

Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра програмного забезпечення

## ІЛЮСТРАТИВНИЙ МАТЕРІАЛ

до захисту магістерської кваліфікаційної роботи

**Янкового Івана Олександровича**

на тему:

**"РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ РОЗПОДІЛЕННЯ  
ОБЧИСЛЮВАЛЬНИХ РЕСУРСІВ У МЕРЕЖІ INTERNET"**

група 1ПІ-17м

спеціальність

**121 – Інженерія програмного забезпечення**

Науковий керівник:

к.т.н., доцент кафедри ПЗ

**Кательніков Д. І.**

**Об'єкт дослідження:** процес створення розподілених обчислювальних систем з використанням мережі Інтернет.

**Предмет дослідження:** алгоритми та програмне забезпечення для створення кросплатформеного децентралізованого сервіс-орієнтованого розподіленого обчислювального середовища з гнучкою транспортною системою та шифруванням.

**Мета роботи:** полягає в підвищенні ефективності використання розподілених обчислювальних ресурсів.

**Наукова новизна:** в роботі отримали подальший розвиток методи організації розподілених обчислювальних систем з врахуванням сучасних можливостей операційних систем та мережі Інтернет.

***В рамках МКР виділено наступний перелік завдань:***

- 1.**Спроекувати і реалізувати архітектуру кроссплатформенної децентралізованої сервіс-орієнтованого розподіленого обчислювального середовища.
- 2.**Реалізувати можливість вирішення завдань потокової обробки даних.
- 3.**Реалізувати можливість спілкування по довільним протоколам передачі даних.
- 4.**Реалізувати можливість взаємодії сервісів шляхом обміну довільними серіалізованими даними.
- 5.**Розробити засоби для розгортання та підтримання роботи розподіленого середовища.

***Практична цінність:*** система може застосовуватися в якості веб-сервера, системи обліку користувачів та надання їм функцій файлового сховища, веб-сервісу обробки зображень, сервісу відеоконференцій, для виконання різного роду обчислень потребуючих великої кількості ресурсів.

***Структура та обсяг роботи:*** магістерська кваліфікаційна робота складається з п'яти розділів. В першому розділі проводиться аналіз стану проблеми, порівняльний аналіз існуючих реалізацій та постановка задачі розробки. В другому розділі розробляються архітектура та алгоритми. В третьому розділі розробляються модулі програмного забезпечення. В четвертому розділі проводиться тестування системи. В п'ятому розділі проводиться розрахунок ефективності вкладених інвестицій та періоду їх окупності.

***Розподілені обчислення*** (розподілена обробка даних) — спосіб розв'язання трудомістких обчислювальних завдань з використанням двох і більше комп'ютерів, об'єднаних в мережу.

Розподілені обчислення є окремим випадком паралельних обчислень, тобто одночасного розв'язання різних частин одного обчислювального завдання декількома процесорами одного або кількох комп'ютерів. Тому необхідно, щоб завдання, що розв'язується було сегментоване — розділене на підзадачі, що можуть обчислюватися паралельно.

При цьому для розподілених обчислень доводиться також враховувати можливу відмінність в обчислювальних ресурсах, які будуть доступні для розрахунку різних підзадач. Проте, не кожне завдання можна «розпаралелити» і прискорити його розв'язання за допомогою розподілених обчислень.

## ***HPC (High Performance Computing, Високопродуктивні обчислення)***

HPC-системи зазвичай виконують близькозв'язані паралельні завдання, які має сенс запускати на обчислювальній системі зі з'єднаннями, що мають досить невеликі значення латентності

## ***HTC (High Throughput Computing, Обчислення високої пропускної здатності)***

HTC-системи, навпаки, призначені для незалежних, послідовних завдань, виконання кожної з яких можна планувати незалежно на великій кількості обчислювальних ресурсів, що входять в різні адміністративні організації

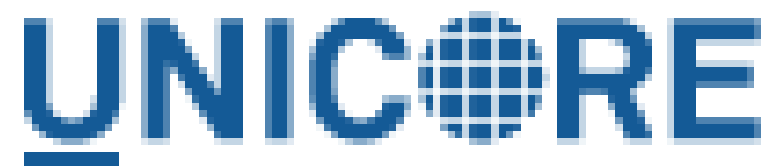
Закриті  
комерційні  
«хмари-гіганти»

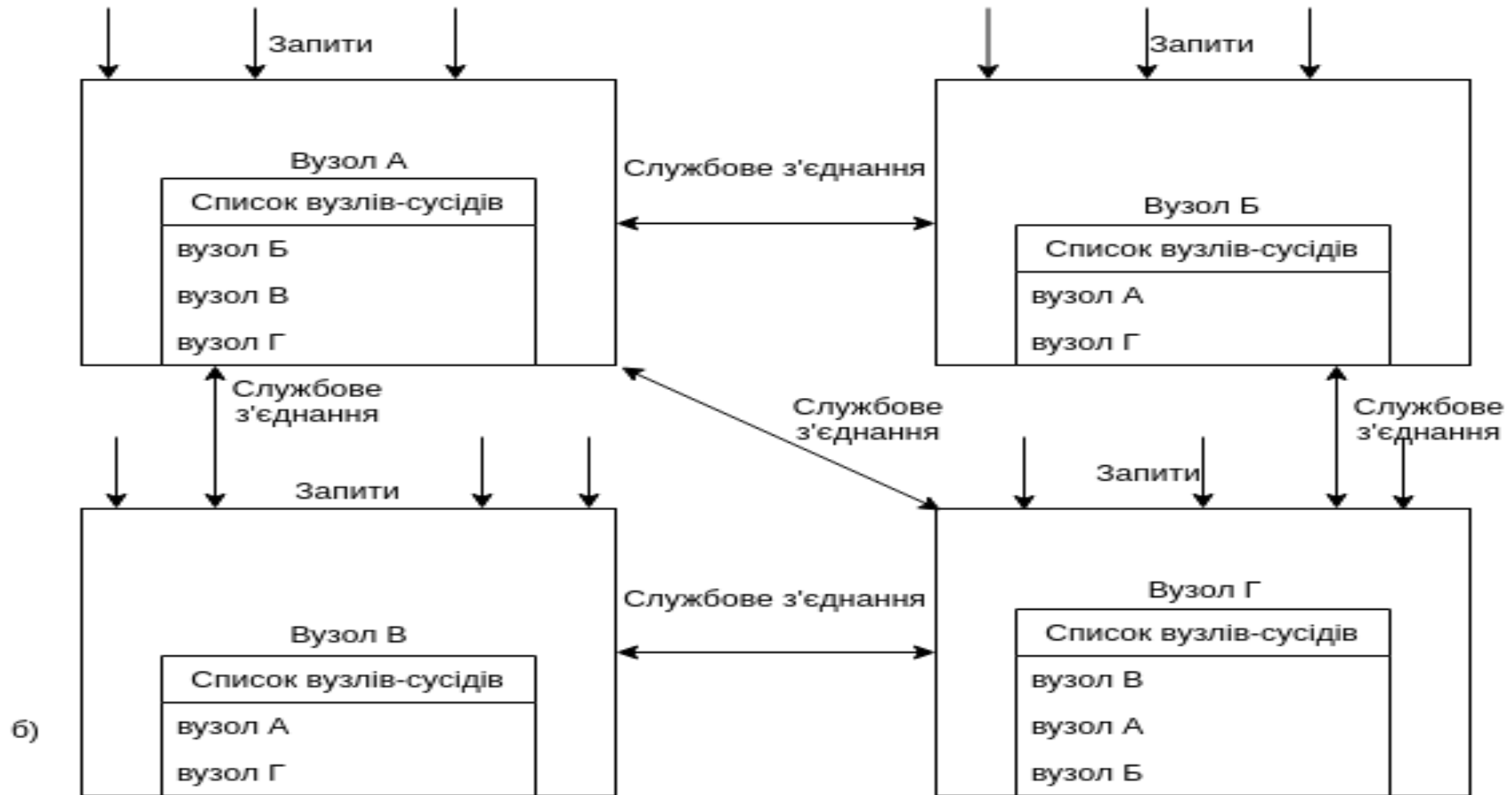


Продукти з відкритим  
вихідним кодом і  
вільною ліцензією



globus







## Запуск команди HELP

```
# (stuck? type HELP)> HELP
HELP = List of Available Commands
-----
HELP                displays this message
LIST                displays a list of all nodes
EXEC                executes a command on all nodes
DIST:<file to split> <cmd> distributes a job on all nodes and returns data
GETBUFFER           pulls any output from a node
CLEARBUFFER        clears any buffered output from a node
PUSH:<filename>     push a file to remote node(s)
PULL:<filename>     pull a file from remote node(s)
NEIGHBORS           displays a list of all neighbor nodes
SHELL:#            opens remote shell on client #
NODE:# <cmd>        issues another command only to node #
MESH:# <cmd>        send message/cmd to remote node
SCAN:<ip>:<start port>:<stop port> perform port scan
WGET <url>          download file from URL
EXIT               shut down the nodes
QUIT              EXITS server

# (stuck? type HELP)> █
```

**Результат - команда виконалась успішно та повернула список можливих команд.**

## *Результати оцінювання комерційного потенціалу розробки*

Сума балів	СБ <sub>1</sub> = 43	СБ <sub>2</sub> = 44
Середньоарифметична сума балів $\overline{СБ}$	= 43,5	

Отже, з отриманих даних видно, що **нова розробка має високий рівень комерційного потенціалу.**

В результаті впровадження результатів наукової розробки витрати на виготовлення програмного продукту зменшаться на 35 грн (що автоматично спричинить збільшення чистого прибутку підприємства на 35 грн), а кількість користувачів, які будуть користуватись збільшиться: протягом першого року – на 150 користувачів, протягом другого року – на 125 користувачів, протягом третього року – 100 користувачів.

*Реалізація програмного продукту до впровадження результатів наукової розробки складала **500 користувачів**, а прибуток, що отримував розробник до впровадження результатів наукової розробки – **250 грн**.*

Оскільки  $E_B = 94\% > T_{\text{мін}} = 0,3 = 30\%$ , то у **інвестор буде зацікавлений вкладати гроші в дану наукову розробку**.

Термін окупності вкладених у реалізацію наукового проекту інвестицій  $T_{\text{ок}} = 10,94 = 1,06$  року.

Обрахувавши термін окупності даної наукової розробки, можна зробити висновок, що **фінансування даної наукової розробки буде доцільним**.

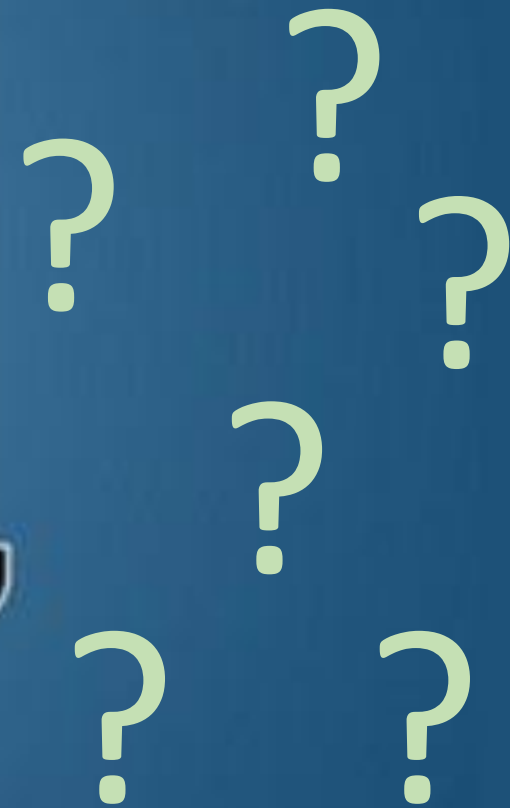
## *В результаті проведеної роботи були отримані наступні результати:*

- Була спроектована і реалізована архітектура кроссплатформенного децентралізованого сервіс-орієнтованого розподіленого обчислювального середовища.
- Надані можливості вирішення в «хмарі» завдань, пов'язаних з потоковою обробкою даних.
- Спроектована концепція транспортів і транспортних каналів, що дозволила підтримати можливості спілкування з довільним протоколам передачі даних. Були представлені приклади сервісів, що працюють по протоколах TCP і HTTP.
- Для забезпечення можливості активної взаємодії сервісів, в тому числі сервісів різних видів, які вирішують різні завдання, спроектована система передача довільних серіалізованих даних між сервісами, що призвело до архітектури, в якій різні сервіси спільно сприяють вирішенню поставлених перед ними завдань.
- Показана програмна реалізація сервісу та проведене тестування усіх сервісів.
- Проведено економічне обґрунтування доцільності даної розробки.

## Отримана система може застосовуватися в якості:

- веб-сервера
- системи обліку користувачів та надання їм функцій файлового сховища
- веб-сервісу обробки зображень, сервісу відеоконференцій

## запитання?



```
import random

guesses_made = 0

name = raw_input('Hello! What is your name?\n')

number = random.randint(1, 20)
print 'Well, (0), I am thinking of a number between 1 and 20'

while guesses_made < 6:

    guess = int(raw_input('Take a guess: '))

    guesses_made += 1

    if guess < number:
        print 'Your guess is too low'

    if guess > number:
        print 'Your guess is too high'

    if guess == number:
        break

if guess == number:
    print 'Good job, (0)! You guessed my number in (1) guesses!'
else:
    print 'Nope. The number I was thinking of was (0)'
```