

Барабаш О. В., доктор техн. наук, професор. Тел.: +380 (95) 870 24 90. E-mail : bar64@ukr.net
Куліковська Ю. А. аспірант. Тел.: +380 (97) 15307 96. E-mail : jul5@ukr.net
(Державний університет телекомунікацій, м. Київ)

АНАЛІЗ ЕКСПЛУАТАЦІЙНИХ ПРОБЛЕМ РОЗПОДІЛЕНИХ СИСТЕМ

Barabash O. V., Kulikovs'ka Yu. A. Analysis of operational problems of distributed systems. Some basic concepts and the characteristics of the Open Distributed Processing Systems are considered, those classification is set out, so as for the requirements for their designing problems in a terms of its practical application. Several types of the Open Distributed Processing Systems were described according to the standard Reference Model for Open Distributed Processing, such as the access transparency, localization, movement changes location, access concurrency, scalability and the replication failures.

The problem of the distributed systems openness, ensuring the operations safety and reliability considered. So as the administration of Distributed Systems associated with the resources fragmentation, regarding the following tasks: the system balancing for the data recovery components in case of their distortion, collecting of the statistics from the node system software, autonomous update on the nodes of the system. The tasks for the effectiveness of the balanced providing for the Distributed Processing Systems were described in a point of the dynamic and static balancing.

Keywords: distributed information systems, operation, administration, scaling, load balancing, data recovery, data protection

Барабаш О. В., Куліковська Ю. А. Аналіз експлуатаційних проблем розподілених систем. Визначені основні поняття та особливості розподілених систем, приведена їх класифікація, сформульовані вимоги до їх проектування з точки зору проблематики експлуатації таких систем. Розглянуті питання адміністрування розподілених систем, пов'язані із фрагментацією ресурсів, а також вирішенням наступних задач: балансування навантаження на вузли системи, відновлення даних у разі їх спотворення, збір статистики з вузлів системи, оновлення програмного забезпечення на вузлах системи в автоматичному режимі.

Ключові слова: розподілені інформаційні системи, експлуатація, адміністрування, масштабування, балансування навантаження, відновлення даних, захист інформації

Барабаш О. В., Куликовская Ю. А. Анализ эксплуатационных проблем распределенных систем. Определены основные понятия и особенности распределенных систем, приведена их классификация, сформулированы требования к их проектированию с точки зрения проблематики эксплуатации таких систем. Рассмотрены вопросы администрирования распределенных систем, связанные с фрагментацией ресурсов, а также решением следующих задач: балансирования нагрузки на узлы системы, восстановления данных в случае их искажения, сбора статистики из узлов системы, обновления программного обеспечения на узлах системы в автоматическом режиме.

Ключевые слова: распределенные информационные системы, эксплуатация, администрирование, масштабирование, балансирование нагрузки, восстановление данных, защита информации

1. Постановка задачі

Розвиток суспільства нерозривно пов'язаний з неухильним зростанням об'ємів і різноманітністю інформації, яка підлягає перетворенню, передачі, обробці, зберіганню та відображенню. Глобалізація інформатизації суспільства та активний процес науково-технічного розвитку сприяє формуванню єдиного світового інформаційно-телекомунікаційного простору та призводить до появи розподілених систем (РС) [1]. Існування таких систем зумовлює появу класу задач, які, насамперед, мають за мету досліджувати питання підвищення якості та швидкості обробки в *першу* чергу великої кількості даних і в *другу* чергу – даних в РС [2]. В зв'язку з цим зростає значення РС для зберігання і обробки даних, як засобу розв'язання цієї проблеми [3].

Одним із основних завдань будь-якої РС є аналіз властивостей отриманих даних, які, в силу ряду причин, не можуть бути оцінені на одному вузлі. Для досягнення поставленої мети і прискорення часу обробки необхідно на *першому* етапі розіслати дані на розподілені вузли системи, а на *другому* – зібрати дані з розподілених вузлів і агрегувати ці дані в загальне глобальне уявлення. Наявність динаміки, що накладає часті зміни в локальні

значення, які в свою чергу впливають на загальні глобальні зміни всієї системи, ускладнюють задачу реалізації та проектування РС. Тому створення ефективних і адаптивних РС дозволяє значно прискорити швидкість обробки даних [3].

Проведемо аналіз задач і питань, які виникають при проектуванні та експлуатації РС.

2. Поняття розподіленої системи

Аналіз літератури показав, що на сьогоднішній день існує велика кількість означень поняття «розподілена система» в залежності від сфери її застосування, а це вказує на відсутність загальноприйнятого і в той же час суворого визначення розподіленої системи. Проте, ґрунтуючись на визначенні відомого фахівця в галузі інформатики Е. Таненбаума, більш вузьку РС можна визначити як набір з'єднаних каналами зв'язку незалежних комп'ютерів, які з точки зору користувача деякого програмного забезпечення виглядають єдиним цілим [4].

В запропонованому означенні необхідно звернути увагу на те, що самі по собі незалежні комп'ютери не можуть представлятися користувачеві єдиною системою. Забезпечити це можна тільки за допомогою додаткового спеціального програмного забезпечення, що називається програмним забезпеченням проміжного рівня. Саме його допомогою користувачі вважають, що мають справу з єдиною системою, а всі відмінності між комп'ютерами і способами зв'язку між ними залишаються прихованими для користувачів.

З огляду на це, в статті РС буде розглядатися, як програмно-апаратна система, в якій виконання операцій (дій, обчислень), необхідних для забезпечення цільової функціональності системи, розподілено (фізично або логічно) між різними виконавцями. У обчислювальній сфері під РС, як правило, розуміють програмно-апаратну систему, створену для конкретного практичного застосування, функціонал якої розподілений на різних вузлах територіально [5].

За останні кілька років популярність розподілених систем та їх роль тільки зростає. Серед основних причин цього факту можна виділити наступні [5]:

– *Географічно розподілене обчислювальне середовище.* Сьогодні в більшості випадків обчислювальне середовище за своєю природою являє собою територіально розподілену систему.

– *Вимога збільшення продуктивності обчислень.* Швидкодія традиційних однопроцесорних систем стрімко наближається до своєї межі. Щоб підвищити продуктивність в сотні або тисячі разів і при цьому забезпечувати хорошу масштабованість рішення необхідно звести в єдине ціле численні процесори і забезпечити їх ефективну взаємодію. Цей принцип реалізується у вигляді великих багатопроцесорних систем і багатомашинних комплексів.

– *Спільне використання ресурсів.* Важливою метою створення та використання розподілених систем є надання користувачам (і додаткам) доступу до віддалених ресурсів і забезпечення їх спільного використання. У даному формулюванні термін ресурс відноситься як до компонентів апаратного забезпечення обчислювальної системи, так і до програмної абстракції, з якими працює розподілена система.

– *Відмовостійкість.* Характерною рисою розподілених систем, яка відрізняє їх від одиничних комп'ютерів, є стійкість до часткових відмов, тобто система продовжує своє функціонування при цьому загальна продуктивність майже не знижується. Подібна можливість досягається за рахунок надмірності, коли в систему додається додаткове обладнання (апаратна надмірність) або процеси (програмна надмірність), які роблять можливою правильне функціонування системи при непрацездатності або некоректної роботи деяких з її компонентів. В цьому випадку розподілена система намагається приховувати факти відмов або помилок у одних процесах від інших процесів.

3. Класифікація розподілених систем

Наявність в розподілених системах різних ознак дозволяє класифікувати їх за кількістю елементів у системі, за рівнем організації розподілених систем, за типом наданих ресурсів, а також ряду інших параметрів.

На сьогоднішній день можна виділити дві основні класифікації розподілених систем, які умовно можна розподілити на шість підтипів. *По-перше*, це класифікація за розмірами систем і способам адміністрування. *По-друге*, розподілені системи можна розглядати з точки зору їх функціональності, тобто за типом наданих ресурсів і типів прикладних задач, під вирішення яких вони оптимізовані.

Розглянемо класифікацію за кількістю елементів у системі. Такі розподілені системи поділяються на: *кластерні*, розподілені системи *корпоративного* рівня, *глобальну* систему.

Кластер – це кілька десятків комп'ютерів, об'єднаних за допомогою локальної мережі. Адміністрування здійснюється вручну.

Розподілена система корпоративного рівня – десятки і навіть сотні комп'ютерів, при роботі яких необхідно встановлювати правила спільного використання ресурсів. Масштаб таких систем, як правило, невеликий, і можна використовувати прямі адміністративні заходи для організації роботи ресурсів і користувачів.

Глобальна система – величезна кількість комп'ютерів, число яких може досягати декількох мільйонів, розподілених по світу і об'єднаних глобальною мережею. Адміністративне програмне забезпечення вбудовано в проміжне програмне забезпечення. Прикладом глобальної розподіленої мережі є Інтернет, де в якості наданого ресурсу є інформаційне поле.

Враховуючи ознаку за типом наданих ресурсів системи можна розподілити на [5]:

- розподілені обчислювальні системи (Computational Grid) – це тип систем, в яких основним ресурсом є обчислювальна потужність всієї системи. Основний напрямок розвитку систем подібного типу полягає в нарощуванні обчислювальних потужностей системи, за допомогою збільшення числа обчислювальних вузлів. Прикладом розподілених обчислювальних систем є кластери;
- розподілені інформаційні системи (Data Grid) – тип систем, в яких основним ресурсом є обсяг пам'яті даних, що не потребують величезної кількості обчислювальних ресурсів. Ці системи розглядаються як величезні сховища даних;
- семантичний Грід (Semantic Grid) – надає семантичні описи ресурсів і можливості простого пошуку ресурсів та сервісів, забезпечуючи їх інтегруєбельності і інтегроване спільне використання в контексті завдання, формульованій в термінах предметної області.

Для спрощення розгляду розподілених систем, доцільно умовно розподілити їх на типи.

На Рис. 1 наведено приклади розподілених систем відносно їх класифікації та типів.

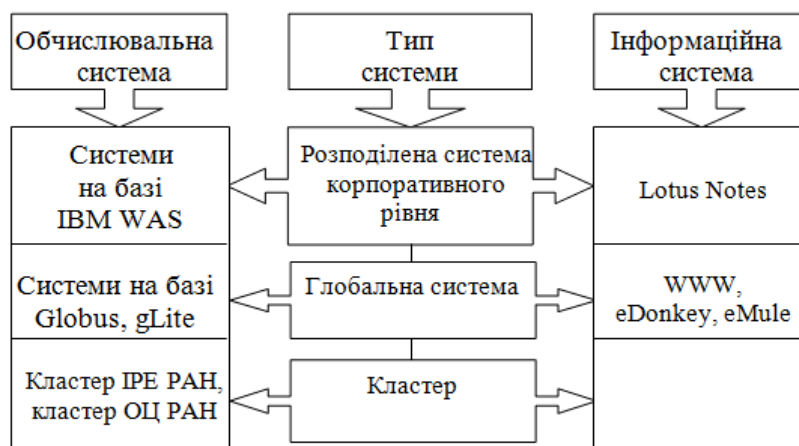


Рис. 1. Класифікація розподілених систем

4. Вимоги до розподілених систем

Щоб досягти мети свого доцільного існування, а саме поліпшення виконання запитів користувача, розподілена система повинна задовольняти деяким необхідним вимогам. Можна сформулювати наступний набір вимог, які дозволяють цілісно функціонувати розподіленій системі: прозорість, відкритість системи, інформаційна безпека, масштабованість РС. Розглянемо кожну характеристику докладніше [5-7].

4.1. Прозорість. Під прозорістю РС розуміється її здатність приховувати свою розподілену природу, а саме, розподіл процесів і ресурсів по множині комп'ютерів, і представлятися для користувачів і розробників додатків у вигляді єдиної централізованої комп'ютерної системи. Стандарти еталонної моделі для розподіленої обробки у відкритих системах Reference Model for Open Distributed Processing (RM-ODP) визначають декілька типів прозорості. Найбільш важливі з них перераховані нижче.

Прозорість доступу. Незалежно від способів доступу до ресурсів і їх внутрішнього представлення, звернення до локальних і віддалених ресурсів здійснюється однаковими способами. На базовому рівні різниця архітектур обчислювальних платформ є прихованою, але, що більш важливо, досягається угода про те, як ресурси різнорідних машин, будуть представлятися користувачам розподіленої системи єдиним чином.

Прозорість місцеположення. Дозволяє звертатися до ресурсів без знання їх фізичного місця розташування. У цьому випадку ім'я запитуваного ресурсу не повинно давати жодного уявлення про те, де ресурс розташований. Тому важливу роль для забезпечення прозорості місцеположення грає іменування ресурсів.

Прозорість переміщення. Переміщення ресурсу або процесу в інше фізичне місцеположення залишається непомітним для користувача РС. Тут варто зазначити, що виконання вимоги прозорості місцеположення не гарантує прозорості переміщення. Тобто, якщо розподілена система приховує місце розташування ресурсу, це не означає, що його можна змінити непомітно для користувача.

Прозорість зміни місця розташування. Більш суворе по відношенню до попереднього вимога приховати факт переміщення ресурсу під час його використання.

Прозорість паралелізму доступу. Різні користувачі РС повинні мати можливість паралельного доступу до загальних даних. При цьому необхідно забезпечити паралельне спільне використання ресурсами системи, а відповідно, забезпечити приховування факту спільного використання ресурсів.

Прозорість масштабованості розподіленої системи. Масштабованість є однією з найважливіших характеристик РС. До недавнього часу основна методика, що дозволяла значно збільшити потужність кластерів, полягала у нарощуванні різних ресурсів системи, приміром, оперативної пам'яті, кількості та обсягу жорстких дисків. Даний метод, в ряді випадків, є неефективним. Зокрема, в силу обмеженості проектного рішення, нарощування обчислювальних потужностей не завжди можливо. Але не тільки обмеженість проектного рішення є стримуючим фактором, що забезпечує неможливість масштабованості системи. У програмне забезпечення, яке здійснює взаємодію всіх компонентів системи з операційною системою, часто закладено обмеження на загальну кількість обчислювальних вузлів системи.

Прозорість реплікації. З метою забезпечення збереження даних, особливо на розподілених файлових системах, необхідно забезпечити реплікацію даних. Користувачеві не має бути відомо, що реплікація даних існує. Для приховування даного чинника, необхідно, щоб у наданих даних або ресурсів, були однакові імена.

Прозорість відмов. Мається на увазі, що система повинна намагатися приховувати часткові відмови, дозволяючи користувачам і додаткам виконати свою роботу незалежно від збоїв в апаратних або програмних компонентах розподіленої системи, а також приховувати

факт їх подальшого відновлення. У зв'язку з тим, що будь-який процес, комп'ютер або мережеве з'єднання можуть відмовляти незалежно від інших в довільні моменти часу, кожен компонент розподіленої системи повинен бути готовий до збоїв в інших компонентах і обробляти подібні ситуації відповідним чином.

4.2. Відкритість системи. На відміну від ранніх РС, які за своєю суттю були обмеженими і закритими, оскільки вони створювалися в основному в межах окремих організацій та для вирішення конкретних завдань, сучасні розподілені системи створюються все більш відкритими. Застосування принципу відкритості до РС стало можливим завдяки розвитку ліній передачі даних, збільшення продуктивності процесорів, а також загального розвитку інформаційних технологій. Під відкритістю РС розуміється можливість взаємодії з іншими відкритими системами. Відкриті системи повинні мати такі характеристики:

- РС повинні відповідати чітко визначеним інтерфейсам;
- системи, що входять до складу РС повинні легко взаємодіяти між собою;
- системи повинні забезпечувати переміщення додатків.

Відкритість системи може бути досягнута за допомогою: мов програмування, апаратних платформ, програмного забезпечення.

4.3. Інформаційна безпека. Особливе місце в сучасних РС займає їх безпека. Безпека РС є, в загальному випадку, сукупністю 3 факторів [7]:

- забезпечення конфіденційності даних та ресурсів;
- забезпечення конфіденційності доступу до ресурсів для множини користувачів;
- забезпечення цілісності ресурсів і даних.

Необхідність створення розподілених систем, які забезпечують необхідну безпеку даних і всієї структури РС, виникає постійно [8, 9]. Багато питань безпеки можуть бути вирішені на рівні окремих вузлів РС, наприклад, шляхом установки фаєрволів і антивірусного ПЗ на окремі вузли системи, введенням політики аутентифікації користувачів та іншими методами. Але в силу особливості архітектури більшості РС, даний підхід не завжди є ефективним.

Програмне забезпечення не завжди може забезпечити необхідну конфіденційність даних в розподіленій системі. Наприклад, програмне забезпечення не завжди може повноцінно захистити від MITM і DDOS атак на розподілену мережу. Найчастіше методи захисту від подібних атак не завжди є прийнятними для вузлів обчислювальної мережі.

Важливим показником, при організації захисту РС, є рівень доступності системи. Рівень доступності РС визначається не тільки доступністю ресурсу в момент часу t , а й принципами організації захисту РС, так як більшість програмних засобів, що забезпечують захист від атак, спрямовані на відмову в обслуговуванні. Велику увагу даному питанню приділяється багатьма виробниками антивірусного ПЗ.

4.4. Надійність РС. У зв'язку з появою нових методів і алгоритмів, вимогливих до обчислювальних ресурсів і, найголовніше, до ресурсів часу, необхідність в доступності розподілених систем в момент часу t стає вкрай актуальним. Основним показником, який визначає надійність всієї РС, є відмовостійкість. Відмовостійкість – це найважливіша властивість обчислювальної системи, яка полягає в можливості продовження дій, заданих програмою, після виникнення певних несправностей.

4.5. Масштабованість розподілених систем. У загальному випадку масштабованість визначають, як здатність обчислювальної системи ефективно справлятися зі збільшенням числа користувачів або підтримуваних ресурсів без втрати продуктивності та без збільшення адміністративного навантаження на її управління. Масштабованість є важливою властивістю обчислювальних систем, якщо їм може знадобитися працювати під великим навантаженням, оскільки означає, що вам не доведеться починати з нуля і створювати абсолютно нову інформаційну систему. Для розподілених систем зазвичай виділяють декілька параметрів,

характеризують їх масштаб: кількість користувачів і кількість компонентів, що складають систему, ступінь територіальної віддаленості мережевих комп'ютерів системи один від одного і кількість адміністративних організацій, що обслуговують частини розподіленої системи.

5. Задачі експлуатації розподілених систем

При проектуванні та експлуатації розподілених систем на практиці, реалізація властивостей для функціональної роботи системи є непростою задачею. Основні складнощі випливають саме з того факту, що компоненти розподіленої системи територіально віддалені один від одного і виконуються на незалежних комп'ютерах в мережі. Ігнорування цієї обставини на етапі проектування і розробки відповідного програмного забезпечення часто призводить до суттєвих проблем в функціонуванні розподіленої системи.

Ряд робіт [5-7, 10] вітчизняних та зарубіжних науковців надає можливість сформулювати та проаналізувати основні задачі, що виникають при проектуванні та експлуатації розподілених систем. В порівнянні з традиційними централізованими системами (РС забезпечують значно меншу вартість розгортання і простоту реалізації), РС мають і ряд істотних недоліків. Основними задачами РС в порівнянні з традиційними системами є:

- питання адміністрування системи;
- обмеженість масштабованості РС;
- переміщення програмного забезпечення.

5.1. Адміністрування системи. Фрагментація ресурсів в РС зумовлює необхідність створення гнучких засобів адміністрування з можливістю їх налаштування. Оскільки, в глобальних розподілених системах адміністрування має відбуватися в автоматичному режимі, то у зв'язку з цим виникають такі основні особливості адміністрування розподілених систем:

- балансування навантаження на вузли системи;
- відновлення даних у разі виникнення помилки;
- збір статистики з вузлів системи;
- оновлення програмного забезпечення на вузлах системи в автоматичному режимі.

Цей перелік є узагальненим, тобто кожна конкретна розподілена система може провокувати виникнення й інших проблем. Проте, вказані проблеми є найбільш поширеними при проектуванні системи, а тому заслуговують особливої уваги. Останні дві проблеми добре вивчені, і на ринку програмного забезпечення розподілених систем існує безліч розроблених програмних засобів, які забезпечують як збір статистики, так і оновлення програмного забезпечення. Науковий же інтерес представляють методи балансування навантаження на вузли системи і методи відновлення даних у разі виникнення помилок. У силу специфіки РС, а також гетерогенності обладнання та архітектури РС, не існує єдиного методу проектування, який забезпечував би вирішення цих проблем.

Балансування навантаження. Важливою задачею при проектуванні РС є забезпечення ефективного балансування навантаження на вузли системи. Правильно вибрана стратегія балансування навантаження надає вирішальний вплив на загальну ефективність і швидкість роботи розподіленої системи. Сьогодні існує багато підходів [11] для вирішення даної проблеми.

У загальному випадку можна виділити узагальнену класифікацію методів балансування навантаження обчислювальних вузлів. За характером розподілу навантаження на обчислювальні вузли розрізняють: динамічне балансування (перерозподіл); статичне балансування.

Статичне балансування, найчастіше, виконується в результаті апріорного аналізу. При розподілі ресурсів по обчислювальним вузлам аналізується модель розподіленої

системи, з метою виявлення найкращої стратегії балансування. При цьому необхідно враховувати структуру РС, а також конфігурацію обчислювальних вузлів. Проте, існує недолік даного методу, який полягає в необхідності асоціації вузлів з різною конфігурацією обладнання з обчислювальною складністю завдання, що не завжди представляється можливим.

Динамічне балансування РС полягає в адаптації навантаження на вузли розподіленої системи в ході роботи, що в свою чергу дозволяє ефективніше використовувати ресурси мережі. Необхідність динамічного балансування виникає в тому разі, коли можливо спочатку апріорно припустити загальну завантаженість мережі. Такі ситуації найбільш часто виникають, наприклад, в задачах математичного моделювання, коли в ході обчислень на кожній ітерації, складність обчислення підвищується і, відповідно, загальний час обчислень також збільшується. Також динамічне балансування дозволяє використовувати програмне забезпечення, яке буде інваріантне до архітектури розподіленої системи.

Задача відновлення даних у разі виникнення помилок. У ході експлуатації РС найбільш часто виникає проблема відстеження збоїв і подальше відновлення даних. Дана ситуація може виникнути, наприклад, в ході збою живлення одного з вузлів РС. Автоматичне відновлення даних є складним завданням, яка включає в себе певну кількість проблем. У ході відновлення необхідно з'ясувати характер помилки, класифікувати її і в автоматичному режимі провести відновлення всіх даних. При цьому повинна бути збережена не тільки вся цілісність пов'язаних між собою даних, а й доступність інших даних, так як відновлення повинно відбуватися без блокування основних ресурсів на читання-запис, тобто розподілена система повинна функціонувати без зупинки. В наш час існує багато підходів для вирішення даної проблеми. Наприклад, одним з методів відновлення в інформаційних розподілених системах (розподілених СУБД) є використання так званого журналу транзакцій, в якому зберігається вся інформація про всі зміни, що відбулися в базі даних. Складність у даному випадку полягає в правильності класифікації помилок і правильності застосування методів відновлення даних в автоматичному режимі.

5.2. Питання обмеженості масштабованості. Масштабованість розподілених систем одне з першочергових завдань при проектуванні РС. Розподілені системи дозволили уникнути головного недоліку централізованих систем – обмеженості нарощування обчислювальних потужностей системи. Існують три основні показники масштабованості системи [6] :

- масштабованість РС по відношенню до її розмірів. Система вважається масштабованою по відношенню до її розмірів, якщо вона забезпечує простоту підключення до неї нових вузлів;
- географічна масштабованість. Система вважається географічно масштабованою, якщо до її мережі можливе підключення нових вузлів, без прив'язки до конкретної географічної зони (країна, місто, дата - центр і т.д.), тобто глобально розподілених вузлів;
- масштабованість управління. Система вважається масштабованою в плані управління ресурсами, якщо при зростанні загальної кількості вузлів системи, адміністрування системи не ускладнюється.

При вирішенні завдання масштабованості системи необхідно вирішити ряд питань, з яких можна виділити чотири основних.

Збільшення кількості вузлів системи, що не завжди надається можливим, у зв'язку з обмеженістю служб, алгоритмів, так як часто багато служб налаштовані на використання

конкретної кількості обладнання, наприклад, на використання тільки одного конкретного сервера, конкретної архітектури.

Обмеженість можливостей сервера, який здійснює агрегування даних, зібраних з вузлів системи в загальне глобальне уявлення.

Обмеженість мереж передачі даних. Так як, в розподілених системах вузли можуть територіально перебувати в різних географічних точках, питання надійності та його проблематика є актуальною. При незначній швидкості передачі даних можливе зниження загальної надійності і продуктивності РС.

Обмеженість алгоритмів обробки даних. Мінімальне перенавантаження мережі, може бути досягнуто за рахунок впровадження відповідних алгоритмів збору даних з вузлів системи.

5.3. Задача переміщення програмного забезпечення. Задача полягає у неможливості запуску створеного додатку на різних архітектурах. Стрімкий розвиток програмних архітектур, мов програмування, а також загальне розвиток всієї ІТ-індустрії в цілому - все це призвело до необхідності створення методологій переміщувального програмного коду. В глобально розподілених системах (вузли використовують різне гетерогенне обладнання з різними операційними системами) питання переміщувального програмного забезпечення є першочерговим. Наприклад, для об'єднання обчислювальних машин в одну глобальну обчислювальну GRID мережу потрібно написати клієнтський додаток для кожного обчислювального вузла, з урахуванням специфіки його архітектури і встановленої операційної системи, що є складним завданням. Щодня вимоги до збільшення мобільності програмних продуктів, призводить до необхідності проведення досліджень у даному напрямку. Проблемам забезпечення багатоплатформності присвячено ряд опублікованих наукових робіт, в яких реалізовані основні підходи і методи до створення переміщуваних додатків [12].

Отже, проблеми які виникають в розподілених системах, можна узагальнити, як показано на Рис. 2.

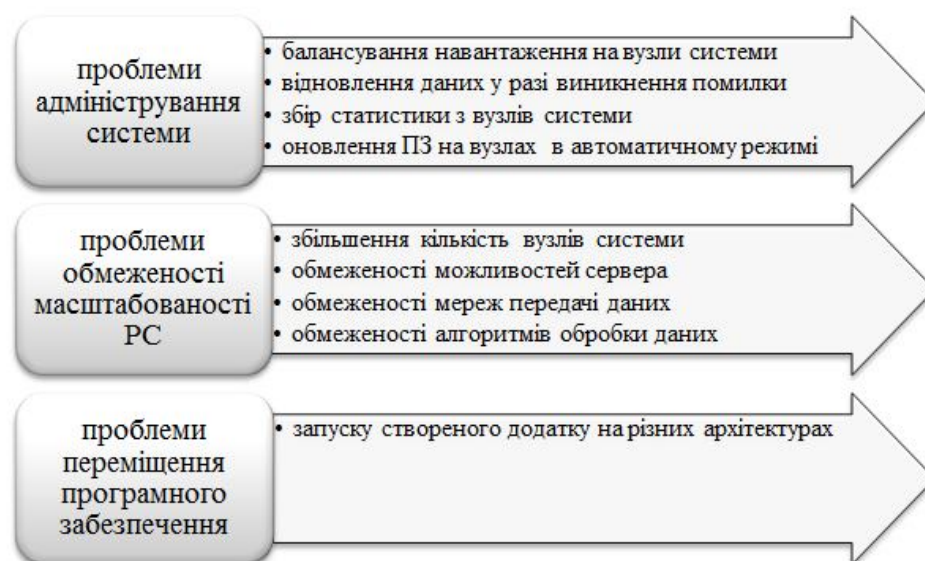


Рис. 2. Узагальнення проблем розподілених інформаційних систем

6. Висновки

На сьогоднішній день РС є інструментарієм, що дозволяє вирішувати певний клас складних завдань, більшість з яких інші методи неспроможні розв'язати. Обмеженість нарощування обчислювальних потужностей, яка властива централізованим системам, усунена у випадку розподілених систем. Проте і ці системи мають певні недоліки, зумовлені використанням в їх складі устаткування різних виробників з різними типами архітектур. У зв'язку з великою різноманітністю аспектів побудови обчислювальних систем, а також різноманітністю існуючих операційних систем, виникає необхідність у створенні методів адаптивного планування розподілу потоків в РС, що дозволить значно прискорити швидкість обробки вхідних заявок на обслуговування та збільшення загальної продуктивності системи. Наявна позитивна тенденція у вирішенні даних проблем надає можливість стверджувати про стрімкий технологічний стрибок у даному напрямлені.

Література

1. Барабаш О. В. Построение функционально устойчивых распределенных информационных систем. – Київ : НАОУ, 2004. – 226 с.
2. Tsvetkov V. Ya., Lobanov. Big Data as Information Barrier / V. Ya. Tsvetkov, A. A. Lobanov // European Researcher. – 2014. – Vol.(78). – № 7-1. – P. 1237-1242.
3. Цветков В. Я. Базы данных. Эксплуатация информационных систем с распределенными базами данных / В. Я. Цветков. – Москва : МИИГАиК, 2009. – 88 с.
4. Tanenbaum A. Distributed systems / A. Tanenbaum, M. Van Steen. – Pearson Prentice Hall, 2007.
5. Косяков М. С. Введение в распределенные вычисления / М. С. Косяков. – Санкт-Петербург : НИУ ИТМО, 2014. – 155 с.
6. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. Ван Стеен. – Санкт-Петербург : Питер, 2003. – 877 с.
7. Алпатов А. Н. Проблемы распределенных систем [Электронный ресурс] / А. Н. Алпатов, В. Я. Цветков // – Режим доступа : <http://cyberleninka.ru/article/n/problemuy-raspredeleennyh-sistem>
8. Бойченко О. В., Модель корпоративного інформаційного захисту об'єкту інформатизації / О. В. Бойченко, Я. І. Торошанко // Наукові записки Українського науково-дослідного інституту зв'язку. – 2011. – №4(20). – С. 15-19.
9. Бойченко О. В. Внутрішні та зовнішні конфлікти в інформаційних системах / О. В. Бойченко, Я. І. Торошанко // Матер. VI междунар. научно-техн. симпозиума «Новые технологии в телекоммуникациях 99», 21 - 25 января 2013, Вышков. – С. 206-207.
10. George Coulouris. Distributed Systems Concepts and Design / George Coulouris, Jean Dollimore, Tim Kindberg. – 3-rd edition. – Addison-Wesley.
11. Бабич А. В. Алгоритмы динамической балансировки нагрузки в распределенной системе активного мониторинга / А. В. Бабич, Г. Б. Берсенев // Известия ТулГУ. Технические науки. – 2011. – №. 3. – С. 251-261.
12. James D. Mooney. Bringing Portability to the Software Process / James D. Mooney // Technical Report TR 97-1, Dept. of Statistics and Computer Science. – West Virginia University, Morgantown WV, 1997.

Дата надходження в редакцію: 14.07.2015 р.

Рецензент: д.т.н., проф. Ю. В. Кравченко