першого рівня. Обслуговуючими пристроями у такій системі є ланцюги функціональних блоків, у яких здійснюється послідовне обслуговування заявок, кожний тип сервісу обслуговується в окремому ланцюзі.

Другий рівень – рівень технічної обробки. Схема обслуговування заявок за типом сервісу передбачає послідовне виконання операцій, для виконання яких потрібна задана кількість апаратних ресурсів, кожену операцію можна представити як заявку на обслуговування. Розглянемо заявки другого рівня, де обслуговуючими пристроями є апаратні ресурси. Тут заявки другого рівня організовуються у черги до відповідних ресурсів. Політики використання ресурсів визначаються методами керування ресурсами обчислювальної системи. Архітектури розподілу ресурсів, організація обслуговування заявок другого рівня суттєво впливають на швидкість обслуговування. Однак така архітектура системи обробки заявок другого рівня є постійною, про її роботу можна робити висновки за статистичними даними затримок в обслуговуванні заявок першого рівня.

Оскільки вхідний потік заявок другого рівня однозначно визначається кількістю заявок першого рівня, що обслуговуються у системі, то система керування вхідним потоком заявок першого рівня, яка побудована з урахуванням статистики завантаженості ресурсів системи другого рівня, дасть змогу зменшити втрати заявок через затримки, пов'язані з нестачею ресурсів.

Архітектуру дворівневої системи керування зображено на рис. 2.



Рис. 2. Схема обслуговування заявок на сервері оператора мобільного зв'язку

Постає питання: як організувати роботу системи керування вхідним потоком заявок, щоб потік заявок другого рівня був якомога рівномірнішим.

Обслуговування заявок першого рівня у функціональних блоках породжує потік заявок другого роду, для виконання яких використовується задана кількість ресурсів сервера. Отже, якщо у деякому функціональному блоці обробляється одночасно велика кількість заявок першого роду, при цьому заявки другого роду, породжені відповідним функціональним блоком, потребують для свого виконання значної кількості ресурсів, тоді може виникнути проблема браку ресурсів сервера, що призведе до затримки в обслуговуванні заявок першого роду, і, як наслідок, перевищення допустимого часу обслуговування, втрати заявок, зниження якості обслуговування абонентів.

У [1] запропонована стратегія керування вхідним потоком заявок, яка полягає в тому, щоб не допускати двох сплесків навантаження протягом часу обслуговування заявок у ресурсозатратних функціональних блоках. Запропонований метод передбачає відстеження піків навантаження та введення затримки для деякої частини заявок другого піка, що дозволяє уникнути перевантаження ресурсів сервера. Однак реалізація цієї стратегії та запропонованого методу потребує постійного моніторингу роботи сервера, а на практиці з метою заощадження ресурсів система моніторингу не завжди активна через брак ресурсів, тому відслідковування моментів надходження надлишкової кількості заявок не завжди можливе. Саме тому доцільно розробити *схему згладжування вхідного навантаження*.

Схема згладжування вхідного навантаження являє собою набір значень максимально допустимої кількості заявок (послідовність $\{k_i\}$), що надходять на вхід системи за малий інтервал часу Δt_i у заданій послідовності. Кількість елементів послідовності n підбирають так, щоб виконувалось рівняння

$$t = \sum_{i=1}^n \Delta t_i,$$

де t – середній час перебування заявки першого рівня у системі.

Необхідно підібрати таку послідовність $\{k_i\}$, щоб виконувались дві умови:

1. Заявки, які одночасно обслуговуються у системі, повинні використовувати об'єм ресурсів V , близький до загальної максимально можливої кількості ресурсу V_{max} . Дисперсія послідовностей таких об'ємів має бути мінімальною.

2. Дисперсія елементів послідовності $\{k_i\}$ повинна бути мінімальною.

Тривалість перебування заявки першого рівня у функціональних блоках (ФБ) є випадковою величиною, що залежить від швидкості обробки породжуваних у цьому ФБ заявок другого рівня. Спираючись на середньостатистичні значення, отримані системою моніторингу, вважатимемо, що час перебування заявки у функціональному блоці відомий (t_j , де j – номер ФБ). $t = \sum_{j=1}^m t_j$, де m – кількість функціональних блоків у системі.

Відома кількість ресурсу (v_j , де j – номер ФБ), яка необхідна для обслуговування заявок другого рівня, породжуваних заданим ФБ.

Як розраховувати об'єм ресурсу V , який використовується у поточний момент часу, показано в [2], основний принцип полягає в тому, що використовується зворотна система відліку часу. Прийmemo за нульовий час закінчення обслуговування заявки, тобто $t^0=t$, далі позначимо періоди часу, коли заявки переходять між функціональними блоками: $t^1 = t^0 - t_1, \dots, t^j = t^{j-1} - t_j, \dots, t^m = t^{m-1} - t_m = 0$. Всі заявки, які надійшли протягом інтервалу $[t^1, t^0]$ в момент часу t^0 , обслуговуються у першому функціональному блоці. Заявки, які надійшли до системи протягом інтервалу $[t^j, t^{j-1}]$ в момент часу t^0 , обслуговуються у j -му функціональному блоці.

Отже, об'єм ресурсу V^0 , що використовується у момент часу t^0 , – це сума об'ємів ресурсу v_j^0 , зайнятого заявками, які перебувають у j -му ФБ ($j = \overline{1, m}$) в момент часу t^0 .

$$V^0 = \sum_{j=1}^m v_j^0$$

Значення v_j^0 залежить від кількості заявок, які надійшли до системи протягом часу $[t^j, t^{j-1}]$, та визначається як добуток кількості заявок на об'єм ресурсу, що необхідний для обслуговування однієї заявки у відповідному функціональному блоці. Оскільки застосовується *схема згладжування вхідного навантаження*, то максимально допустима кількість заявок, які припадають на цей інтервал часу, відома. v_j^0 – це добуток кількості заявок, що у момент часу t^0 перебувають у j -му ФБ, на об'єм ресурсу, який потрібен для обслуговування заявок другого рівня, породжених в j -му ФБ.

Задля забезпечення ефективного згладжування необхідно, щоб умова 1 виконувалася не тільки для об'єму V^0 , але й для всіх V^i , ($i = \overline{1, n}$).

Цільова функція має три складові:

- Дисперсія елементів послідовності $\{k_i\}$ повинна бути мінімальною.
- Дисперсія елементів послідовностей $\{V^i\}$ мінімальна.
- Середнє значення елементів послідовності $\{V_j\}$ прямує до максимально можливої кількості ресурсу V_{max} заданого типу, що виділяється для обслуговування заявок вибраного типу сервісу.

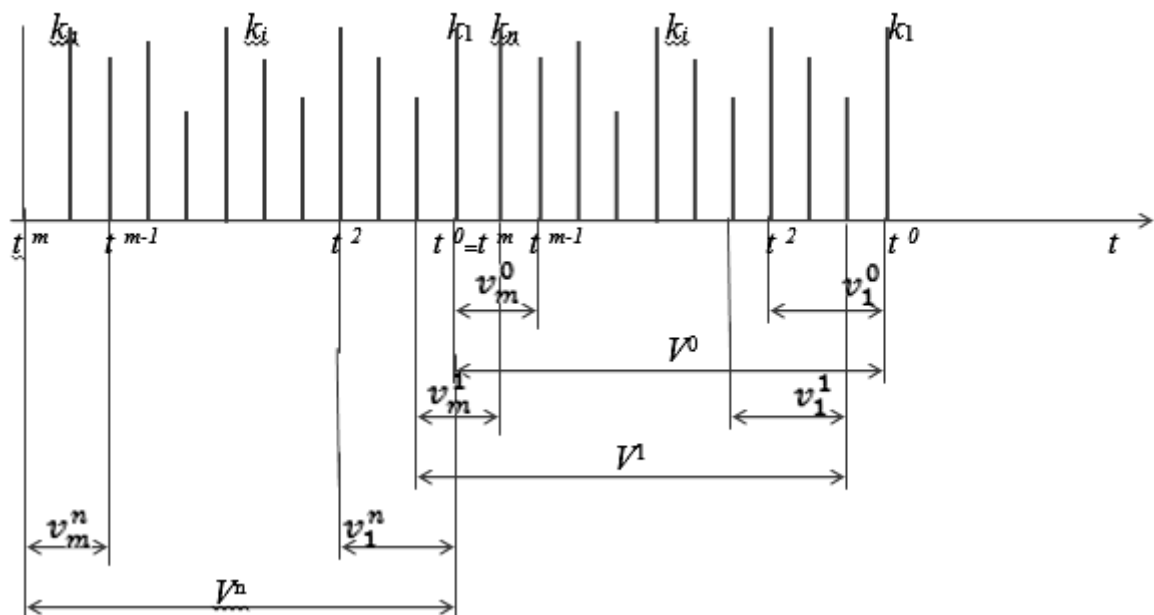


Рис. 3. Схема згладжування вхідного навантаження, з урахуванням об'єму ресурсу, що використовується

Методом розв'язання вибір послідовності $\{k_i\}$ здійснюється за допомогою генетичного алгоритму, так, щоб задовольнялися умови 1 і 2.

Геномом виступають елементи послідовності $\{k_i\}$.

Кросовер: зміна значень елементів послідовності $\{k_i\}$.

Завершення алгоритму:

- за часом;
- кількістю розглянутих поколінь;
- виродження популяції.

В результаті отримуємо послідовність.

У разі використання запропонованої схеми згладжування вхідного навантаження забезпечується максимально допустима однорідність потоку заявок другого рівня, оскільки береться до уваги послідовність операцій, які виконуються із заявкою на сервері мобільного оператора, об'єму ресурсу, що потрібен для забезпечення цих операцій. Підбирається послідовність максимально допустимих значень кількості заявок, що надходять у систему за малий інтервал часу. Критеріями оцінки є наближеність загального об'єму, що використовується, до максимально допустимого. При цьому забезпечується мінімальне середньоквадратичне відхилення від середнього значення кількості заявок, що пропускаються у систему. Фактично наведено спосіб керування чергою заявок, які надходять на сервер оператора.

1. Globa Larysa, Skulish Maria, Reverchuk Andrei. Server resources load monitoring serving different types of services // International Conference TCSET'2014 – February 2014. 2. Globa Larysa, Skulish Maria, Reverchuk Andrei. Manage of incoming application flow to prevent shortage of server resources // International Conference TCSET'2014 - February 2014. 3. Totok, V. Karamcheti. Exploiting Service Usage Information for Optimizing Server Resource Management // ACM Transactions on Internet Technology, Vol. 11, No. 1, Article 1, Publication date: July 2011.