

О. В. КАСІЛОВ, канд. техн. наук, доцент, НТУ «ХПІ»;
М. Ф. СТРУЧАЄВ, студент, НТУ «ХПІ»

РОЗРОБКА УНІВЕРСАЛЬНОЇ БІЛІНГОВОЇ СИСТЕМИ

У статті розглядаються особливості побудови універсальної білінгової системи для сфери телекомунікацій і зв'язку.

The article deals with peculiarities of constructing a universal billing system for telecommunications and communications.

Області використання білінгових систем можуть бути самими різноманітними. Найсприятливішою для розвитку і поширення білінгових систем виявилася сфера телекомунікацій і зв'язку.

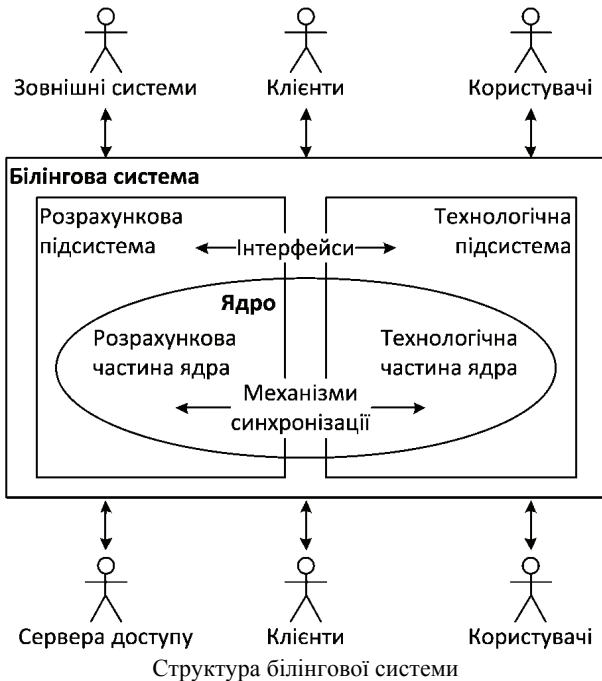
Розробка білінгових систем до недавнього часу була дуже спеціалізованим напрямком розробки ПЗ. Не дивлячись на те, що ця область була в деякій мірі вивчена, останніми роками сталися серйозні зміни, які кардинально вплинули на вигляд систем розрахунків. В першу чергу це пов'язано з дуже стрімким розвитком інтернет-сервісів, різних видів телефонії і послуг передачі даних. З появою цих технологій почав зростати і розвиватися ринок телекомунікаційних послуг, виникла необхідність в білінгових системах, які займаються обліком надання, споживання і оплати цих послуг [1].

Про зростаючий інтерес до проблем розробки білінгових систем в нашій країні і за кордоном свідчать поява публікацій на тему білінга в ІТ-пресі [2, 3].

Як в будь-якій досить молодій галузі знань, у сфері розробки білінгових систем є ряд невирішених проблем. Зокрема, недостатньо розроблена теоретична база побудови білінгових систем, не вирішено питання конвергентності білінга, тобто його можливості охоплювати і обробляти універсальним чином будь-які послуги [4, 5].

Розробка білінгової системи повинна вестися з урахуванням принципу універсальності. Передбачається, що послуги, які можуть додаватися в білінгову систему, комбінуватися і змінювати свої характеристики в процесі розвитку технологій, повинні оброблятися нею абсолютно однотипно. Це дозволить розширювати сервісні функції білінгової системи без додавання нових модулів, а також дасть можливість об'єднувати тарифікацію і оплату різних послуг у рамках однієї системи.

Структура системи. Проектовану систему необхідно розділити на дві підсистеми (див. рисунок). Одна з цих підсистем проводитиме облік клієнтів і відповідатиме за розрахунки з ними (розрахункова підсистема) друга взаємодіятиме з технологічними серверами і включатиме елементи технології послуг (технологічна підсистема), що надаються.



Розрахункова підсистема відповідає за облік грошей, контроль за наданням послуг залежно від грошей і обробку даних статистики, результати якої відображаються на можливості подальшого надання послуг.

Технологічна підсистема відповідає за управління даними, необхідними для надання послуг і збір статистики по наданих послугах.

Пропонується розділити розрахункову і технологічну частини з наступних причин:

1. Для розрахункової підсистеми не важливий спосіб надання послуги, її характер і сенс. У її завдання входить лише облік "абстрактного" ресурсу. Це дозволяє модифікувати і налаштувати технологічну підсистему без змін в розрахунковій підсистемі. Тобто технологічні нововведення, що відбуваються з розвитком інформаційних сервісів, не призводять до написання нового коду або ускладнення структури бази даних.

2. Для технологічної підсистеми не важливий спосіб обліку грошей, несуттєво, у рамках якого тарифу надається послуга, скільки вона коштує. Тому зміна прейскуранта, введення нових платіжних планів, поява альтернативних форм оплати і тому подібне приведе тільки до зміни розрахункової підсистеми, не торкнувшись роботи технологічної підсистеми.

3. Різні вимоги до підсистем диктують різну їх архітектуру. Наприклад, швидкодія підсистем має бути досягнута по різних критеріях: розрахункова

підсистема повинна забезпечувати швидкодію при великих запитах (один запит обробляє багато записів), а технологічна - при великій кількості одночасних запитів (один запит обробляє 1-2 записи). Для розрахункової підсистеми дуже важлива надійність, але відмовостійкість для неї – не головна вимога. Для технологічної підсистеми локальні втрати даних, навпаки, неприпустимі, але для неї критична відмовостійкість, тому що відмова в наданні сервісу спричиняє за собою втрати в грошах. Отже, технології побудови баз даних і алгоритмів в підсистемах можуть (і повинні) відрізнятися.

4. Вихід з ладу однієї з підсистем не призводить до "підвисання" другої, оскільки підсистеми слабо пов'язані між собою і в той же час самодостатні для виконання своїх завдань. Це служить додатковою страховкою при збоях.

5. У цій моделі розділені дві найбільш ресурсоємні операції: авторизація і облік грошей. Підрахунком грошей і управлінням правами доступу займаються різні процеси, які можуть бути рознесені навіть на різні сервери.

Оскільки білінгова система може бути великим комплексом, що виконує ряд сильно або слабо взаємозв'язаних завдань, виділимо ядро системи як мінімальну функціонально необхідну її частину.

Ядро відповідає за рух потоку даних (data flow) і потоку грошей (cash flow) і реалізує ланцюжок «гроші => послуга => гроші». Ядро складається з частини бази (чи декількох баз) даних і алгоритмів, які здійснюють логіку роботи ядра.

Переваги такого підходу наступні:

1. Визначення мінімальної працюючої конструкції, що забезпечує життєво важливі функції білінгової системи.

2. Можливість нарощування і модернізації модулів без зміни структури ядра.

3. Можливість отримання процесами відомостей безпосередньо з ядра як єдиного достовірного джерела.

Ядро охоплює частину розрахункової підсистеми (розрахункова частина ядра), і частину технологічної підсистеми (технологічна частина ядра). Обидві частини ядра знаходяться в постійній взаємодії. Коректно відобразити дані, уникнути циклів і мінімізувати процес обробки інформації - це головне завдання ядра.

Для синхронізації роботи двох частин ядра використовується механізм синхронізації, що переводить по певних алгоритмах зміни, які сталися в одній частині ядра, до змін в іншій. Цей механізм має бути добре оптимізований з точки зору процесорного часу і потужності, оскільки є найбільш важливим в роботі усєї системи в цілому, відповідаючи за її злагоджену роботу. Швидкодія цього механізму впливає на можливість роботи системи в режимі реального часу.

Окрім основних завдань, описаних вище, білінгова система може вирішувати ряд допоміжних проблем, пов'язаних з характером надання і обліку послуг.

Розрахункова частина ядра відповідає за облік грошей, контроль за наданням послуг, обробку статистики в мінімально необхідному об'ємі. Такі операції як клієнтський облік, бухгалтерський облік, ділерські системи, призначені для користувача і адміністративні інтерфейси і тому подібне не є частиною ядра, вони добудовуються поверх ядра як модулі.

Функції розрахункової частини ядра: реєстрація клієнтів; організація безлічі платіжних планів і їх відображення в ресурси; підписка клієнтів на вибрані ними платіжні плани; прийом, фіксація і обробка платежів різних форм; контроль за наданням послуг; облік наданих послуг, коригування залишків.

Усі ці функції повинні враховуватися при моделюванні ядра, оскільки інакше буде порушений технологічний процес.

В той же час такий набір є достатнім для того, щоб оператор міг робити послуги на комерційній і некомерційній основі.

Технологічна частина ядра – це скелет технологічної підсистеми, мінімальна функціонально осмислена конструкція.

Сама білінгова система не робить клієнтові послуг. Уся інтелектуальна робота по наданню послуги лежить на серверах доступу, які є незалежними програмно-апаратними комплексами, що взаємодіють з білінговою системою для з'ясування достовірності клієнта, що запрошує сервіс, прав цього клієнта, обмежень на отримання сервісу і тому подібне. Сервера доступу можуть передавати в систему білінга інформацію про авторизацію клієнта, отриманні ним тієї або іншої послуги, її об'ємах і характеристиках.

Незалежно від того, як реалізовані сервера доступу, механізми взаємодії їх з білінговою системою є однотипними, і початкові дані для ухвалення відповідних рішень можуть зберігатися в білінговій системі в уніфікованому форматі. Навіть якщо сервер неможливо настроїти на вибраний в системі білінга формат, взаємодія може відбуватися через програми, що перетворюють формати даних. Це дуже істотна властивість для досягнення гнучкості системи.

Формат даних, якими обмінюються технологічна частина ядра і сервера доступу, має бути єдиним. Такий підхід дозволяє додавати нові послуги без узгодження форматів даних на стику «технологічний сервер – технологічна частина ядра», вимагається лише додати в білінг опис характеристик нових ресурсів.

Список літератури: 1. *Sheopard S.* Telecom Convergence: How to Profit from the Convergence of Technologies, Services, and Companies. – McGraw-Hill Professional, 2002. 2. *Волков Д. А.* Интеграция автоматизированных систем в единое информационное пространство / *Д. А. Волков* // Биллинг. Компьютерная телефония. – 2002. – № 6 (19). 3. *Шваб А. О.* Новые технологии организации баз данных – основа построения современной биллинговой системы / *А. О. Шваб* // Биллинг. Компьютерная телефония. – 2002. – № 6 (19). 4. *Канайлов А. Р.* Универсальная тарификация / *А. Р. Канайлов* // Биллинг. Компьютерная телефония. – 2002. – № 6 (19). 5. *Голубев В.* Проблемы конвергенции / *В. Голубев* // Биллинг. Компьютерная телефония. – 2002. – № 5 (18).

Надійшла до редколегії 24.03.2011.