

Ідентифіковано основні фактори ризику вантажовласників у процесі транспортування, сформовано економіко-математичну модель оптимізації схеми доставки вантажів з урахуванням ризиків.

**Ключові слова:** схема доставки, ризик, модель, оптимізація, час, якість

Identified the main risk factors for cargo, formed the economic and mathematical model of optimization schemes of delivery of goods taking into account the risks.

**Keywords:** delivery scheme, the risk model, optimization, time, quality

**УДК 004.7**

**К. В. ХАРЧЕНКО**, канд. техн. наук, доц., Інститут прикладного системного аналізу, НТУУ «КПІ», Київ

## **АРХІТЕКТУРА ВІДКРИТОЇ P2P ХМАРНОЇ СИСТЕМИ**

Запропонована архітектура відкритої хмарної системи на основі P2P мережі. Реалізація такої архітектури дозволяє побудову приватної або публічної хмарної системи. Описана концепція побудови P2P мережі та взаємодію клієнт-хмара. Описаний загальний протокол обміну даними між клієнтами та хмарою.

**Ключові слова:** високопродуктивні обчислювальні системи, хмарні обчислення, P2P системи, спільні хмарні системи.

**Вступ.** Історія створення пірінгових систем, Peer-to-Peer, або P2P почалася зі створення децентралізованих мереж USENET (1979), FidoNet (1984), Napster (1999). Аналогічний принцип побудови однорангових розподілених мереж використовувався в 1984 році в розробці архітектури Advanced Peer to Peer Networking фірми IBM [1]. Переваги використання пірінгових систем полягають у розподілі та зменшенні вартості використання системи, одноранговому об'єднанню ресурсів, масштабованості та автономності системи. Пірінгові системи дуже широко використовувалися у галузі зберігання та обміну даними між користувачами (іноді користувачі порушували авторські права на розповсюдження мультимедійної інформації в таких мережах, наприклад KaZaA, EDonkey, Napster). Іншими прикладами пірінгових обчислювальних систем є проекти SETI at Home [11], Folding@Home, World Community Grid (IBM) та BOINC[14]. В телекомунікаційній галузі успішно зарекомендувала себе система Skype, яка також побудована на P2P технології.

Широке використання хмарних систем, Cloud Computing, почалася з реорганізації обчислювальних потужностей дата центру компанії Amazon, яка створила можливість використання віртуальних систем класу IAAS (Infrastructure as Service). Хоча сама ідея хмарних систем була запропонована Джоном МакКарти (John McCarthy) ще у 1960-х роках і розвинена у Дугласом Паркхілом (Douglas Parkhill) у [12]. Надалі отримали розвиток інші класи хмарних систем, такі як PAAS (Platform as a Service), SAAS (Software as a Service). Головні переваги використання хмарних систем полягають у мінімізації ресурсів на підтримку інфраструктури дата центрів та масштабуванні потужностей. Сучасний розвиток хмарних систем перейшов у площину створення публічних, приватних та гібридних хмарних систем.

Ідея об'єднання пірінгових систем та хмарних обчислень полягає у створенні хмарного сервісу, де обчислювальні потужності приватного або публічного сервісу

© К. В. ХАРЧЕНКО, 2013

РaaS будуть надавати самі користувачі за схемою спільного використання обчислювальних ресурсів. Користувачі такої системи Community Cloud [13] отримують можливість завантажити обчислювальну задачу у P2P хмару, а у вільний час надавати певний обсяг вільних обчислювальних ресурсів іншим користувачам.

**Мета роботи.** Метою роботи є дослідження архітектури пірінгових мережевих систем для використання у побудові відкритої хмарної системи з можливістю рівноправної участі сторін. У роботі пропонується розглянути найбільш типову архітектуру побудови хмарної системи з використанням пірінгових мереж.

- **Архітектура P2P хмарної системи.** Архітектура P2P хмарної системи має такі складові:

- клієнт зі встановленою віртуальною машиною та доступом до мережі Інтернет;
- централізований сервер реєстрацій;
- система авторизації клієнта, яка також виконує підрахунок кількості використаних та наданих іншим користувачам ресурсів;
- система розподілу навантаження (балансер);
- певна кількість серверів з віртуальними машинами, які надають обчислювальні потужності клієнтам;
- транспортний рівень передачі даних за допомогою HTTPs запитів з форматами JSON, XML для передачі клієнтських вхідних та вихідних даних.

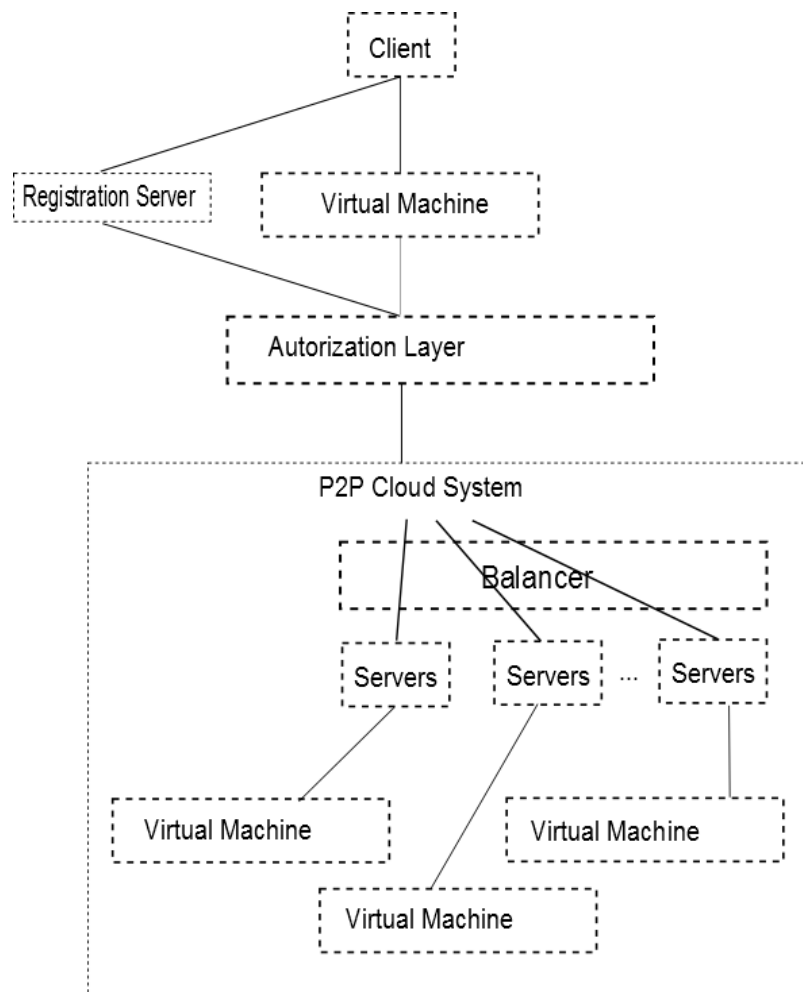


Рис. 1 – Архітектура відкритої хмарної системи

### Концепція обчислень у відкритій P2P хмарній системі

Базові принципи побудови P2P хмарної системи полягають у наступному:

- користувач готує захищений байт-код для виконання у хмарі з можливістю map-reduce;
- користувач реєструється в хмарній системі за допомогою централізованого серверу реєстрації;
- система авторизації дозволяє клієнту задіяти певну кількість обчислювального ресурсу;
- система авторизації дозволяє серверу зареєструватися в хмарі і надати певну кількість обчислювальних ресурсів іншим клієнтам;
- користувачі P2P обчислювальної хмари надають вільні ресурси для виконання байт-коду інших користувачів у віртуальному середовищі;

- користувач розміщує програму і дані у захищеному вигляді в балансер для виконання у хмарі;
- балансер розміщує задачі користувача на вільних ресурсах P2P хмари;
- результати обчислень у захищеному вигляді передаються користувачеві;
- користувач має можливість розмістити стільки задач, скільки ресурсу він надав протягом виконання задач від інших користувачів за попередній час;
- клієнти можуть будувати власні сегменти хмари і розподіляти виконання задач тільки на певній множині серверів.

- **Протокол обміну даними.** Протокол обміну даними між клієнтом і хмарною системою має такі компоненти:

- дані для авторизації клієнта у хмарі;
- опис у форматі JSON або XML захищеного байт-коду для виконання у хмарній системі;
- опис вхідних даних у захищеному вигляді у форматі JSON або XML;
- опис вихідних даних у захищеному вигляді у форматі JSON або XML;

Лістинг 1 - Запит до P2P хмари на XML для завантаження байт коду.

```
<?xml version="1.0" encoding="UTF-8" ?>
<authorization login="user1" password="093lkdfg">
<bytecode>
DFS23AFFS2IOG...
</bytecode>
</xml>
```

Лістинг 2 - Приклад запиту до P2P хмари на XML для завантаження вхідних даних.

```
<?xml version="1.0" encoding="UTF-8" ?>
<authorization login="user1" password="093lkdfg">
<inputuserdata>
<freeuserdatastructure>
...
</freeuserdatastructure>
```

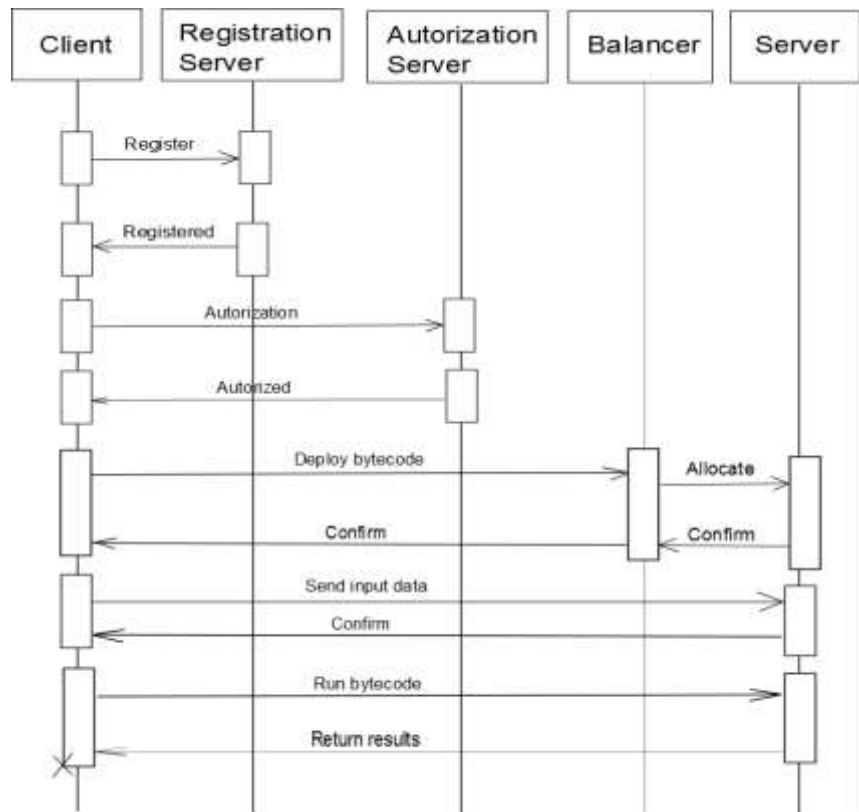


Рис. 2 – UML діаграма послідовності обміну даними у P2P хмарній системі

```
</inputuserdara>
</xml>
```

Лістинг 3 - Приклад запиту від P2P хмари до клієнта на XML для отримання вихідних даних.

```
<?xml version="1.0" encoding="UTF-8" ?>
<authorization login="user1" password="093lkdfg">
<serverside>
<resources computation_credits=10>
</serverside>
<outputuserdata>
<freeuserdatastructure>
...
</freeuserdatastructure>
</outputuserdara>
</xml>
```

**Парадигма паралельних обчислень у P2P хмарі.** Добре відома парадигма паралельних обчислень map-reduce може бути використана для організації обчислень у P2P хмарі. За такою парадигмою користувач розміщує певну кількість задач у хмарі і отримує результати розрахунків.

Такий цикл може бути повторений клієнтом багато разів в асинхронному режимі, тобто цикл розміщення задач (map) дозволяє одночасне розташування у хмарі різноманітних задач.

З точки зору паралельних алгоритмів, для такої архітектури хмари найкраще будуть виконуватися задачі великої гранулярності з мінімальною кількістю обмінів між паралельними задачами.

**Приклад системи нарахування рейтингу у P2P хмарі.** Проста концепція для нарахування рейтингу у P2P хмарі може бути використана для збалансованого розподілу ресурсів між учасниками. За кожну одиницю обчислювального ресурсу ( $C_{\text{наданий}}$ ) в хмарі користувач може отримати  $\frac{1}{2}$  одиниці ресурсу ( $C_{\text{використаний}}$ ). Такий механізм дозволить мотивувати користувачів хмари надавати вільні ресурси іншим користувачам певний проміжок часу, щоб отримати у необхідний момент пікову обчислювальну потужність. За кожну одиницю обсягу переданих вхідних та вихідних даних ( $R_{\text{наданий}}$ ) користувач зможе отримувати  $\frac{1}{2}$  одиниці обсягу для власних даних ( $R_{\text{використаний}}$ ).

$$C_{\text{використаний}} = \frac{1}{2} C_{\text{наданий}}$$

$$R_{\text{використаний}} = \frac{1}{2} R_{\text{наданий}}$$

Певна кількість балів рейтингу для учасника P2P хмари зараховується тільки у разі повного виконання байт коду задачі і надання вихідних результатів клієнту. Дані в хмарі зберігаються тимчасово згідно встановленого параметру і певний учасник хмари не гарантує збереження або успішне виконання байт коду користувача. У разі виявлення помилки виконання байт коду, балансер автоматично розміщує задачу на іншому вузлі P2P хмари.

**Планування завантаження серверів в хмарі та балансер.** Для рівномірного розподілу завантаження вузлів у P2P хмарі планувальник завантаження (балансер) розміщує задачі у наявних серверах, які авторизовані у системі. Кожен клієнт, що

авторизувався в хмарі автоматично стає вузлом P2P хмари та надає свої власні ресурси іншим клієнтам згідно встановлених квот по завантаженню процесорів, максимального часу виконання задачі та кількості переданих даних.

Користувачі, які мають меншу кількість балів рейтингу у системі, мають більшу ймовірність отримати задачу на розрахунок від балансера, таким чином найновіші учасники хмари зможуть надати власні ресурси в першу чергу і мати змогу розмістити свої задачі в найближчий час.

Віртуальна машина на серверах учасників P2P хмари не дозволяє обмінюватися даними безпосередньо від сервера до сервера, такий обмін можливий тільки через протокол обміну вхідними та вихідними даними через клієнтську частину.

**Відкритий код P2P хмари та безпека використання системи.** Пропонується реалізувати описану архітектуру P2P хмари у вигляді системи з відкритим кодом. Це дозволить учасникам чітко відстежувати потенційні проблеми реалізації та надавати суспільному проекту P2P хмари нові версії системи, згідно ліцензуванню за допомогою GNU Public License, Creative Common License, BSD License і т.д.

Байт код передається у віртуальну машину у зашифрованому вигляді і дешифрується частинами безпосередньо перед виконанням за допомогою циклічного коду від користувача. Безпека виконання байт коду програми реалізується за рахунок засобів віртуальної машини, яка не дозволяє отримати доступ до даних під час виконання програми, наприклад [5].

**Висновки.** P2P архітектура PaaS хмарної системи спільного використання може бути застосована у різноманітних реалізаціях відкритих, приватних або корпоративних обчислювальних платформах. В хмарній системі може бути використана віртуальна машина Java, .NET, Sandboxed Python або інші, в залежності від вимог користувачів, що створюють P2P хмарну систему. В конфігураціях для персонального використання P2P хмарні системи дозволяють об'єднувати потужності наявних обчислювальних пристроїв з персональними комп'ютерами, мобільними пристроями, файловими серверами.

На відміну від проектів розподілених обчислювальних систем, таких як SETI@HOME, користувачі можуть розміщувати власні обчислювальні задачі у хмарній P2P системі та надавати іншим користувачам ресурси своїх обчислювальних систем.

**Список літератури:** 1. Advanced Peer-to-Peer Networking (APPN) [Електронний ресурс]. – Режим доступу: [http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp?topic=/com.ibm.zos.znetwork/znetwork\\_181.htm](http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp?topic=/com.ibm.zos.znetwork/znetwork_181.htm). 2. A Practical Distributed Mutual Exclusion Protocol in Dynamic Peer-to-Peer Systems (2004) by *Shiding Lin, Shi-ding Lin, Ming Chen, Zheng Zhang, Qiao Lian*, In 3rd International Workshop on Peer-to-Peer Systems (IPTPS'04) 3. *Antony Rowstron, Peter Druschel*, Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems, MIDDLEWARE, 2001, 329—350 pp. 4. Improving P2P in Cloud Computing based on Performance by *Sarada B. S.* [Електронний ресурс]. – Режим доступу: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.206.3360>. 5. System and method to protect java bytecode code against static and dynamic attacks within hostile execution environments, US PATENT WO 2011057393 A1. 6. JXTA 2: A high-performance, massively scalable P2P network [Електронний ресурс]. – Режим доступу: <http://www.ibm.com/developerworks/library/j-jxta2/>. 7. JMaay : p2p Java Library [Електронний ресурс]. – Режим доступу: <http://sourceforge.net/projects/jmaay/>. 8. XMPP Standards Foundation [Електронний ресурс]. – Режим доступу: <http://xmpp.org/about-xmpp/faq/>. 9. openkad, Java implementation of the Kademlia DHT [Електронний ресурс]. – Режим доступу: <https://code.google.com/p/openkad/>. 10. P2P-MPI home

page [Електронний ресурс]. – Режим доступу: <http://grid.u-strasbg.fr/p2pmpi/>. 11. SETI home page [Електронний ресурс]. – Режим доступу: <http://setiathome.berkeley.edu/>. 12. Douglas F. Parkhill. The Challenge of the Computer Utility. Addison-Wesley Publishing Company, 1966 – 207 pp. ISBN 0-201-05720-4 13. The NIST Definition of Cloud Computing, by Peter Mell and Timothy Grance, NIST Special Publication 800-145, [Електронний ресурс]. – Режим доступу: <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>. 14. BOINC home page [Електронний ресурс]. – Режим доступу: <http://boinc.berkeley.edu>.

*Надійшла до редколегії 15.09.2013*

УДК 004.7

**Архитектура открытой P2P облачной системы / Харченко К. В. . // Вісник НТУ «ХП».** Серія: Нові рішення в сучасних технологіях. – Х: НТУ «ХП», – 2013. - № 56 (1029). – С.167-172 . – Бібліогр.: 14 назв.

Предлагается архитектура открытой облачной системы на основе P2P сети. Реализация такой архитектуры позволяет построение приватной или публичной облачной системы. Описана концепция построения P2P сети и взаимодействие клиент-облако. Описан общий протокол обмена данными между клиентом и облаком.

**Ключевые слова:** высокопродуктивные вычислительные системы, облачные вычисления, P2P системы, общие облачные системы.

Architecture of an open cloud systems based on P2P networks is described. An implementation of this architecture allows constructing public or private cloud systems. The concept of building P2P network and client-cloud interaction is proposed. A basic protocol for data exchange between customers and cloud system is described.

**Keywords:** high-performance computing, cloud computing, P2P systems, community cloud.

УДК 004.94

**О. Ю. МУЛЕСА**, викладач, ДВНЗ «УжНУ», Ужгород

## **ТЕХНОЛОГІЯ КІЛЬКІСНОГО ОЦІНЮВАННЯ ПРЕДСТАВНИКІВ ГРУП ВИСОКОГО РИЗИКУ ІНФІКУВАННЯ ВІРУСОМ ІМУНОДЕФІЦИТУ ЛЮДИНИ В УМОВАХ НЕВИЗНАЧЕНОСТІ**

Розглядається задача оцінювання міри належності особи до групи високого ризику інфікування вірусом імунодефіциту людини. Розроблено технологію кількісного оцінювання представників таких груп з урахуванням висновків експертів та нечіткого характеру вхідних даних.

**Ключові слова:** інформаційна технологія; група високого ризику інфікування вірусом імунодефіциту людини; інформаційно-аналітична система.

**Вступ.** Моделі і методи кількісного оцінювання об'єктів є предметом багатьох досліджень, які ведуться у напрямках узагальнення властивостей об'єктів, побудови нових моделей, оцінювання характеристик об'єктів, що застосовується для розв'язання прикладних задач у різних науково-прикладних галузях. Зокрема, задачі такого типу виникають у сфері профілактики та боротьби з ВІЛ/СНІДом при проведенні заходів з охоплення медико-соціальними послугами представників груп високого ризику інфікування ВІЛ (ГВР) [1]. Проблемними задачами, які виникають в цьому контексті, є задачі оцінювання кількості осіб з певної території (населеного пункту чи регіону), що належать до ГВР та задачі визначення для заданої особи, яка характеризується набором індивідуальних ознак, міри її належності до ГВР [2-3]. При визначенні кількості осіб, що належать до ГВР, спеціалісти установ профілактики та боротьби з ВІЛ, як правило, керуються результатами соціологічних

© О. Ю. МУЛЕСА, 2013