

Архитектура на система за предварително изчисляване на математически уравнения

Методи Димитров

Abstract: *Many of the software applications use the same equation. This requires their constant recalculation, leading to loss of time and resources. This paper tries to resolve that problem and proposes architecture for a system of prior calculation of mathematical equations. The proposed system helps to build a global database of all prior calculated equations. From this moment on, when some application needs to calculate an equation instead of losing time for calculation, the application connects to the global database and downloads the result.*

Key words: *Architecture, prior calculated, mathematical equations, system*

ВЪВЕДЕНИЕ

В днешно време компютрите съпътстват хората в почти всички области на техния живот. При своята работа всяка една компютърна програма извършва най-различни изчисления. Тъй като приложенията до известна степен пресъздават реалният свят то рано или късно ще възникне необходимостта едни и същи уравнения, използвани за пресмятане на различни неща, да бъдат преизчислени. Това означава загуба на компютърно време и ресурси за преизчисляване на същите уравнения отново и отново. Колкото по-дълго човечеството използва компютрите толкова по-вероятно е едни и същи уравнения с едни и същи входни параметри да се използват отново в различни приложения. От друга страна е много вероятно през времето на работа на дадено приложение, конкретни условия да бъдат повторени отново и отново, което също би довело до загуба на време за преизчисляване на вече получени резултати. От друга страна с увеличаване на сложността на уравненията, времето за пресмятането им би се увеличило значително. Това от би довело до нерационалното използване на изчислителната мощ и ресурсите на компютрите, както и до тяхната неефективна работа.

В [1] и [2] е направено изследване с цел сравняване на времето за намиране на отговора на математически уравнения с помощта на пресмятане и с помощта на търсене сред група от предварително пресметнати стойности. При втория случай, всяко уравнение се пресмята само веднъж, а отговора се запазва за по-късна употреба и при необходимост от повторно пресмятане вместо да се губи време за изчисление на резултата, той се търси сред предварително пресметнатите стойности.

От направените изследвания [1,2] се вижда, че търсенето на резултата сред група от предварително пресметнати уравнения може да бъде по-бързо от самото пресмятане. Това би съхранило време и ресурси, като би направило работата на компютърните приложения по-бърза и ефективна.

Съхраняването на стойностите на предварително пресметнатите уравнения се прави в хеш-базираните структури. Това позволява свързването на ключ (уравнението) и стойност (отговора на уравнението). Използването на хеш-базираните структури прави търсенето на конкретна стойност много бързо.

Текущият доклад представя архитектура на система за предварително изчисляване на математически изрази.

ИЗЛОЖЕНИЕ

Предложената архитектурата на системата за предварително изчисление на математически изрази е показана на фиг.1. Тя съдържа няколко основни части:

1) Клиентски модул – реализиран е като библиотека. Това позволява предложената система да се свърже с дадено клиентско приложение още в процеса на разработката му. След като библиотеката бъде включена в приложението, тя

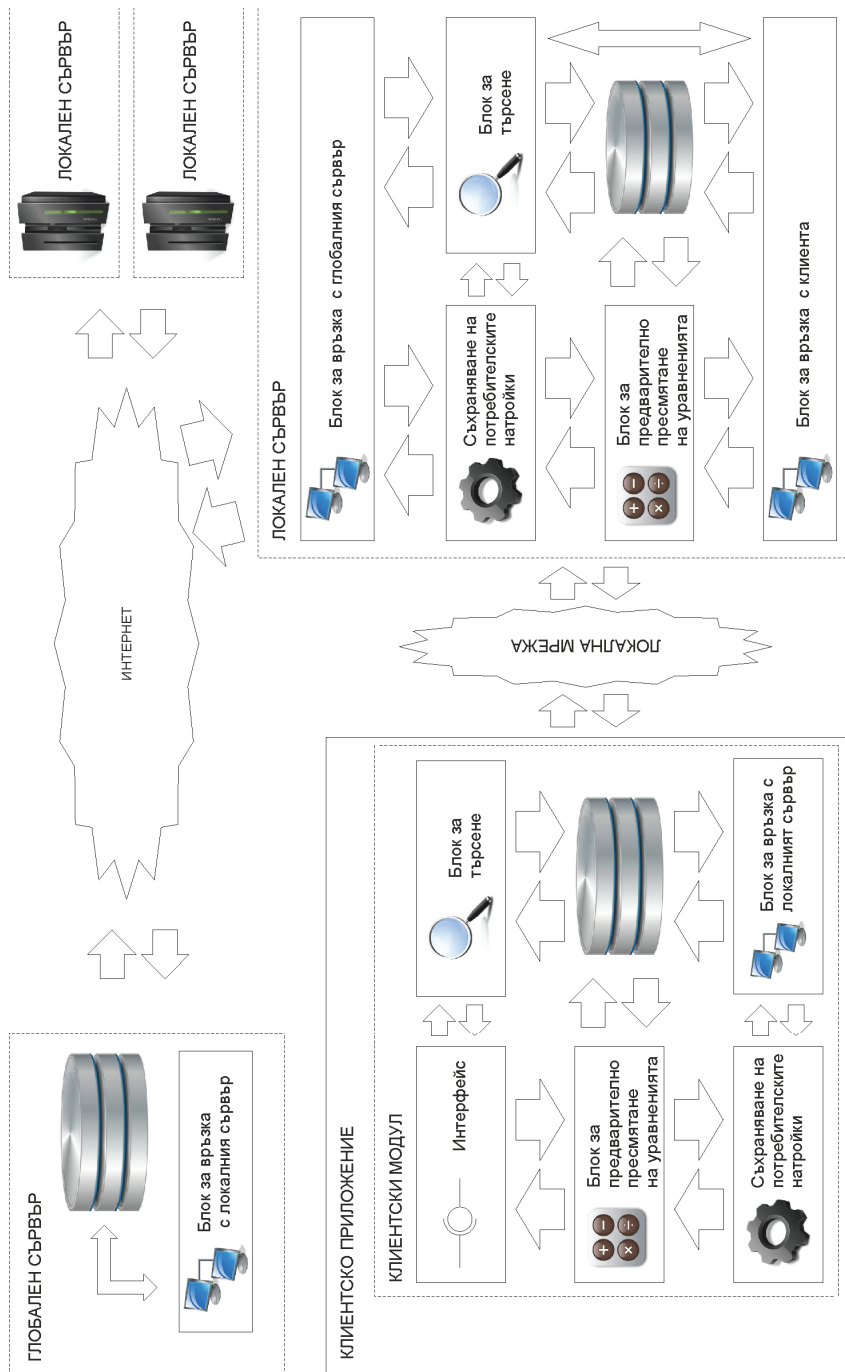
започва да се грижи да предварителното пресмятане на уравненията. От този момент нататък всяко едно пресмятане извършено от приложението може да бъде поето от системата. Този модул на системата съдържа следните основни части:

- Интерфейс – този блок съдържа различни функции и класове, с чиято помощ клиента взаимодейства и управлява клиентския модул извършващ предварителните пресмятания;
- Съхраняване на клиентските настройки – този блок съхранява различни настройки свързани с работата на приложението. Тук се пазят настройки свързани с това колко уравнения да бъдат предварително пресметнати, какви са най- често срещаните стойности на техните променливи и др.;
- Блок за предварително пресмятане – тук става предварителното пресмятане на уравненията. Пресмятането може да бъде направено в началото, още при стартиране на приложението или да бъде правено постепенно в процеса на работа на приложението, като това довежда до плавното натрупване на база данни и спестяване на процесорно време и ресурси;
- Блок за търсене – когато клиента има нужда от намиране на резултата на дадено уравнение, той се обръща към този блок. Блока претърсва базата от данни за търсеното решение, ако не открие такова, блока се обръща към блока за пресмятане, който да пресметне уравнението и да добави резултата в базата от данни. Ако уравнението не е било пресметнато, преди обръщение към блока за предварително пресмятане може да бъде направено обръщение към локалния сървър, който може да съхранява пресметнатия резултат.
- База от данни – представлява хеш- базирана структура или релационна база от данни. Този блок съхранява уравненията и съответните резултати от пресмятанията им;
- Блок за връзка с клиентския сървър - ако на клиентския компютър има инсталиран локален сървър за предварително пресмятане на математически уравнения, тогава клиента би могъл да се свърже към него с помощта на този блок.

2) Локален сървър – когато приложенията използващи метода на предварителното пресмятане на уравнения стане прекалено голям или уравненията станат прекалено много, управлението на всяко едно уравнение по- отделно би довело до значително спадане на производителността. Това се дължи на факта, че всяко едно самостоятелно управляващо се приложение ще започне множество предварителни пресмятания. Това едновременно пресмятане би довело до значително заемане на процесорно време и ресурси и спадане на производителността.

Ролята на локалния сървър е да управлява изчисляването на всички уравнения на текущия компютър. Във всеки един момент той е във връзка със всички стартирани приложения и с текущо изчисляващите се уравнения.

От друга страна използването на едно и също приложение от множество потребители по света би довело до това, че едни и същи уравнения би трябвало да се преизчисляват отново и отново от различните машини. За да се избегне този факт локалният сървър позволява свързване към глобален сървър, в който всяко едно приложение регистрира текущо пресметнатите от него уравнения. По този начин, ако дадено приложение има нужда да открие отговора на дадено уравнение, преди да започне пресмятането, то може да провери в глобалния сървър дали някое друго приложение вече не е намерило резултата и вместо да го пресмята просто да го вземе наготово.



Фигура 1. Архитектура на системата за предварително изчисляване на математически изрази

Връзката с глобалната база от данни, позволява изтеглянето на група от предварително пресметнати стойности върху локалния сървър. По този начин всички уравнения, управлявани от локалния сървър, биха могли да се възползват от получените данни. Това води до увеличаване на бързодействието от намиране на отговора от спестено време за постоянно обръщение към глобалния сървър.

Локалният сървър съдържа следните блокове:

- Блок за съхраняване на клиентските настройки – този блок пази различни настройки свързани с работата на приложението. Тук се пазят настройки свързани с това колко уравнения да бъдат предварително пресметнати, какви са най-често срещани стойности на техните променливи и др.;
- Блок за предварително пресмятане – тук става предварителното пресмятане на уравненията. Те биха могли да се пресметнат в началото на работа на приложението или постепенно, като това би довело до плавното увеличаване на базата данни и спестяване на процесорно време и ресурси;
- Блок за търсене – когато клиента има нужда от намиране на резултата на дадено уравнение, той се обръща към този блок посредством блока за връзка със сървъра. След това се претърсва базата данни за търсеното решение и ако такова бъде открито, то се изпраща на клиента. Ако обаче не бъде открито решение на уравнението, блока се обръща към блока за пресмятане, който да пресметне уравнението и да добави резултата в базата от данни; Блока позволява и обръщане и към глобалната база от данни, преди започване на пресмятане. Това би спестило време ако резултатите са намерени на друг компютър свързан към системата. В такъв случай вместо да се пресмятат наново, резултатите се свалят от другия компютър;
- База от данни – представлява хеш- базирана структура или релационна база от данни. Този блок съхранява уравненията и съответните резултати от пресмятане им;
- Блок за връзка с клиента – с помощта на този блок локалният сървър се свързва с клиента (клиентското приложение);
- Блок за връзка с глобалния сървър – с помощта на този блок локалният сървър се свързва с глобалния сървър. По този начин от интернет могат да бъдат изтеглени група от предварително пресметнати уравнения и да се спести процесорно време и ресурси от пресмятане.

3) Глобален сървър – представлява регистър на всички пресметнати до момента уравнения. Той съхранява връзката между дадено уравнение и компютъра, който е направил пресмятането и съответно съхранява полученият резултат.

Ако на дадено приложение се наложи да пресметне дадено уравнение, преди започване на изчисленията, то може да провери дали вече някое друго приложение не е направило пресмятането. Ако глобалният сървър пази данни за пресметнатото уравнение, резултата бива изпратен на приложението, което има нужда от него, ако в глобалния сървър няма данни за отговора на уравнението, то приложението го пресмята и след това го регистрира в глобалния сървър, така, че при нужда пресметнатият резултат да бъде използван отново.

Глобалният сървър съдържа два блока:

- База от данни – представлява релационна база от данни, свързваща конкретно уравнение и компютъра съхраняващ резултат от пресмятането на това уравнение.
- Блок за връзка с локалните сървъри – с помощта на този блок става връзката между локалните и глобалния сървър.

ЗАКЛЮЧЕНИЕ

Във връзка с представената архитектура на система за предварително пресмятане на математически уравнения могат да се направят следните по-важни изводи:

1. Предложената архитектура има за цел подобряване на бързодействието на приложенията, чрез намаляване на времето за намиране на отговор на дадено уравнение;
2. Архитектурата е изградена на три нива: клиентско приложение, локален и глобален сървър. Това я прави много гъвкава като дава възможност да се изгради една глобална разпределена база данни със всички пресметнати уравнения, от която могат да се възползват всички приложения използващи предложеният начин на намиране на отговора на математическите уравнения.

ЛИТЕРАТУРА

[1] Dimitrov M., Evaluation of the advantages of calculating mathematical equations using a hash-based structures mapping key to value, Conference of Varna Free University "Informatics in Scientific Knowledge 2012", Varna, 2012

[2] Dimitrov M., Evaluation of the advantages of calculating mathematical equations using a hash-based structures mapping key to value on Ubuntu platform, Eight international scientific-practical conference, Vinnytsia National Technical University, Ukraine, 2012

За контакти:

маг. инж. Методи Димитров, Катедра *Информатика и информационни технологии*, Русенски университет "Ангел Кънчев", тел.: 082-888 470, e-mail: mdimitrov@uni-ruse.bg

Докладът е рецензиран.