

ДОДАТОК А

Перелік джерел посилання за науковими напрямками керівника та науковців
кафедри програмної інженерії

7. Sharonova N., Kyrychenko I., Shapovalova D. Comparative Analysis of Instant Messaging Protocols and Technologies for Effective Communication in Computer-Mediated Environments CEUR Workshop Proceedings, 2023, 3396, pp. 102–117.

14. Kachko O., N. Bilous , Semerkov V. Research on methods for secure web applications development Information Technologies in Innovation Business (ITIB), 7–9 October, 2015, Kharkiv, Ukraine Proceedings of ITIB, p.26–27, ISBN 978–966–659–214–2.

ДОДАТОК Б

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ



Ім'я користувача:
Кардаш Євген Вікторович каф.ПІ

ID перевірки:
1016305117

Дата перевірки:
31.05.2024 20:23:24 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
31.05.2024 20:27:40 EEST

ID користувача:
100013622

Назва документа: 2024_М_ПІ_ІПЗм-22-2_Светлінський_О_А_скорочений

Кількість сторінок: 53 Кількість слів: 10334 Кількість символів: 75884 Розмір файлу: 1,003.87 KB ID файлу: 1016101345

1.83%
Схожість

Найбільша схожість: 1.53% з джерелом з Бібліотеки (ID файлу: 1016091000)

0.3% Джерела з Інтернету

10

Сторінка 55

1.63% Джерела з Бібліотеки

42

Сторінка 55

0% Цитат

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

3

Рисунок Б.1 – Результат перевірки кваліфікаційної роботи магістра на плагіат

ДОДАТОК В

Слайди презентації

Харківський національний університет радіоелектроніки

Дослідження ефективності та оптимізації технологій міжпроцесної взаємодії у серверних додатках

Виконав: ст.гр. ІПЗм-22-2 Светлінський О.А.

Керівник: проф. каф. ПІ Лєсна Н. С.

Актуальність дослідження

Спостерігаючи швидкі темпи розвитку інформаційних технологій, необхідно зосередити увагу на питаннях оптимізації та підвищення ефективності. Зростання обсягів обробки даних та збільшення завдань, породжують необхідність поліпшення багатьох аспектів, одним з яких є міжпроцесна взаємодія. Застосування відповідних технічних рішень при міжпроцесній взаємодії дозволяє підвищити пропускну здатність, скоротити час відгуку та оптимізувати використання ресурсів, покращити прозорість роботи програми, що безпосередньо впливає на кінцевий результат, який бачить користувач.

Мета роботи

Метою даної роботи є глибоке порівняння існуючих протоколів спілкування між процесами у контексті серверних додатків, вивчення потенційних напрямків вдосконалення цих технологій. Аналіз та ефективна оптимізація технологій міжпроцесної взаємодії - важливий етап забезпечення стабільної та продуктивної роботи серверних додатків у сучасному високотехнологічному середовищі.

3

Постановка задачі

Для того, щоб отримати результат, що б задовольнив мету, необхідно вирішити набір наступних питань:

- проаналізувати існуючі протоколи та технології;
- визначитись з методами та критеріями;
- розробити програмний застосунок для проведення експериментів;
- отримані дані використати для виміру обраних метрик для порівняння технологій;
- проаналізувати отримані результати;
- запропонувати потенційне розширення дослідження та охарактеризувати доречність роботи у майбутньому.

4

Аналіз предметної області

Qt RO

Unix Sockets

Message Queue

Shared Memory

D-Bus

TCP

FIFO

Pipes

5

Аналіз предметної області

D-Bus	Специфічна, але можливо інтегрувати. Має унікальні переваги
Message Queue	Базується на Shared Memory, більш складна за FIFO, використовує повідомлення
FIFO	Стандартна технологія IPC
TCP	Зазвичай використовується при роботі в мережі, але можна і як IPC
Unix	Технологія сокетів, як і TCP, але працює тільки як IPC
Qt Remote Objects	Екстремально специфічна
Shared Memory	Є загальним принципом, ефективність на 95% залежить від конкретної реалізації
Pipes	Те ж саме, що FIFO, але примітивніша

6

Критерії дослідження

Критерій швидкості обміну даними

Чи не найважливіша характеристика технології міжпроцесорної взаємодії - це швидкість передачі даних. Чим більше даних передано за одиницю часу – тим краще.

Критерій безпеки у використанні

Чи може розробник, написавши неякісний код, завдати шкоди системі та процесам окрім поточного. Чи можуть статися витoki пам'яті.

Критерій величини затримки отримання повідомлення

В ідеальній ситуації повідомлення передавалися б миттєво, але завжди існує затримка. Це є окремою характеристикою від швидкості передачі, бо на неї впливають інші чинники.

Критерій легкості в інтеграції

Яким б не була ефективна та зручна технологія, якщо її неможливо інтегрувати або це потребуватиме рефакторингу більшої частини коду - її не можна назвати найкращою альтернативою.

7

Функціональні вимоги

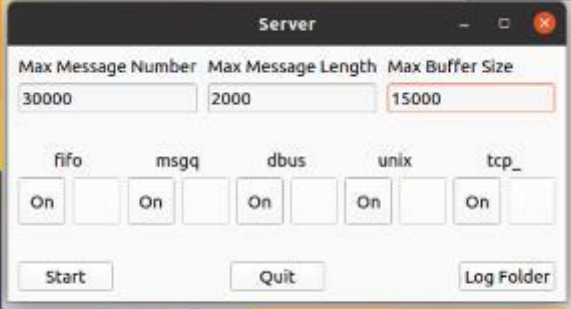
Тестовий додаток потрібен мати наступний реалізований функціонал:

- користувач обирає кількість повідомлень, яку відправлятиме додаток;
- користувач може виставити обмеження у розмірі повідомлення та буферу;
- користувач обирає яку технологію міжпроцесної взаємодії використовуватиме система;
- користувачеві повинно показуватися стан додатку;
- користувач бачить шлях за яким зберігаються файли-звіти;
- користувач може скачати загальні чи специфічні для технології файли у форматі .csv для подальшого аналізу.

8

Початок роботи

Інтерфейс користувача Обмін повідомленнями



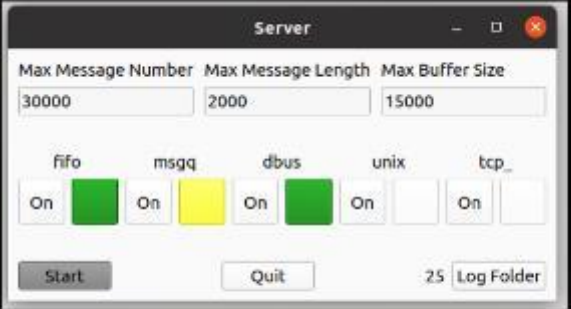
The screenshot shows the 'Server' application window with the following configuration:

Max Message Number	Max Message Length	Max Buffer Size
30000	2000	15000

Communication protocols and their status:

Protocol	Status
fifo	On
msgq	On
dbus	On
unix	On
tcp_	On

Buttons: Start, Quit, Log Folder



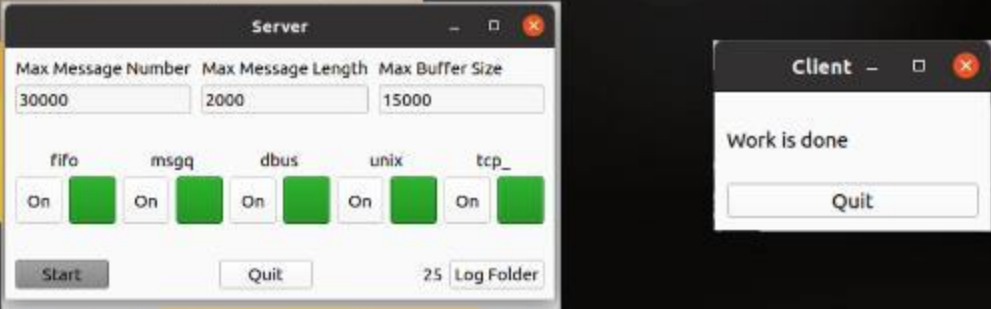
The screenshot shows the 'Server' application window with the 'Start' button highlighted in grey, indicating it is disabled. The configuration remains the same as in the previous screenshot.

Buttons: Start, Quit, 25 Log Folder

9

Кінець роботи

Інтерфейс користувача



The screenshot shows the 'Server' application window with the following configuration:

Max Message Number	Max Message Length	Max Buffer Size
30000	2000	15000

Communication protocols and their status:

Protocol	Status
fifo	On
msgq	On
dbus	On
unix	On
tcp_	On

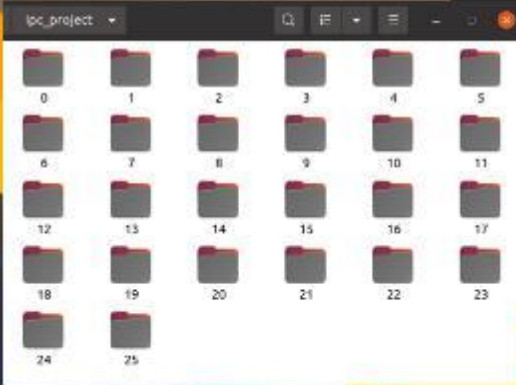
Buttons: Start, Quit, 25 Log Folder

The 'Client' application window is also visible, displaying the message 'Work is done' and a 'Quit' button.

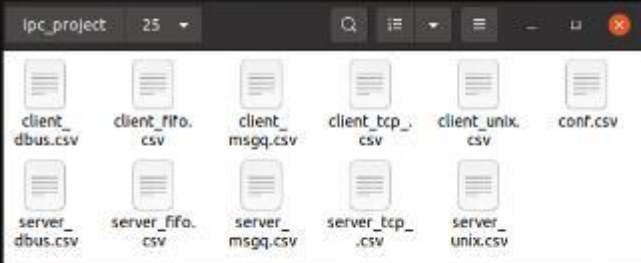
10

Інтерфейс користувача

Сховище результатів



Файли з даними



11

Інтерфейс користувача

Приклад загального конфігураційного файлу

```
oleh@oleh-VirtualBox:~/ipc_project/1$ cat conf.csv
ipc_type,overall_transfer_time(μs),message_number,message_data_length,buffer_size(byte)
dbus,795352,10000,500,1500
fifo,461969,10000,500,1500
msgq,477738,10000,500,1500
tcp,536097,10000,500,1500
unix,2740423,10000,500,1500
```

Приклад файлу для FIFO

```
oleh@oleh-VirtualBox:~/ipc_project/1$ cat server_fifo.csv | head
id,data,time_created(μs),time_transferred(μs),time_diff(μs)
1,lnNVXln5YvodA67Iz50ydyvFJe880cRqFpCszYAKGc1CZozpsaTPXYZzUNExf187xTjnp1LVAswKULACNU01qnXB0ZyV7oTt50WkYfUWwGrpSg8CwYsZut0Jf1HI0yDrKwF
6uuSHTSeV5XhhAajcaNhFKXKekjn8Vo2DYLHodqBt0wEPxepACuIX2n5CP3v1wz05A3Yq2GuFwMsIQXI0G0o61jYFbK6L7KG7Cco41MX3GF8wHLhVnblu90L8KU5va2V3eP20
I74D0XbscTRFY08Bah0tnhIsvSBt2xub8eQMsala2Fp5ot2cJXrBB09t7P5rF6oD6DnpbzFT7TWoDv1acYbrvIokzIptUqPw,1716876484958894,1716876484951358,464
2,14NoBfgrszFvIoJBAjsVrV3emqVKbNC3So1xLgdBlHh0Wj0SP1kkbFN801I1nKv5zxtBHK4qXc41BNoQp01GRCEI1N9LVuEIVvEa696Tb8CcGosv0h1In6QFIW0c6TjPEmp8
NnYm1TsSD06qyB30YITjs29Y1pJJ4Rnfn6D7o8Cf1BMXAPLYEctxUeJcJ51BK6gL81GfeIKDFgaFtjbyC4jWmsze0Yfn5CsjkxnC5uG8Pg88EauGUarANL9kqg2qM909jR3UX
p0ZCnd0pM6Gfyd2o6LYbCUJvzFl02xzFzN1bLNEV6I1vnrXq2v55Fwq31fju7WNI1tht6KEI33hUL9yotrYweCpV1M01qP7E,1716876484958981,1716876484951382,401
3,8ev5EyxrrYF5v2uEjI1DJeXrrYgNnbQK59f8xrtfpyvAoFD0m4PvYka094PKVdK0cpK043thqc1U8jgdTzWr1TjE22hCE7qPLK621WvHTNn1oXIdsx08W530x45vkf4Jmx
dYfxoYKfU1eLHhvxhrtM0duVL6AyLgvM208gnGk77KDRa0vMhCJ61chvc3e6ouu3m0gt8do9I-Oj19oAGx45N9ep3IjfwVYZn26eUuZcas881q6rb1rpyQSpY2LKNhzhYlQsWh
o6cop8lmsfU5cnY757Y6ydznyR0Wj1R9f47GhuzCCscnGX6Dv010qbKXdsno7LH1hMY9YPr02xhwY6nnIREto1eMsGTS,1716876484951898,1716876484951412,314
```

12

Скрипт, що аналізував дані

Додатковий функціонал

Результат роботи скрипта

```

File Edit Format Run Options Window Help
import csv
import random as rd

ipcs = ['dbur', 'fifer', 'magg', 'top', 'mick']

bytes_per_ms = [3, 6, 9, 0, 0, 0]
msg_max_time_ms = [0, 3, 0, 0, 0]
dir_id = 15
dir_max = 0

for i in range(0, 1000):
    dte = rd.choice(['dbur', 'fifer', 'magg', 'top', 'mick'])
    for ipc in ipcs:
        dir = '%s\\%s\\%s\\%s\\%s\\%s' % (ipc, dte, ipc, ipc, ipc, ipc)
        df = rd.choice(['server', 'ipc'])
        full_time = df['overall_transfer_time'] * 1000 * 1000
        msg_length = df['message_data_length'] * 1000 * 1000
        msg_number = df['message_number'] * 1000 * 1000
        buffer_size = df['buffer_size_bytes'] * 1000 * 1000
        total_bytes = msg_length * msg_number
        bytes_per_ms = total_bytes / full_time
        msg_max_time = df['max_time'] * 1000 * 1000
        msg_time_per_byte = msg_max_time / msg_length
        print('server: ' + ipc + '.csv')
        print('FullTime: ' + full_time)
        print('MsgLength: ' + msg_length)
        print('MsgNumber: ' + msg_number)
        print('')
        print('BytesPerMs: ' + bytes_per_ms)
        print('MsgMaxTime: ' + msg_max_time)
        print('MsgTimePerByte: ' + msg_time_per_byte)
        print('')
        print('-----')
        print('')
    bytes_per_ms[ipcs.index(ipc)] += bytes_per_ms * 1000 / 1000
    msg_max_time[ipcs.index(ipc)] += msg_max_time

```

```

File Edit Shell Debug Options Window Help
...
server_magg.csv
FullTime: 475122
MsgLength: 500
MsgNumber: 10000
BytesPerMs: 10.523612882389748
MsgMaxTime: 10.27655735853337
MsgTimePerByte: 37.7428
MsgTimePerByte: 1.46484375
-----
server_top.csv
FullTime: 407746
MsgLength: 500
MsgNumber: 10000
BytesPerMs: 10.251163433363367
MsgMaxTime: 10.81387388934421
MsgTimePerByte: 786.2781
MsgTimePerByte: 1.46484375
-----
server_unix.csv
FullTime: 500020
MsgLength: 500
MsgNumber: 10000
BytesPerMs: 9.502468783493233
MsgMaxTime: 9.670272837483395
MsgTimePerByte: 74.1683
MsgTimePerByte: 1.46484375
-----

```

13

Опис проведених досліджень

Параметри передачі даних.

Було проведено по 5 тестів на кожний набір параметрів

Назва набору тестів	Кількість повідомлень	Розмір повідомлення (кількість символів)	Розмір буферу (байти)
Початковий	10000	500	1500
Кількість повідомлень	25000	500	1500
Розмір повідомлення	10000	1500	4000
Розмір буферу	10000	500	15000
Всі параметри	30000	2000	15000

14

Опис проведених досліджень

Усереднені швидкості передачі даних технологіями (Kb/ms)

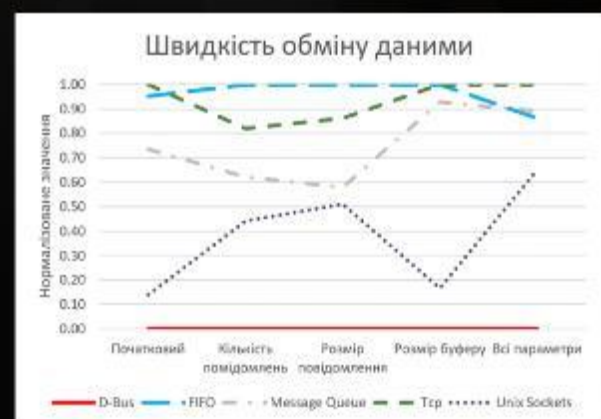
	Початковий	Кількість повідомлень	Розмір повідомлення	Розмір буферу	Всі параметри
D-Bus	5.92	6.75	10.38	5.72	11.00
FIFO	9.29	10.49	13.44	9.36	13.19
Message Queue	8.53	9.07	12.15	9.10	13.27
TCP	9.47	9.81	13.02	9.36	13.55
Unix Sockets	6.42	8.40	11.94	6.33	12.65

15

Аналіз даних



Абсолютні значення



Нормалізовані значення

16

Опис проведених досліджень

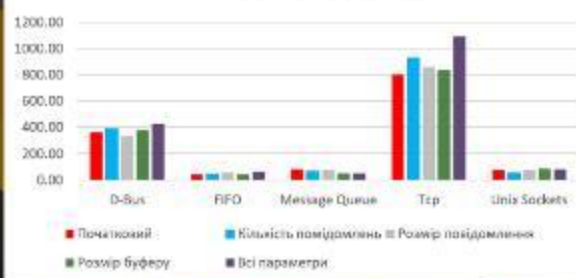
Усереднені величини затримки отримання повідомлення (µs)

	Початковий	Кількість повідомлень	Розмір повідомлення	Розмір буферу	Всі параметри
D-Bus	364.02	384.50	331.30	376.38	425.52
FIFO	43.54	40.77	55.66	43.07	59.44
Message Queue	78.96	67.28	75.55	53.23	53.76
TCP	801.16	926.44	857.59	833.94	1088.38
Unix Sockets	74.76	49.41	77.67	87.12	82.92

17

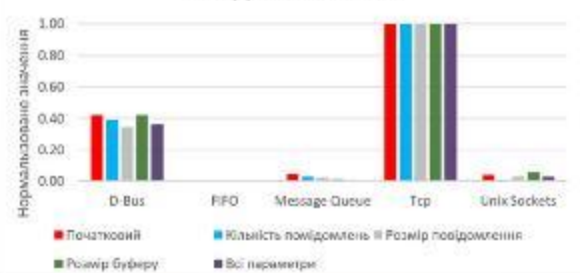
Аналіз даних

Величина затримки обміну повідомленнями (µs)



Абсолютні значення

Величина затримки обміну повідомленнями



Нормалізовані значення

18

Опис проведених досліджень

Властивості легкості в інтеграції технології

	D-Bus	FIFO	Message Queue	TCP	Unix Sockets
Масштабовані структури повідомлень	Так	Ні	Ні	Так	Так
Присутня у стандартній бібліотеці	Ні	Так	Ні	Так	Так
Не має додаткових залежностей	Ні	Так	Ні	Так	Так
Зручний API	Так	Ні	Ні	Так	Так
Виклики функцій не блокують потік	Так	Ні	Ні	Так	Так

19

Опис проведених досліджень

Властивості легкості в інтеграції технології

	D-Bus	FIFO	Message Queue	TCP	Unix Sockets
Додаткові перевірки валідності даних зі сторони інтерфейсу	Так	Ні	Ні	Ні	Ні
Перевірка стану отримувача даних	Так	Ні	Ні	Так	Так
Наявні структури повідомлень, замість роботи з байтами	Так	Ні	Ні	Ні	Ні
Буферизація даних	Так	Так	Так	Так	Так
Стійкість з'єднання між процесами	Так	Ні	Ні	Так	Так

20

Аналіз даних

Векторний опис технологій за критеріями

	D-Bus	FIFO	Message Queue	TCP	Unix Sockets
Критерій швидкості обміну даними	0	0.96	0.75	0.94	0.38
Критерій величини затримки отримання повідомлення	0.61	1	0.98	0	0.97
Критерій легкості в інтеграції	0.66	0.33	0	1	1
Критерій безпечності	1	0.13	0.13	0.6	0.6
Всього:	2.27	2.42	1.86	2.54	2.95

21

Аналіз даних

Вагомні коефіцієнти критеріїв

Критерій швидкості обміну даними	Критерій величини затримки отримання повідомлення	Критерій легкості в інтеграції	Критерій безпечності
0.4	0.25	0.1	0.25

Формула адитивної лінійної згортки:

$$X_i = \sum_{j=0}^n \alpha_{ij} \beta_j,$$

де X_i – значення згортки для i -тої альтернативи,

n – кількість критеріїв, у нашому випадку це 4,

α_{ij} – значення нормованого критерію за альтернативою,

β_j – ваговий коефіцієнт для критерію.

22

Аналіз даних

Векторний опис технологій з результатами згортки

	D-Bus	FIFO	TCP	Unix Sockets	Вагомі коефіцієнти
Критерій швидкості обміну даними	0	0.96	0.94	0.38	0.4
Критерій величини затримки отримання повідомлення	0.61	1	0	0.97	0.25
Критерій легкості в інтеграції	0.66	0.33	1	1	0.1
Критерій безпеки	1	0.13	0.6	0.6	0.25
Результати згортки:	0.47	0.70	0.58	0.63	

23

Опис сформованих рекомендацій

- За обраними критеріями та коефіцієнтами найкраща IPC – FIFO
- Слід розмежовувати IPC з високою пропускнуою здатністю, для великих повідомлень, та IPC з мінімальною затримкою, для малих сигнальних повідомлень.
- Швидкість обміну даними між IPC не є дуже суттєвою, коли величина затримки може різнитися у десятки разів.
- Якщо потрібна IPC у серці архітектури для високих навантажень – краще використовувати низькорівневі технології. Якщо для малих, чи допоміжних – високорівневі альтернативи краще, наприклад D-Bus.

24

Висновки

У результаті проведення дослідження для кваліфікаційної роботи було:

- проаналізовано предметну область та було розглянуто обрані технології міжпроцесної взаємодії напрямку передачі даних.
- знайдено переваги та недоліки для кожної з технологій
- запропоновані актуальні варіанти вимірювання кожної з представлених метрик
- сформульовані критерії та формули, які були використані для обчислення числових значень цих метрик
- проведені експерименти, після яких записані тестові дані, що були проаналізовані, оброблені та завдяки використанню адитивної лінійної згортки з використанням вагомих коефіцієнтів

ДОДАТОК Г

Апробація результатів роботи

УДК 004.415:004.2

DOI: <https://doi.org/10.30837/IYF.PIS.2024.361>

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ТА ОПТИМІЗАЦІЇ ТЕХНОЛОГІЙ МІЖПРОЦЕСНОЇ ВЗАЄМОДІЇ У СЕРВЕРНИХ ДОДАТКАХ

Светлінський О. А.

Науковий керівник – д.т.н., доц. Чуприна А. С.

Харківський національний університет радіоелектроніки, каф. ПІ

м. Харків, Україна

e-mail: oleh.svietlinskvi@nure.ua

This work aims to research and analyze the efficiency and optimization of existing interprocess interaction technologies, specifically the mechanisms provided by an operating system, and to select criteria for this research. The focus of the study is on the technology of interprocess interaction during data transfer within a single system, specifically a server application. The existing technologies and protocols for inter-process data transfer were evaluated, including their features, advantages, and disadvantages, as well as their working principles. Methods of comparison were described and demonstrated, and formulas for calculating numerical indicators were proposed as a result of this work.

У сучасному світі, де технології та інформація вже стала невід'ємною частиною життя, ефективність та оптимізація технологій міжпроцесної взаємодії є однією з основних частин розвитку галузі. Попередні покоління обчислювальних машин могли обробляти десятки програм чи процесів, але сьогодні ця кількість може перевищувати сотні, або навіть тисячі [1]. Зростання обсягів обробки даних та збільшення завдань, що стоять перед серверними програмами, вимагають удосконалень у багатьох аспектах, одним із яких є міжпроцесна взаємодія.

Серед можливих проблем у зв'язку між процесами або просто характеристик, які необхідно взяти до уваги, можна зазначити наступні: стан гонитви, безпека та стабільність передачі даних, складність у розробці, накладні витрати на продуктивність [2].

Беручи до уваги усі показники вище та аналіз області міжпроцесорних технологій, було обрано наступні ключові метрики для порівняння технологій:

- безпечність у використанні: чи може розробник забезпечення, написавши неякісний код завдати шкоди даним, процесам чи системі у цілому;
- розмір накладних витрат пам'яті: свої накладні витрати оперативної пам'яті буде вводити майже кожна технологія;
- швидкість обміну даними: на цей параметр можуть впливати дуже багато факторів під час роботи операційної системи і сама архітектура технології, але у дослідженні вони були знехтувані задля спрощення результатів експериментів; Міжпроцесорна взаємодія має багато спільного із спілкуванням по мережі Інтернет, тому як і у дослідженнях з

ефективності технологій передачі даних по мережі [3], швидкість обміну даними є чи не найважливішою характеристикою;

– легкість в інтеграції коду: яким не була ефективна та зручна технологія, якщо її неможливо інтегрувати або це потребуватиме рефакторингу більшої частини коду – її не можна назвати найкращою альтернативою.

Наступним етапом є експериментальне дослідження ефективності та доцільності різних технологій міжпроцесорної взаємодії у серверних додатках. Для цього створено декілька тестових серверних додатків, що імітують генерацію записи до журналу подій системи з різним навантаженням, середнім розміром цих повідомлень, типом даних, задля отримання максимальної кількості даних; та окремих додаток, який буде приймати ці записи, використовуючи різні технології міжпроцесорної взаємодії: як низького рівня – pipe, shared memory, message queue [4], так і більш високого – tcp [5], d-bus.

Таким чином, після проведення дослідження, як його результат, буде отримано набір даних про безпечність, ефективність, швидкість та легкість у використанні для обраних технологій. За допомогою цих даних можна буде визначити найкращу технологію міжпроцесорної взаємодії для того чи іншого робочого випадку. У майбутньому планується перевірка висунутих гіпотез на прикладі справжнього серверного додатку, який приймає клієнтів. Це дозволить підтвердити отримані раніше експериментальні дані, допоможе створити рекомендації щодо створення нових технологій та їх методологію проектування.

Список використаних джерел:

1. Eric G. Swedin, David L. Ferro. *The Computer: A Brief History of the Machine That Changed the World*, 2022. 320 p.
2. Inter-Process Communication (IPC): Bridging the Gap in Operating Systems. URL: <https://www.linkedin.com/pulse/inter-process-communication-ipc-bridging-gap-operating-aritra-pain> (дата звернення: 07.01.2024).
3. Sharonova N., Kyrychenko I., Shapovalova D. Comparative Analysis of Instant Messaging Protocols and Technologies for Effective Communication in Computer-Mediated Environments CEUR Workshop Proceedings, 2023, 3396, pp. 102–117.
4. Lecture 6: Interprocess Communication. Fall 2019 Jason Tang URL: <https://redirect.cs.umbc.edu/~jtang/archives/cs421.f19/lectures/L06IPC.pdf> (дата звернення: 23.10.2023).
5. Inter-process communication in Linux: Sockets and signals. URL: <https://opensource.com/article/19/4/interprocess-communication-linux-networking> (дата звернення: 01.12.2023).

ДОДАТОК Д

Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимогам ДСТУ 3008: 2015

Експертний висновок результатів перевірки кваліфікаційної роботи

студент
(посала)

програмної інженерії
(кафедра)

Светліньський О.А.

(прізвище, ім'я, по-батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту
1	2
	7.1 Загальні положення
	7.3 Нумерація сторінок звіту
	7.5 Рисунки
	7.6 Таблиці
	7.7 Переліки
7.7.2	Якщо подають переліки одного рівня підпорядкованості, на які у звіті немає посилань, то перед кожним із переліків ставлять знак «тире». Якщо у звіті є посилання на переліки, підпорядкованість позначають малими літерами української абетки, далі — арабськими цифрами, далі — через знаки «тире». Після цифри або літери певної позиції переліку ставлять круглу дужку.
	7.8 Примітки
	7.9 Висновки
	7.10 Формули та рівняння
7.10.6	Пояснення познач, які входять до формули чи рівняння, треба подавати безпосередньо під формулою або рівнянням у тій послідовності, у якій їх наведено у формулі або рівнянні. Пояснення познач треба подавати без абзацного відступу з нового рядка, починаючи зі слова «де» без двокрапки. Позначки, яким встановлюють визначення чи пояснення, рекомендовано ви-рівнювати у вертикальному напрямку.
	7.11 Посилання
	7.13 Список авторів
	7.14 Скорочення та умовні позначки
	7.15 Додатки
6.2.2	Якщо додатки є продовженням тексту основної частини звіту, нумерація сторінок додатків — це продовження нумерації сторінок звіту. Кожний додаток повинен мати заголовок, який друкують вгорі малими літерами з першої великої симетрично до тексту сторінки. Над заголовком, але посередині рядка, друкують слово «ДОДАТОК» і відповідну велику літеру української абетки, крім літер Г, Є, З, І, І, Й, О, Ч, Ь, яка позначає додаток. Текст кожного додатка починають з наступної сторінки.

Рисунок Д.1 – Експертний висновок перевірки роботи. Перша сторінка

<p>Методичні вказівки до виконання кваліфікаційної роботи магістра... ЗАТВЕРДЖЕНО кафедрою ПІ протокол № 5 від 13.11.2023р. 3.2 Оформлення пояснювальної записки згідно з ДСТУ 3008:2015 Звіти у сфері науки і техніки. Структура та правила оформлення. Шаблон затверджений засіданням кафедри №3 від 16.10.2023.</p>	<p>Рисунок повинен розміщуватися одразу після його згадування у тексті, або на наступній сторінці. Під рисунком повинен бути підпис із словом Рисунок, порядковим номером цього рисунку, через тире з великої літери – назва рисунку та в круглих дужках вказується джерело з якого взятий цей рисунок, або то, що його виконано самостійно.</p>
<p>Методичні вказівки до виконання кваліфікаційної роботи магістра... ЗАТВЕРДЖЕНО кафедрою ПІ протокол № 5 від 13.11.2023р. 3.2 Оформлення пояснювальної записки згідно з ДСТУ 3008:2015 Звіти у сфері науки і техніки. Структура та правила оформлення. Шаблон затверджений засіданням кафедри №3 від 16.10.2023.</p>	<p>Додатки нумеруються за допомогою літер української абетки. Слово ДОДАТОК та його назва розташовуються посередині сторінки без абзацного відступу, пишеться заголовними літерами звичайним начертанням. Після заголовку ставиться один пустий рядок</p>

Експерт

_____ (підпис)

02.06.2024

Рисунок Д.2 – Експертний висновок перевірки роботи. Друга сторінка

ПЗМ-22-2
(група)

Сторінка кваліфікаційної роботи
3
44
45, далі за текстом
65, далі за текстом

Рисунок Д.3 – Експертний висновок перевірки роботи. Третя сторінка



Рисунок Д.4 – Експертний висновок перевірки роботи. Четверта сторінка