

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
« ____ » _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи зберігання
даних SAN на основі стандарту FC Gen”

КБПЗ - 2025

Виконав здобувач вищої освіти
II курсу, групи КІ-24М
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Семенюк О.А.
« ____ » _____ 2025 р.

Керівник проекту
доктор філософії (PhD)
_____ Усік П.С.
« ____ » _____ 2025 р.
Рецензент _____

АНОТАЦІЯ

Семенюк О.А. Дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи зберігання даних SAN на основі стандарту FC Gen.

Метою розробки є дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen.

Об'єктом дослідження є процес зберігання даних SAN на основі стандарту FC Gen.

Предметом дослідження є методи зберігання даних SAN на основі стандарту FC Gen.

Методи дослідження базуються на методах теорії побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, SAN, FC Gen

ABSTRACT

Semeniuk O.A. Research and software implementation of a SAN data storage system based on the FC Gen standard. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software has been developed that is intended for a SAN data storage system based on the FC Gen standard.

The purpose of the development is the research and software implementation of a SAN data storage system based on the FC Gen standard.

The object of the research is the SAN data storage process based on the FC Gen standard.

The subject of the research is the SAN data storage methods based on the FC Gen standard.

The research methods are based on the methods of the theory of computer network construction, methods of mathematical statistics, and methods of software development.

The result of the work is the software implementation of a SAN data storage system based on the FC Gen standard.

In the process of working on the software model, an analysis of existing hardware and software tools was performed. All components of the developed software are fully described.

A user-friendly user interface has been developed. Instructions for working with the software are provided.

The program can be used on a PC with Windows 10/11.

The program is developed in the Python environment.

Keywords: computer engineering, SAN, FC Gen

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	7
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	10
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	10
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	21
2.3 Розгорнута постановка завдання	26
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	27
3.1 Опис функціонування системи	27
3.2 Розробка структурної схеми.....	31
3.3 Розробка функціональної схеми	34
3.4 Розробка діаграми процесів.....	40
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	42
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	42
4.2 Захист розробленого програмного забезпечення.....	60
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	64
6 НАУКОВА НОВИЗНА	68

						ВКРМ-123.25.0060.00.00.ПЗ		
Вим	Арк.	№ докум.	Підп.	Дата				
Розроб.	Семенов О.А.				Дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen	Літ.	Аркуш	Аркушів
Перев.	Усік П.С.					М	1	93
Н.контр.	Коваленко А.С.				ЦНТУ КІ-24М			
Затв.	Смірнов О.А.							

7	МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ	69
7.1	Визначення цільової аудиторії кінцевого готового продукту	69
7.2	Оцінка привабливості шляхом застосування методів експертних оцінок ...	70
7.3	Вибір методу оцінки вартості ПЗ	70
7.4	Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості.....	71
7.5	Пропозиція алгоритму просування проєкту розробки ПЗ	73
7.6	Оптимізація каналів збуту та шляхів реалізації ПЗ	74
7.7	Визначення ключових факторів успіху конкретного проєкту.....	75
8	ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	76
8.1	Вступ.....	76
8.2	Аналіз умов праці на робочому місці ІТ-фахівця.....	77
8.3	Пропозиції щодо підвищення працездатності ІТ-фахівців.....	79
8.4	Розрахунок системи загального штучного освітлення виробничого приміщення де працюють ІТ-фахівці.....	81
8.5	Висновки до розділу.....	84
9	ОСНОВНІ ВИСНОВКИ.....	85
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	87

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

- КВ – коефіцієнт варіації
- КЗ – канал зв'язку
- ПС – програмна середа
- СеМО – експонентна мережа масового обслуговування
- СМО – система масового обслуговування
- СПД – система передачі даних

КБПЗ_2025

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

ВСТУП

Актуальність теми. У світі IT-інфраструктури сховище даних є критично важливим; саме там зберігаються дані, існують безпечні копії, і це основа продуктивності програм. Незалежно від того, скільки ядер процесора чи скільки пам'яті може мати сервер, кожен сервер очікує на дані.

Отже, сфера застосування цього середовища включає ранні дискові накопичувачі, стрічки для безпечного та економічно ефективного резервного копіювання даних, а також програмні реалізації, що забезпечують доступ, продуктивність та безпеку. Крім того, обов'язки адміністратора сховища включають захист копій даних за допомогою конфігурацій RAID або рішень для реплікації між сайтами.

Мантра адміністраторів сховищ: «одна копія будь-якого набору даних – це єдина точка відмови, яка чекає на катастрофу». Втрата даних ніколи не є прийнятним варіантом з точки зору програми.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем зберігання даних SAN на основі стандарту FC Gen.
- Дослідження системи зберігання даних SAN на основі стандарту FC Gen.
- Програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen.

Об'єктом дослідження є процес зберігання даних SAN на основі стандарту FC Gen.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Предметом дослідження є методи зберігання даних SAN на основі стандарту FC Gen.

Методи дослідження базуються на методах теорії побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод зберігання даних SAN на основі стандарту FC Gen.
- Розроблено вітчизняний продукт зберігання даних SAN на основі стандарту FC Gen, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі зберігання даних SAN на основі стандарту FC Gen.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічній конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2025 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Fibre Channel був стандартизований ще в 1994 році та призвів до появи мереж SAN. Для корпоративного середовища це було очевидним рішенням для зберігання даних, хоча воно було досить дорогим (і складнішим при підключенні географічних точок). Через деякий час iSCSI став дешевшою альтернативою, але тепер він явно проник і в корпоративне середовище. Донині це інтерфейс з низькою затримкою та високою пропускнуою здатністю (хоча мережі Ethernet все ще швидші). FC працює на тому ж рівні, що й Ethernet (L2), але вони повністю несумісні один з одним.

FC – Fibre Channel – це повнодуплексний, послідовний, блочно-орієнтований, точка-точка комунікаційний інтерфейс (мережева технологія), призначений для високошвидкісної передачі даних (швидкість 1, 2, 4, 8, 16, 32, 128 Гбіт/с, наразі переважно 16 та 32 Гбіт/с). Він зазвичай використовується для підключення дискових масивів (пристроїв зберігання даних) до серверів у мережах SAN, тому це блокова передача, і ми підключаємося безпосередньо до дискового простору (він відображається як локальний диск на сервері). Зазвичай, оптичні волокна використовуються як фізичне середовище для передачі. FC розроблений з високою надійністю, втрата кадрів неможлива, а кадри повинні доставлятися в правильному порядку.

FCP – Fibre Channel Protocol (протокол оптоволоконного каналу) – транспортний протокол, який передає команди SCSI через мережу FC. FCP працює через FC, подібно до TCP/IP через Ethernet. Порівняно з TCP/IP, FCP спеціально розроблений для однієї мети, тому він є більш оптимальним та компактним (з меншими накладними витратами).

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

FCoE – Fibre Channel over Ethernet (оптичний канал через Ethernet) –

стандарт, що дозволяє інкапсулювати кадри FC в мережі Ethernet, зазвичай з використанням мідних кабелів (вита пара мідного дроту) та швидкості 10 Гбіт/с. Це зменшує складність центру обробки даних, оскільки ми можемо використовувати стандартні комутатори та кабелі, зберігаючи при цьому властивості FC. На відміну від iSCSI, який працює через TCP/IP, FCoE працює через Ethernet (на тому ж рівні, що й IP – L3). Тому він має менше накладних витрат, але проблема полягає в тому, що він не маршрутизується через TCP/IP WAN. Він підтримує стандартні мережеві адаптери, але часто використовується спеціальний конвергентний мережевий адаптер (CNA). Cisco намагається просувати FCoE та підтримує його на комутаторах Nexus, деякі з яких підтримують Unified Fabric, тому вони мають уніфіковані порти, які можна перемикає між Ethernet, FC та FCoE. FCoE здатний заглядати в рівень FC та діє як комутатор FC.

1.2 Область застосування

Для Fibre Channel (FC) нам потрібна спеціальна плата адаптера шини хоста (щось на зразок мережевої карти) на серверах та виділені комутатори FC. Для адресації використовуються довгі всесвітні імена (WWN), що є еквівалентом MAC-адрес, і нічого подібного до IP-адрес або DNS-імен не використовується (тому адреси не можна використовувати з пам'яті). Мережа FC є плоскою, і зазвичай кажуть, що використовується комутація FC, і маршрутизація відсутня.

Фабрика (Fabric) – Fibre Channel Fabric або Switched Fabric – це мережева топологія, що використовується в мережах FC (найпоширеніша з трьох можливих). Фабрика – це єдина замкнута мережа (SAN). Вона складається з одного або кількох комутаторів (FC-комутаторів), до яких підключені кінцеві пристрої (сервери та сховища). Вона дуже схожа на класичну локальну мережу (LAN), кілька пристроїв можуть бути активними одночасно, середовище не

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

використовується спільно. Для високої доступності мережа SAN зазвичай будується з двома окремими фабриками (дві взаємовідключені групи комутаторів, кінцеві пристрої підключені до обох), тобто двома незалежними шляхами.

VSAN – віртуальна мережа сховищ даних (VLAN) – це еквівалент VLAN (віртуальної локальної мережі) від Ethernet (створеної Cisco, яка наразі є стандартом ANSI). За допомогою VSAN ми можемо групувати порти та створювати віртуальну структуру. Один порт може належати кільком VSAN, а порти з різних комутаторів можуть бути призначені одній VSAN. У VSAN ми маємо окремий трафік, а також політику безпеки, зони, членство тощо.

NIC – Мережева плата (NIC) – мережева карта, компонент, що підключає комп'ютер до комп'ютерної мережі.

HBA – адаптер шини хоста – еквівалент мережевої карти, компонент, який підключає комп'ютер до мережі SAN. Термін найчастіше використовується для позначення карти інтерфейсу FC, але також для спеціалізованої карти для iSCSI. HBA реалізує різні поширені операції в апаратному забезпеченні і таким чином розвантажує процесор (наприклад, розвантажує протокол TCP/IP та iSCSI).

WWN – World Wide Name – унікальний ідентифікатор, що використовується у світі Fibre Channel, еквівалент MAC-адреси Ethernet, довжина якого становить 8 байт (64 біти).

WWPN – Всесвітнє ім'я порту – WWN, призначене порту в Fibre Channel Fabric, Cisco іноді використовує термін Port WWN, наприклад 50:40:20:c0:00:f6:93:da

WWNN – Всесвітнє ім'я вузла – WWN, призначене вузлу (кінцевому пристрою, дисковому масиву, комутатору) у Fibre Channel Fabric. Один і той самий WWNN можна побачити на всіх портах пристрою, але щоразу – різний WWPN, наприклад 50:40:20:c0:00:f6:93:db

Зонування (маскування) – зонування VSAN/SAN, ми створюємо зони, і лише пристрої в заданій зоні можуть взаємодіяти один з одним. Це

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

налаштовується на комутаторі. Використання не є обов'язковим, але на практиці завжди використовується з міркувань безпеки.

Багатошляховість – якщо між сервером (ініціатором) та масивом (ціллю) існує кілька шляхів, вони представлені як один та використовуються висока доступність та балансування навантаження.

Налаштування FC на Windows Server

Для FC SAN майже не виконується налаштування на стороні сервера, все відбувається на стороні масиву (створення та розподіл дискового простору) та на стороні мережі (зонування). Нам потрібен FC HBA на сервері та встановлений відповідний драйвер. Сервер автоматично визначає LUN, доступні в мережі. Якщо ми використовуємо кілька шляхів, нам потрібно лише налаштувати Multipath, інакше ми побачимо представлений диск кілька разів.

Таким чином, виходячи з вищеперахованого, дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ-2023

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Мережа зберігання даних (SAN) з'єднує сервери з пристроями зберігання даних за допомогою високошвидкісної мережі. SAN покращують доступ до даних, масштабованість та управління. У цьому посібнику пояснюється, як працюють SAN та які їхні переваги.

Ключові висновки:

– Мережа зберігання даних (SAN) пропонує зберігання даних на рівні блоків, що дозволяє кільком хостам ефективно спільно використовувати ресурси зберігання, підвищуючи продуктивність, масштабованість та централізоване управління.

– Поширені протоколи SAN, такі як Fibre Channel, iSCSI та Fibre Channel over Ethernet (FCoE), сприяють ефективній передачі даних та зв'язку між серверами та пристроями зберігання даних.

– Впровадження SAN має кілька переваг, включаючи покращений захист даних, можливості аварійного відновлення та підвищену продуктивність, хоча це також пов'язано зі складнощами та витратами під час розгортання та обслуговування.

Розуміння мережі зберігання даних (SAN)

Мережа зберігання даних (SAN) – це складна архітектура, розроблена для забезпечення зберігання даних на рівні блоків централізованим та консолідованим способом. На відміну від традиційних сховищ із прямим підключенням (DAS), які підключають сховище безпосередньо до одного комп'ютера, мережі зберігання

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

даних дозволяють кільком хостам безперешкодно використовувати одні й ті ж ресурси зберігання. Така консолідація особливо корисна для програм, що потребують передачі великих обсягів даних та потреб, чутливих до продуктивності, забезпечуючи високу доступність та масштабованість. Крім того, мережеве сховище даних пропонує альтернативне рішення для ефективного управління даними. SAN підключає сервери до спільних пристроїв зберігання даних, що сприяє ефективному управлінню та обробці великих обсягів даних завдяки добре структурованій інфраструктурі зберігання. SAN працюють незалежно від локальної мережі, використовуючи виділені шляхи для передачі даних, що значно підвищує їхню продуктивність порівняно з системами NAS або DAS.

Ключові компоненти архітектури SAN

Архітектура мережі зберігання даних (SAN) складається з кількох ключових компонентів, які працюють разом для створення високоефективного середовища зберігання даних. В основі архітектури SAN лежать хости, підсистеми зберігання даних та мережева структура. Хости – це сервери, яким потрібен доступ до сховища, тоді як підсистеми зберігання даних включають різні пристрої зберігання даних, такі як RAID-системи та масиви зберігання даних. Масив зберігання даних – це спеціалізована підсистема, призначена для зберігання блочних, об'єктних або файлових даних, і вона відіграє вирішальну роль у забезпеченні централізованого управління та покращенні продуктивності корпоративних робочих навантажень, з'єднаних між собою через SAN. Не менш важливими є комутатори SAN, які формують мережеву структуру. Ці мережеві пристрої, включаючи граничні та директорні комутатори, мають вирішальне значення для забезпечення оптимальної продуктивності та надійності. Хостам потрібні відповідні мережеві інтерфейси та програмне забезпечення для керування SAN для ефективного зв'язку, що забезпечує цілісну систему зберігання даних.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

Як працює SAN

Мережі SAN працюють за трирівневою структурою: рівень хоста, рівень структури та рівень зберігання. Рівень хоста складається з серверів, які взаємодіють із мережею SAN, тоді як рівень структури забезпечує зв'язок між цими серверами та пристроями зберігання даних. Рівень зберігання даних – це місце, де зберігаються дані, зазвичай у масивах зберігання даних, і доступ до них здійснюється через логічні пристрої (LUN). Комутатори SAN відіграють ключову роль у підключенні серверів до пристроїв зберігання даних та забезпеченні безперебійного переміщення даних між ними. Такий багаторівневий підхід дозволяє ефективно керувати даними та високошвидкісно передавати дані, що робить мережі SAN ідеальними для середовищ з інтенсивними потребами обробки даних.

Загальні протоколи SAN

Зв'язок у мережі SAN базується на кількох протоколах, які забезпечують безперебійну передачу та керування даними. Поширені протоколи SAN, такі як Fibre Channel, iSCSI та Fibre Channel over Ethernet (FCoE), забезпечують ефективний зв'язок між серверами та пристроями зберігання даних, кожен з яких пропонує унікальні переваги:

– Протокол оптоволоконного каналу (FCP). Fibre Channel в основному використовує протокол Fibre Channel (FCP) для передачі даних, забезпечуючи надійний та надійний метод високошвидкісної передачі даних. Використовуючи волоконно-оптичні кабелі, FCP забезпечує низьку затримку та високу пропускну здатність, що робить його кращим вибором для корпоративних середовищ, де продуктивність та надійність мають першочергове значення.

– Інтерфейс малих комп'ютерних систем Інтернету (iSCSI). iSCSI забезпечує доступ до сховища на рівні блоків через стандартні мережі Ethernet, пропонуючи економічно ефективну альтернативу Fibre Channel. iSCSI використовує існуючу мережеву інфраструктуру для впровадження технології SAN без спеціалізованого обладнання, що підходить для малих та середніх

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

підприємств.

– Оптоволоконний канал через Ethernet (FCoE). Fibre Channel over Ethernet (FCoE) використовує Ethernet для передачі пакетів Fibre Channel, спрощуючи інфраструктуру мереж зберігання даних (SAN), дозволяючи одному пристрою обробляти як IP-пакети, так і дані зберігання. FCoE дозволяє передавати кадри Fibre Channel через мережі Ethernet без втрат, забезпечуючи продуктивність Fibre Channel з гнучкістю Ethernet.

Типи рішень для зберігання даних SAN

Рішення для зберігання даних SAN бувають різних форм, кожна з яких розроблена для задоволення конкретних потреб та середовищ зберігання даних. Рішення SAN варіюються від повністю флеш-мереж SAN для надшвидкого доступу до даних до гібридних SAN, що балансують продуктивність та вартість, та віртуальних SAN для гнучкості у віртуалізованих середовищах:

– All-Flash SAN. Флеш-мережі SAN розроблені для забезпечення максимальної продуктивності з винятковою швидкістю та зменшеною затримкою, що робить їх ідеальними для програм, що потребують інтенсивної обробки даних. Ці системи, хоча й вищі за вартістю, забезпечують неперевершену продуктивність, забезпечуючи безперебійну та ефективну роботу критично важливих програм.

– Гібридна SAN. Гібридні мережі зберігання даних (SAN) поєднують технологію флеш-пам'яті з традиційними обертовими дисками, пропонуючи економічно ефективну альтернативу. Така конфігурація поєднує продуктивність та доступність, що робить гібридні SAN популярними для різноманітних потреб зберігання даних. Рішення гібридного зберігання даних Dell EMC особливо відомі завдяки покращенню використання сховища та одночасному зниженню витрат на управління.

– Віртуальна мережа зберігання даних (vSAN). Віртуальна мережа зберігання даних (vSAN) перетворює фізичні ресурси сховища на віртуалізовані пули, спрощуючи управління та підвищуючи використання ресурсів. Цей

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

програмно-визначений підхід особливо корисний у віртуалізованих центрах обробки даних, дозволяючи створювати масштабовані та гнучкі рішення для зберігання даних.

Переваги впровадження SAN

Впровадження SAN забезпечує такі переваги, як підвищена продуктивність, масштабованість та централізоване управління. SAN покращують захист даних та можливості аварійного відновлення, забезпечуючи високу доступність та безперервність бізнесу:

– Покращена продуктивність і швидкість. Мережі зберігання даних (SAN) розроблені для забезпечення високої продуктивності та низької затримки, що робить їх ідеальними для середовищ з інтенсивним використанням даних. Швидкість передачі даних від 2 Гбіт/с до 32 Гбіт/с дозволяє SAN підтримувати вимогливі програми, такі як редагування відео та аналітика в режимі реального часу. Використовуючи оптоволоконні кабелі, SAN досягають швидкостей, які значно вищі, ніж інші технології зберігання даних, такі як NAS та DAS. Налаштування продуктивності, таке як налаштування параметрів кешу та конфігурацій RAID, може ще більше підвищити швидкість та ефективність SAN. Регулярні огляди та коригування конфігурації забезпечують відповідність SAN потребам високопродуктивних корпоративних програм.

– Масштабованість та гнучкість. Мережі зберігання даних (SAN) пропонують масштабованість, що дозволяє організаціям безперешкодно розширювати ємність сховища. Додавання пристроїв зберігання даних та керування ресурсами зберігання за допомогою віртуалізації забезпечує безперешкодну масштабованість. Ця гнучкість дозволяє мережам зберігання даних задовольняти зростаючі потреби в даних та потреби бізнесу, що розвиваються. Рішення SAN від Dell EMC розроблені для забезпечення високої масштабованості та задоволення різноманітних організаційних вимог.

– Централізоване управління. Мережі SAN спрощують адміністрування ресурсів сховища завдяки централізованому управлінню. Віртуалізовані пули

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

сховищ у мережах SAN забезпечують динамічний розподіл та ефективне керування з єдиного інтерфейсу. Такий централізований підхід призводить до підвищення операційної ефективності та швидшого реагування на зміну потреб у сховищі.

– Захист даних та аварійне відновлення. Технологія SAN включає заходи резервування для забезпечення доступності даних навіть під час збоїв обладнання. SAN-фабрика підвищує надійність, реплікуючи дані на кількох пристроях, усуваючи єдині точки відмови.

Регулярне тестування гарантує, що мережа зберігання даних (SAN) витримуватиме збої, зберігаючи відмовостійкість та резервування. SAN пропонують рішення для аварійного відновлення для швидкого відновлення даних з резервних копій, мінімізуючи час простою та максимізуючи час безвідмовної роботи. Надійні можливості захисту даних розроблені для забезпечення безперервності бізнесу, а резервне копіювання в режимі реального часу мінімізує вплив на нормальну бізнес-діяльність. Добре спроектована мережа SAN гарантує доступність даних. Вона також підтримує продуктивність навіть у разі збоїв компонентів.

Проблеми та міркування щодо розгортання SAN

Розгортання SAN пов'язане з такими труднощами, як технічна складність, витрати та постійне обслуговування. Ретельне врахування цих факторів забезпечує успішне впровадження SAN.

– Складність та технічна експертиза. Для ефективного налаштування та управління SAN необхідні розширені технічні навички в галузі мереж та зберігання даних. Основні навички включають знання протоколів SAN, мережевих комутаторів та масивів зберігання даних. Поточне управління включає регулярні завдання з технічного обслуговування, такі як моніторинг продуктивності, усунення несправностей та оновлення.

Ефективне управління SAN також включає розуміння проблем масштабованості та реагування на зміни вимог до даних.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

обміну файлами завдяки зручним протоколам, тоді як SAN відповідають потребам високопродуктивного блочного сховища. SAN ідеально підходять для високошвидкісних середовищ з низькою затримкою, таких як віртуальні машини та бази даних, тоді як NAS підходить для потокової передачі медіа та спільного обміну файлами.

– Порівняння вартості та масштабованості. Незважаючи на вищі початкові інвестиції, рішення SAN можуть бути більш економічно ефективними для критичних робочих навантажень. Фактори, що впливають на ціноутворення на сховища SAN, включають вимоги до продуктивності, ємності та масштабованості. SAN та NAS можуть співіснувати в одному центрі обробки даних, пропонуючи індивідуальні варіанти масштабованості.

Кроки для налаштування SAN

Налаштування SAN включає критично важливі кроки від планування та проектування до впровадження та постійного управління. Такий структурований підхід забезпечує надійність, масштабованість та ефективність.

– Фаза планування та проектування. Початковий етап зосереджений на рішеннях щодо планування та проектування, що впливають на архітектуру, продуктивність та масштабованість. Для ефективної інтеграції всіх компонентів необхідні спеціалізовані знання в галузі конфігурації мережі та управління сховищами даних. Сумісність апаратного та програмного забезпечення різних постачальників має вирішальне значення для безперебійного налаштування.

– Впровадження та тестування. Процес налаштування SAN починається з встановлення обладнання, після чого проводиться ретельна перевірка продуктивності. Після встановлення перевірка операційних процесів та тестування на нормальний та аварійний режими забезпечують надійну та ефективну роботу SAN.

– Поточне управління та оптимізація. Постійне управління та технічне обслуговування мають вирішальне значення для забезпечення ефективної та безпечної роботи SAN. Регулярне технічне обслуговування, включаючи

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

оновлення, керування виправленнями та перевірки обладнання, допомагає зменшити потенційні проблеми. Постійний моніторинг та усунення несправностей дозволяють виявляти проблеми на ранній стадії, підвищуючи надійність SAN. Надійні заходи безпеки та налаштування продуктивності є важливими для захисту даних та оптимізації SAN.

Найкращі практики управління SAN

Найкращі практики управління SAN забезпечують оптимальну продуктивність та довговічність. Ці практики включають:

- Регулярні огляди ефективності.
- Ефективна конфігурація.
- Моніторинг.
- Усунення несправностей.
- Планування потужностей.
- Регулярний моніторинг та усунення несправностей.

Розглянемо їх більш детально:

– Регулярні оновлення та моніторинг мають вирішальне значення для запобігання збоєм та забезпечення оптимальної продуктивності. Проактивний моніторинг за допомогою таких інструментів, як Zabbix та PRTG, допомагає виявляти проблеми до того, як вони стануть серйозними. Автоматизовані сповіщення повідомляють адміністраторів про потенційні проблеми SAN, перш ніж вони вплинуть на користувачів. Ефективний моніторинг запобігає зниженню продуктивності, виявляючи вузькі місця на ранній стадії.

– Заходи безпеки. Заходи безпеки є критично важливими в середовищах SAN для захисту конфіденційних даних та забезпечення відповідності вимогам. Такі стратегії, як зонування та маскуваня LUN, обмежують доступ до певних LUN, підвищуючи безпеку. Шифрування даних під час передачі та регулярні аудити безпеки допомагають підтримувати надійний захист даних.

– Налаштування продуктивності. Налаштування продуктивності має вирішальне значення для максимізації швидкості та ефективності передачі даних

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

у мережах зберігання даних (SAN). Досягнення у віртуалізації сховищ даних забезпечують кращий об'єднання ресурсів, що дозволяє керувати кількома системами зберігання даних як єдиним цілим. Ці вдосконалення сприяють ефективній обробці даних та розподілу ресурсів у мережах зберігання даних (SAN).

Майбутні тенденції в технології SAN

Новітні тенденції в технології SAN обіцяють покращену продуктивність, гнучкість та інтеграцію з іншими технологіями. Такі тенденції, як NVMe поверх Fabrics, хмарна інтеграція та досягнення у віртуалізації сховищ, мають на меті революціонізувати середовища SAN.

– Перевага NVMe замість Fabrics. NVMe over Fabrics покращує продуктивність SAN, зменшуючи затримку та збільшуючи пропускну здатність порівняно з традиційними протоколами. Ця технологія дозволяє передавати команди NVMe через різні мережеві протоколи, значно підвищуючи ефективність передачі даних. NVMe over Fabrics забезпечує вищу швидкість передачі даних, виконуючи команди NVMe через різні мережеві тканини.

– Інтеграція з хмарним сховищем. Інтеграція мереж SAN з хмарним сховищем дозволяє організаціям створювати гібридні ІТ-середовища, підвищуючи гнучкість та економічну ефективність. Ця інтеграція поєднує локальні та хмарні ресурси, підвищуючи масштабованість. Такі компанії, як HPE, пропонують рішення SAN, які безперешкодно інтегруються з хмарними середовищами.

– Досягнення у віртуалізації сховищ. Віртуалізація сховища даних абстрагує логічне сховище від фізичного, забезпечуючи єдиний інтерфейс керування для різних пристроїв зберігання даних. Цей підхід підвищує ефективність, групує кілька дискових масивів від різних постачальників в один пристрій зберігання даних. Ці вдосконалення покращують управління різноманітними ресурсами зберігання та їх використання в мережах зберігання даних (SAN).

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

Основні постачальники та продукти SAN

Основні постачальники домінують на ринку SAN, кожен з яких пропонує унікальні продукти, що задовольняють різні потреби підприємств. Розуміння цих пропозицій допомагає організаціям вибрати правильне рішення SAN.

– Dell EMC. Dell EMC, значний гравець на ринку SAN, пропонує інноваційні рішення для зберігання даних, такі як PowerMax та PowerStore, для ефективного управління робочим навантаженням. Оснащені розширеними функціями, ці рішення підвищують продуктивність та надійність, що підходить для високопродуктивних середовищ.

– NetApp. Рішення SAN від NetApp гарантують доступність даних на рівні 99,9999%, забезпечуючи високу надійність для критично важливих програм. NetApp використовує різні протоколи для високої продуктивності та робить акцент на сталому розвитку, пропонуючи рішення, що забезпечують економію енергії до 97%.

– HPE. Hewlett Packard Enterprise (HPE) пропонує комплексні продукти SAN, оптимізовані для високої продуктивності та простоти використання в корпоративних середовищах. Сховище даних HPE MSA 1050 SAN розроблено для клієнтів, які цінують вартість, пропонуючи поєднання продуктивності та доступності з можливістю підключення до флеш-пам'яті. Переваги продуктів SAN HPE включають покращену масштабованість та оптимальну продуктивність, адаптовану для різних робочих навантажень.

Висновки

Мережі зберігання даних (SAN) забезпечують надійне, масштабоване та високопродуктивне рішення для сучасних потреб зберігання даних. Розуміючи архітектуру, протоколи та типи рішень для зберігання даних SAN, організації можуть приймати обґрунтовані рішення, що відповідають їхнім бізнес-цілям. Переваги SAN, від підвищеної продуктивності до централізованого управління та захисту даних, роблять їх цінним активом для будь-якого підприємства. Однак розгортання SAN пов'язане з певними труднощами, включаючи складність,

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

вартість та необхідність постійного обслуговування. Дотримуючись передових практик в управлінні SAN та будучи в курсі майбутніх тенденцій, організації можуть оптимізувати свої середовища SAN для максимальної ефективності та надійності. З розвитком технологій SAN залишатимуться наріжним каменем корпоративного зберігання даних, адаптуючись до вимог постійно мінливого цифрового ландшафту.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – це об’єктно-орієнтована мова програмування високого рівня загального призначення з відкритим кодом. Це визначення може бути важким для новачків, тому розглянемо кожну характеристику окремо, щоб зрозуміти, що вона означає:

- Відкритий вихідний код: це безкоштовно та доступно для подальших покращень, таких як додавання корисних функцій або виправлення помилок.
- Об’єктно-орієнтована: заснована не на функціях, але в об’єктах з певними атрибутами й методами.
- Високий рівень: зручний для людини, а не для комп’ютера.
- Загальне призначення: можна використовувати для створення будь-яких програм.

Ця мова використовується в будь-якому програмному забезпеченні, про яке ви тільки можете подумати. Ви можете використовувати його для створення веб-сайтів, штучного інтелекту, серверів, програмного забезпечення для бізнесу та багато іншого. Також застосовується в науці про дані, аналізі даних, машинному навчанні, інженерії даних, веб-розробці, розробці програмного забезпечення та інших галузях.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

Переваги та недоліки Python

Переваги:

– Її легко читати, вчити та писати. Це мова програмування високого рівня з англійським синтаксисом. Це полегшує читання та розуміння коду. Її дійсно легко зрозуміти і вивчити, тому багато людей рекомендують Python новачкам. Вам потрібно менше рядків коду для виконання того ж завдання в порівнянні з іншими основними мовами, такими як C/C++ та Java.

– Підвищує продуктивність. Це дуже продуктивна мова. Завдяки її простоті розробники можуть зосередитися на розв'язанні проблеми. Їм не потрібно витрачати багато часу на розуміння синтаксису або поведінку мови програмування. Ви пишете менше коду та виконуєте більше завдань.

– Інтерпретована мова. Python мова, що інтерпретується, а це означає, що вона безпосередньо виконує код по рядку. Якщо сталася помилка, вона зупиняє подальше виконання та повідомляє про її виникнення. Вона показує лише одну помилку, навіть якщо у програмі їх кілька. Це спрощує налагодження.

– Динамічно типізована. Python не визначає тип змінної, доки ми не запустимо код. Вона автоматично надає тип даних, коли відбувається процес виконання. Фахівець може не турбуватися про оголошення змінних та типи даних.

– Безкоштовна та з відкритим вихідним кодом. Ця мова постачається під схваленою OSI ліцензією з відкритим вихідним кодом. Це робить його безкоштовним для використання та розповсюдження. Ви можете завантажити вихідний код, змінити його та навіть розповсюджувати свою версію. Це корисно для організацій, які хочуть використати свою версію для розробки.

– Підтримка великих бібліотек. Стандартна бібліотека Python є величезною, ви можете знайти майже всі функції, необхідні для вашого завдання. Таким чином ви не залежите від зовнішніх бібліотек.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

розроблених спеціально для аналітичних цілей. Найвідоміші бібліотеки Python для аналізу даних – це pandas і NumPy. Ці інструменти дозволяють робити з вашими даними майже все, наприклад, очищати і аналізувати їх, вивчати статистику або візуалізувати приховані тенденції у ваших даних.

– Для візуалізації даних. Візуалізація даних – це окрема частина аналізу даних, яка допомагає нам подавати інформацію, необроблену чи очищену, у більш змістовній формі. Тут Python знову входить у гру, пропонуючи широкий спектр інструментів візуалізації даних. Найпопулярніші з них – matplotlib і заснований на ній seaborn. Використовуючи їх, ми можемо створювати буквально всі види візуалізації: від найпростіших до складніших.

– Для машинного навчання. Машинне навчання (ML) є основою більшості завдань науки даних. Він є областю штучного інтелекту, пов'язаною з використанням алгоритмів, що дозволяють машинам вивчати закономірності та тенденції на основі історичних даних, щоб робити прогнози на основі невідомих даних. – Використовуючи методи ML, ми можемо створювати моделі, які можуть точно передбачити швидкість відтоку клієнтів компанії, оцінити ризик виникнення у людини певного захворювання, визначити оптимальне розташування автомобілів таксі й т.д. За допомогою Python ми можемо побудувати модель ML, використовуючи лише три рядки коду.

– Для розробки програмного забезпечення. Крім свого багатостороннього застосування в галузях науки про дані, Python використовується на кожному етапі розробки програмного забезпечення, включаючи контроль складання, автоматичну безперервну компіляцію, прототипування, відстеження помилок, тестування та обслуговування програмного забезпечення. За допомогою цієї мови можемо створювати аудіо- або відеопрограми на основі методів штучного інтелекту, машинного навчання, API (інтерфейсів прикладного програмування), GUI (графічних інтерфейсів) або будь-якого іншого типу програмного забезпечення.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

– Для веброзробки. У той час як для створення візуальної частини вебсайту ми переважно будемо використовувати такі мови, як HTML, CSS та JavaScript, для його невидимої частини ми часто вибираємо Python. Серед масштабних вебсайтів та програм, створених за допомогою цієї мови, варто згадати Google, Facebook, Instagram, YouTube, Dropbox та Reddit.

– Для автоматизації задач/скриптингу. Це відмінний інструмент для написання програм для автоматизації різних завдань, що повторюються. Цей процес називається скриптингом. Зокрема, можна робити скрипти для роботи з файлами та папками. Наприклад, можна створювати, перейменовувати, перетворювати, розділяти, об'єднувати або видаляти файли, перевіряти їх наявність помилок. Ви також можете використовувати автоматизацію Python для пошуку та завантаження інформації з Інтернету, заповнення та надсилання онлайн-форм та надсилання регулярних повідомлень або електронних листів.

Яким фахівцям потрібно володіти Python:

- Фахівець з даних.
- Аналітик даних.
- Інженер даних.
- Інженер з машинного навчання.
- Журналіст даних.
- Архітектор даних.
- Повний стек веб-розробника.
- Backend-розробник.
- DevOps-інженер.
- Інженер-програміст.

Можемо зробити висновок, що Python ще довго буде популярною мовою, хоч і має низку недоліків. Цю мову використовують для створення вебсайтів, штучного інтелекту, серверів, програмного забезпечення для бізнесу, аналізу даних, машинного навчання, інженерії даних та для багатьох інших областей. Це перспективна і затребувана навичка, яка необхідна у всіх галузях.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи зберігання даних SAN на основі стандарту FC Gen.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Протягом багатьох років стан середовищ зберігання даних в ІТ різко змінився. Короткий огляд змін проведе вас від ранніх стрічкових систем через еволюцію жорстких дисків (HDD) до розробки RAID-систем та корпоративних масивів. Однією з традиційно вірних речей було те, що сховища, засновані на структурних блоках HDD, розвивалися повільно. Перехід від дисків зі швидкістю 5400 об/хв до дисків зі швидкістю 7200 об/хв як елемента продуктивності (більше даних під головкою) за секунду знадобився десять років, щоб повністю заповнити центри обробки даних. Інші розробки включали щільність магнітної сигнатури на пластині диска та кількість пластин і головок на диск. Ця ретроспектива корисна лише тим, що вказує на те, що середовище зберігання даних в ІТ не розвивалося так швидко, як, скажімо, розробка процесорів або продуктивності та ємності пам'яті.

Ці розробки були в кремнії, а розробка накопичувачів була механічною. Як наслідок, закон Мура застосовувався до процесора та пам'яті, але не до сховища. Характер сховища почав змінюватися з появою твердотільних накопичувачів (SSD).

Спочатку прогрес був помірним. Блискучим кроком, що сприяв виходу на ринок, постачальники накопичувачів зробили платформу SSD такої ж форми/корпуса, як і існуючі жорсткі диски (HDD), з тими ж роз'ємами SCSI, SAS та SATA, що означало сумісність роз'ємів на серверній частині масиву для нової технології. Однак на ранніх етапах контролери корпоративних масивів та, у випадку вбудованих дискових накопичувачів у серверах, стек драйверів ОС не скористалися перевагами зміни продуктивності та інших характеристик дисків. Те, що стек ОС не використовував переваги нової технології, було однією з

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

сховищ обговорюватимуть коефіцієнти розподілу даних для платформ зберігання, які являють собою кількість серверів та програм у мережі, що використовують певний масив або порт масиву для свого доступу до сховища.

Залежно від типів програм та їхніх потреб у продуктивності, це співвідношення може коливатися від невеликих одиниць до 40-50 серверів (сотень віртуальних машин). Як і в будь-якому сценарії забезпечення ресурсами, адміністратор сховища має справу з прогнозами того, скільки потужності та продуктивності використовуватиме будь-який сервер або програма. Але продуктивність програм є змінною; час доби, тиждень, місяць, а також сезонні або подіями викликають піки або падіння попиту на програми.

Ще один момент полягає в тому, що вся база програм не оновлюється одночасно. Типовий сценарій полягає в тому, що в середовищі одночасно існують кілька поколінь продуктивності. Наявність вікон обслуговування спонукає це до перепланування існуючих середовищ на нові сервери та сховища. Деякі застарілі програми можуть не мати середовища, яке працює під керуванням поточної операційної системи або версії програми. Одна може мати десятирічні або старіші операційні системи, підключені до НВА з двома або більше старішими поколіннями технологій. Як же тоді збалансувати цю все ще критично важливу програму з потребами та продуктивністю новіших машин?

Відповідь полягає в поєднанні топології, збалансованого виділення ресурсів, детального моніторингу та автоматизованого пом'якшення наслідків.

Найбільш гнучкою конфігурацією є топологія «ядро-край». Така архітектура дозволяє значне масштабування, зберігаючи при цьому низьку кількість переходів, тобто кількість передач даних з однієї платформи на іншу. Локальність підключення сховища для високопродуктивних програм – це міркування, яке спонукає деяких адміністраторів сховищ розміщувати порти сховища на тому ж блейд-сервері, комутаторі або групі портів, що й сервер програм. Однак така конструкція впливає на гнучкість ІТ-відділу для переміщення програми з однієї серверної платформи на іншу (не є тривіальним

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

міркуванням у високо віртуалізованому середовищі, де платформи гіпервізора можуть часто переносити програми між серверними платформами).

Ще однією альтернативною топологією є повністю сітчаста конструкція; кожен комутатор має прямий ISL до кожного іншого комутатора в структурі. Повна сітчаста конструкція є проблематичною для середовищ з багатьма комутаторами, оскільки вона використовує цінні порти для ISL, які в іншому випадку використовувалися б для підключення серверів та сховища. Платформи Director мають міжшасі (ICL), що забезпечує виняткову пропускну здатність та масштабування між шасі без використання портів. Повна сітчаста конструкція гарантує, що жоден кінцевий пристрій не знаходиться на відстані більше одного переходу від будь-якого іншого кінцевого пристрою.

З точки зору продуктивності, важливо зазначити, що поява повністю флеш-центрів обробки даних також означає, що нові технології зберігання даних, засновані на продуктивності, ємності або обидва варіанти, з'являються з інтервалом від 18 до 24 місяців. Оновлення технологій не означає повну заміну існуючих платформ, а радше те, що ваша мережа зберігання даних повинна бути здатною вмістити приблизно дві такі ітерації за 4-5-річний цикл амортизації капіталу. Однією з переваг Fibre Channel SAN є подвійне резервування апаратного забезпечення та ізольований характер архітектур SAN. Архітектури тканин А та В передбачають, що жоден дефект пристрою, планове технічне обслуговування, випадкова подія людини чи зловмисна діяльність повністю не виведуть підключення до сховища з офлайн-режиму. Підтримка мережі сховища в режимі онлайн дозволяє безперебійне оновлення технологій, що стосується технологічних оновлень елементів мережі зберігання даних та підключеного сервера та сховища.

Однією з додаткових змін, необхідних для повністю флеш-центру обробки даних, є покращений моніторинг, частково завдяки зменшенню затримки, критичності даних та великої кількості даних, що передаються в сучасних SAN.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

3.2 Розробка структурної схеми

Процес планування SAN подібний до будь-якого процесу планування проекту та включає такі фази:

- Фаза I: Збір вимог.
- Фаза II: Розробка технічних специфікацій.
- Фаза III: Оцінка витрат на проект.
- Фаза IV: Аналіз рентабельності інвестицій (ROI) або загальної вартості володіння (TCO) (за необхідності).
- Фаза V: Створення детального плану проектування та впровадження SAN.

Вибираючи критерії, які слід відповідати, слід залучити користувачів, експертів з серверів та сховищ даних (SME) та інших відповідних експертів, щоб зрозуміти роль структури. Оскільки більшість SAN працюють протягом тривалого часу, перш ніж їх оновлювати, враховуйте майбутнє зростання, оскільки SAN складні для реархітектури. Розгортання нових SAN або розширення існуючих для задоволення додаткових робочих навантажень у структурах вимагає критичної оцінки бізнес- та технологічних вимог. Правильна увага до планування забезпечить, щоб SAN після розгортання відповідала всім поточним та майбутнім бізнес-цілям, включаючи доступність, простоту розгортання, продуктивність, майбутнє зростання бізнесу та вартість.

Критичним аспектом успішного впровадження, який часто ігнорується, є постійне управління інфраструктурою. Визначення SME системного рівня для всіх компонентів, що складають SAN, та належне й актуальне навчання з цих компонентів має вирішальне значення для ефективного проектування та операційного управління інфраструктурою.

Під час проектування нової SAN або розширення існуючої SAN слід враховувати такі параметри:

1. Віртуалізація додатків:

- Які додатки працюватимуть у середовищі віртуальних машин (VM)?
- Скільки VM працюватиме на фізичному сервері?
- За яких умов VM будуть мігрувати (робочий та неробочий час; чи потрібен додатковий процесор або пам'ять для підтримки часу відгуку)?
- Чи є потреба в твердотільних носіях для покращення часу відгуку на читання?

2. Однорідні/неоднорідні серверні та сховищні платформи:

- Чи використовуються блейд-сервери чи стійкові сервери?
- Чи впроваджено автоматичне розподілення на рівні?
- Які версії Brocade Fabric OS® (FOS) підтримуються в середовищі зберігання даних від різних постачальників?
- Який запланований цикл оновлення серверів та платформ зберігання даних (2 роки або 3 роки)?

3. Масштабованість:

- Скільки портів користувачів потрібно зараз?
- Скільки пристроїв підключатиметься через шлюз доступу?
- Скільки ISL та Brocade UltraScale ICL потрібно для мінімізації перевантаження в мережі?
- Які відстані для з'єднань ISL та ICL потрібно підтримувати?
- Чи масштабується мережа на межі ядра?

4. Резервне копіювання та стійкість до аварій:

- Чи є централізоване резервне копіювання? (Це визначає кількість ISL, необхідних для мінімізації перевантаження при пікових навантаженнях.)
- Який вплив резервного копіювання на чутливі до затримки програми?
- Чи базується рішення для ліквідації аварій на міжміських ISL метрополітену FC чи на рішенні FC over Internet Protocol (FCIP)?

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

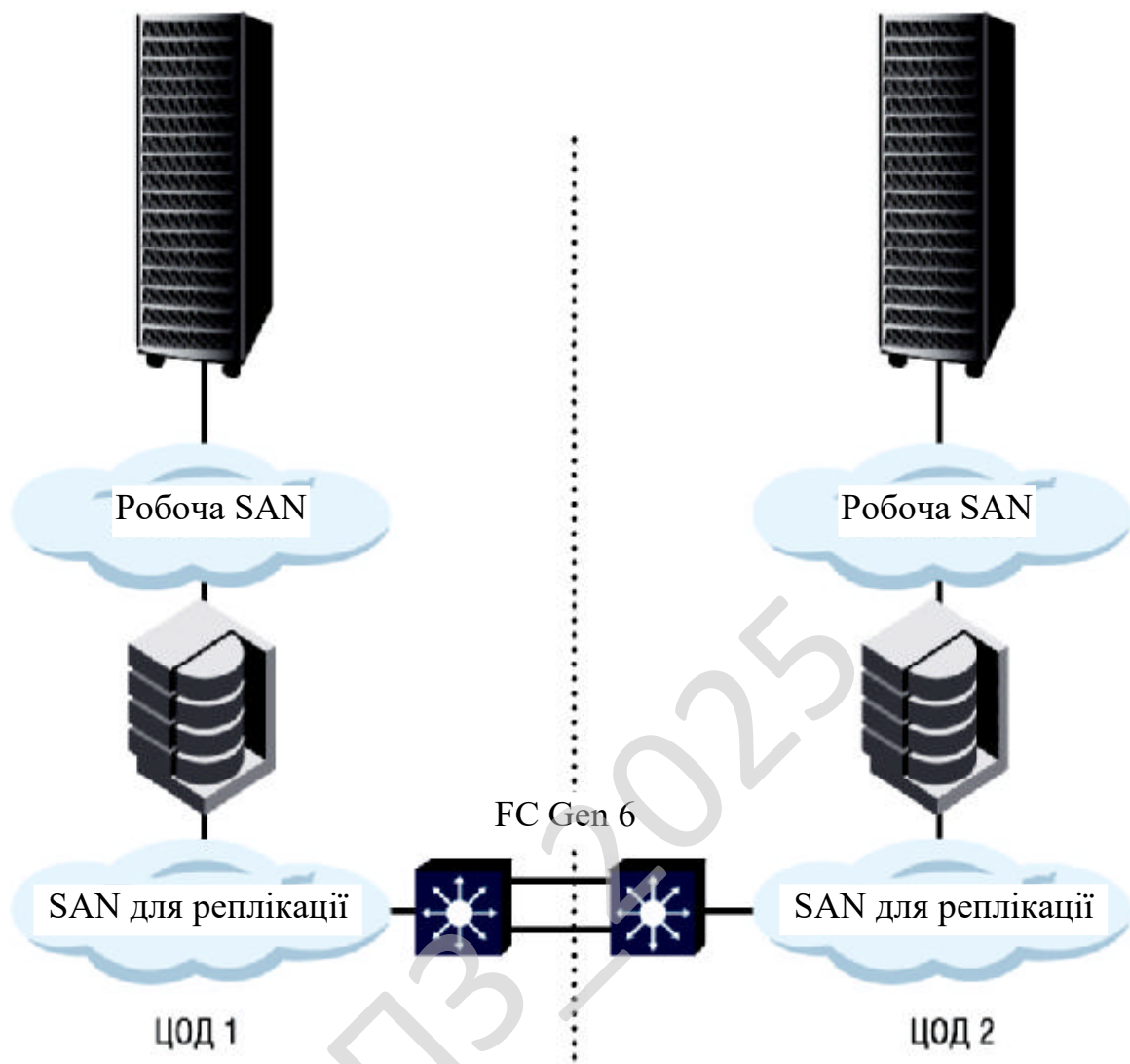


Рисунок 3.1 – Структурна схема системи

5. Діагностика та керуваність:

– Який основний інтерфейс керування для SAN (CLI, Brocade SANnav або інструмент стороннього виробника)?

– Як часто оновлюватимуться Brocade FOS та SANnav?

– Як перевіряється цілісність кабелів та оптичних елементів?

6. Захист інвестицій:

– Чи потрібна підтримка для додавання комутаторів Gen 7 до структури Gen 6?

– Чи потрібна підтримка для технологій зберігання даних, таких як NVMe поверх структур?

– Яка підтримка сумісності пристроїв потрібна?

– Чи потрібна сумісність для інших технологій, таких як UCS?

Типова архітектура SAN складається з периферійних пристроїв, мережевих пристроїв та кабелів. Топологія зазвичай описується з точки зору взаємопов'язаних комутаторів, таких як згорнуте ядро, ядро-край та повна мережа. Рекомендована топологія SAN для оптимізації продуктивності, доступності, управління та масштабованості – це багаторівнева топологія "ядро-край". Підхід "ядро-край" забезпечує хорошу продуктивність без зайвих взаємозв'язків. На високому рівні багаторівнева топологія має велику кількість периферійних комутаторів, що використовуються для підключення пристроїв, та меншу кількість основних комутаторів, що використовуються для маршрутизації трафіку між периферійними комутаторами, як показано на рисунку 3.1.

Fibre Channel over Ethernet (FCoE) забезпечує передачу трафіку Fibre Channel по локальній мережі Ethernet. Інкапсуляція фреймів FC у кадри Ethernet дозволяє сполучати блоковий трафік систем зберігання з іншим трафіком LAN. У результаті спрощується мережна інфраструктура ЦОД і знижуються витрати, оскільки не потрібно обслуговувати кілька мереж. Однак, як вважають деякі фахівці, незалежно від того, яка технологія використовується, фабрика повинна працювати у виділеній мережі зберігання й ні про яку конвергенцію з мережею передачі даних не може бути мови – ризики переважають вигравш у вартості рішення.

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи зображена на рисунку 3.2. Представимо опис розробленого програмного комплексу для дослідження мереж зберігання даних SAN на основі стандарту FC Gen 6 із неоднорідним потоком повідомлень на основі аналітичного й імітаційного моделювання.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

Завдання проектування мережі складається у визначенні пропускних здатностей КЗ із урахуванням неоднорідності трафіку, різноманіття топологій, алгоритмів маршрутизації, варіантів розміщення прикладних програм і наборів даних по вузлах мережі, способів взаємодії користувачів мережі.

Аналітична модель мереж зберігання даних SAN на основі стандарту FC Gen 6 будується у вигляді розімкнутої СеМО для будь-якої топології мережі, що задається аналітично або графічно. На основі аналітичної моделі вирішується завдання визначення пропускних здатностей КЗ у розподілених мереж зберігання даних SAN на основі стандарту FC Gen 6 при обмеженнях на час доставки пакетів або на вартість мережі.

Програма дозволяє розрахувати характеристики мереж зберігання даних SAN на основі стандарту FC Gen 6, такі як:

- пропускні здатності КЗ;
- час затримки пакетів при передачі по кожному каналу й у мережі в цілому;
- завантаження кожного КЗ;
- інтенсивності потоків пакетів у каналах зв'язку;
- імовірності передачі пакетів від користувачів у мережу, між каналами й з мережі до користувачів.

Програма дозволяє варіювати отримані характеристики, такі як:

- пропускні здатності;
- час передачі по КЗ;
- завантаження КЗ;
- час доставки пакетів у мережі зберігання даних SAN на основі стандарту FC Gen 6;
- вартість мереж зберігання даних SAN на основі стандарту FC Gen 6 для одержання необхідних результатів.

Крім того, вона дозволяє вибрати значення пропускних здатностей з дискретного ряду значень, знайдених із традиційної оптимізації безперервних значень пропускних здатностей.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

Аналітична модель мереж зберігання даних SAN на основі стандарту FC Gen 6

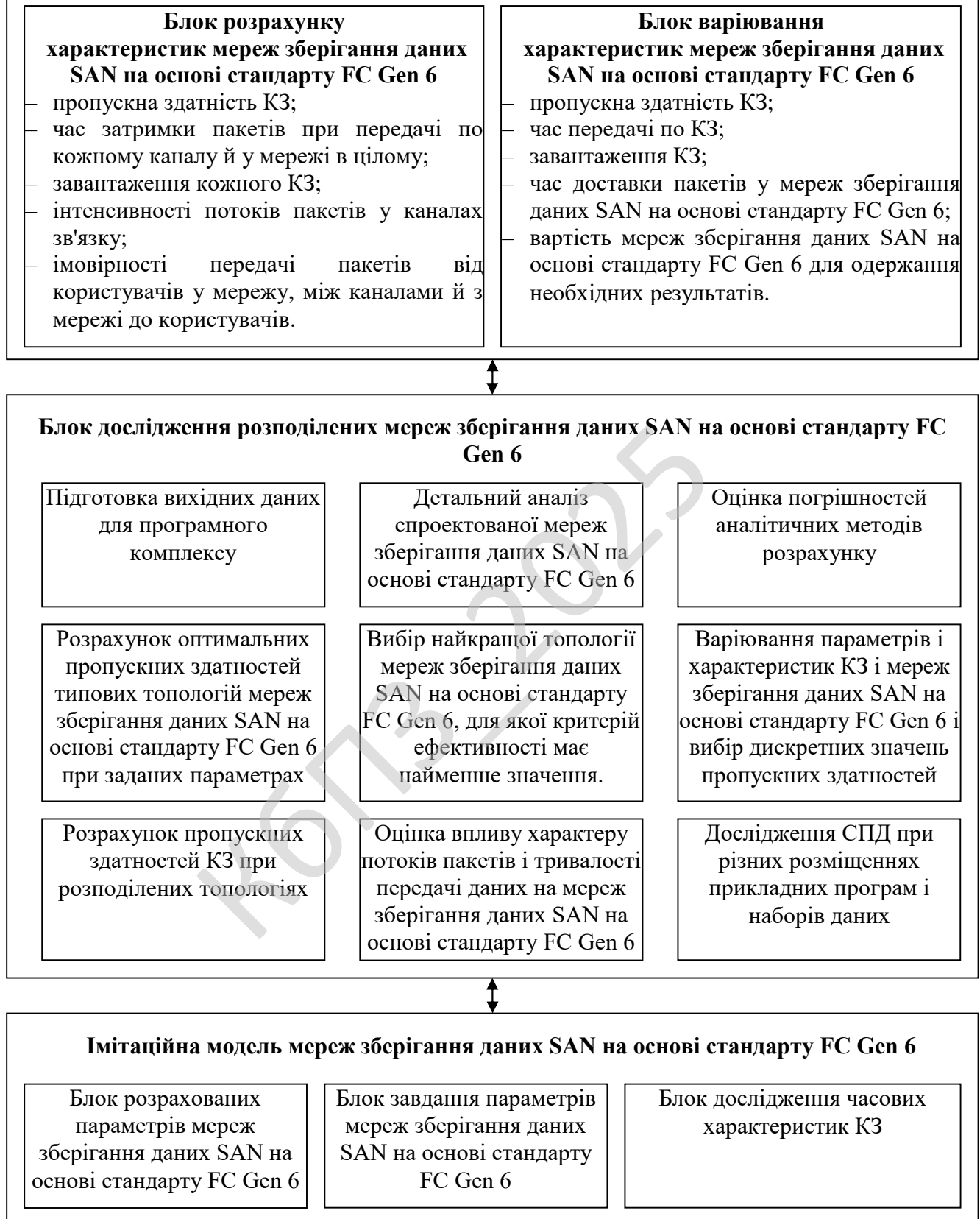


Рисунок 3.2 – Функціональна схема системи

Імітаційна модель мереж зберігання даних SAN на основі стандарту FC Gen 6 генерується автоматично на основі розрахунку характеристик аналітичної моделі у вигляді розімкнутої експонентної СеМО. Отримані характеристики аналітичної моделі використовуються як параметри імітаційної моделі мереж зберігання даних SAN на основі стандарту FC Gen 6, що представляє собою розімкнуту СеМО, вузли якої відповідають каналам зв'язку, а заявки – пакетам у мереж зберігання даних SAN на основі стандарту FC Gen 6. Число джерел заявок в імітаційній моделі дорівнює числу вузлів у мереж зберігання даних SAN на основі стандарту FC Gen 6, при цьому інтенсивності надходження заявок у моделі визначаються як зовнішні інтенсивності пакетів від користувачів до вузлів мереж зберігання даних SAN на основі стандарту FC Gen 6 відповідно. Аналітична модель, реалізована в програмі, дозволяє розрахувати наступні параметри для імітаційної моделі.

1. Імовірність передачі пакетів від користувача j до каналу k .
2. Імовірність передачі пакетів від каналу k до каналу h .
3. Імовірність передачі пакетів від каналу k до користувача j .
4. Середній інтервал часу між вступниками пакетами від користувача j у мереж зберігання даних SAN на основі стандарту FC Gen 6.
5. Середній час передач пакетів у каналах зв'язку k .

Крім **розрахованих параметрів** для імітаційної моделі мереж зберігання даних SAN на основі стандарту FC Gen 6 додатково необхідно **задати наступні параметри:**

а) закони розподілів інтервалів часу між вступниками пакетами від користувачів у мереж зберігання даних SAN на основі стандарту FC Gen 6 із коефіцієнтами варіації цих розподілів;

б) закони розподілів часу передачі пакетів у КЗ із коефіцієнтами варіації цих розподілів.

Відзначимо, що імітаційна модель мереж зберігання даних SAN на основі стандарту FC Gen 6 призначена для детального аналізу характеристик

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

функціонування мережі, спроектованої в процесі аналітичного моделювання. При цьому, у випадку відмінності реального характеру процесів надходження пакетів у мережу або передачі пакетів по каналах зв'язку від експонентного, в імітаційній моделі передбачена можливість варіювання законів розподілу часу передачі (довжин обслуговування) пакетів у кожному із КЗ, а також законів розподілу інтервалів часу між вступними в мережу пакетами. Як такі закони в роботі використовувалися наступні розподіли: експонентний, детермінований; гіпоекспоненційний різного порядку й, відповідно, з різними коефіцієнтами варіації; рівномірний; експонентний з ненульовими зсувами; гіперекспонентний; Гамма-розподіл.

Аналогічно розроблений засіб дослідження часових характеристик каналу зв'язку шляхом генерування імітаційної моделі каналу зв'язку у вигляді СМО типу G/G/1 з можливістю зміни завантаження каналу й варіювання законів розподілу інтервалів часу між пакетами й часу передачі пакетів по каналі. Даний засіб дозволяє одержати значення часових характеристик каналу зв'язку й оцінити погрішність аналітичних методів розрахунку характеристик каналу зв'язку.

Сформульовано методикку дослідження розподілених мереж зберігання даних SAN на основі стандарту FC Gen 6 із неоднорідним трафіком, що містить наступні етапи.

1. Підготовка вихідних даних для програмного комплексу:

- кількість вузлів мережі і їхнє взаємне розташування;
- навантаження мереж зберігання даних SAN на основі стандарту FC Gen 6, створювана користувачами при роботі із прикладними програмами, наборами даних і в процесі обміну повідомлень;
- обмеження на середній час доставки пакетів або на вартість мереж зберігання даних SAN на основі стандарту FC Gen 6;
- максимальна довжина пакета;
- вибір типу каналів і завдання вартісних коефіцієнтів КЗ.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

2. Розрахунок оптимальних пропускних здатностей типових топологій мереж зберігання даних SAN на основі стандарту FC Gen 6 при заданих наступних параметрах:

- типова топологія: зірка, кільце, дерево, повнозв'язна;
- модель взаємодії користувачів мережі: RDA (Remote Data Access), DBS (DataBase Server) і AS (Application Server);
- розподіл прикладних програм і наборів даних по вузлах мереж зберігання даних SAN на основі стандарту FC Gen 6;
- метод маршрутизації.

Після завдання цих параметрів розраховуються оптимальні значення пропускних здатностей КЗ, час передачі пакетів і завантаження каналів.

3. Розрахунок пропускних здатностей КЗ при розподілених топологіях.

4. Вибір найкращої топології мереж зберігання даних SAN на основі стандарту FC Gen 6, для якої обрана залежно від постановки завдання як критерій ефективності характеристика мереж зберігання даних SAN на основі стандарту FC Gen 6 (середній час доставки пакетів або вартість мереж зберігання даних SAN на основі стандарту FC Gen 6) приймає найменше значення.

5. Варіювання параметрів і характеристик КЗ і мереж зберігання даних SAN на основі стандарту FC Gen 6 і вибір дискретних значень пропускних здатностей.

6. Оцінка погрешностей аналітичних методів розрахунку характеристик КЗ.

7. Оцінка впливу характеру потоків пакетів і тривалості передачі даних на характеристики мереж зберігання даних SAN на основі стандарту FC Gen 6:

- вплив третього моменту розподілу вхідного потоку пакетів на характеристики КЗ;
- вплив довжини пакетів на характеристики каналів зв'язку й мереж зберігання даних SAN на основі стандарту FC Gen 6;

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

– вплив законів розподілів трафіку в мережах і часі передачі пакетів у КЗ на характеристики функціонування мереж зберігання даних SAN на основі стандарту FC Gen 6.

8. Дослідження мереж зберігання даних SAN на основі стандарту FC Gen 6 при різних розміщеннях прикладних програм і наборів даних.

9. Детальний аналіз спроектованої мереж зберігання даних SAN на основі стандарту FC Gen 6, що припускає варіювання параметрів мереж зберігання даних SAN на основі стандарту FC Gen 6, у тому числі: довжини запитів і відповідей прикладних програм і наборів даних, імовірність передачі по основному шляху, способи маршрутизації й т.д.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма взаємодії процесів системи, розробленої у результаті виконання магістерської роботи, наведена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі. Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування).

Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Первинною стадією без якої не відбувається розробка програмного забезпечення це звичайно розробка блок-схем. На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З якої видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об'єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю мереж зберігання даних SAN на основі стандарту FC Gen 6.

При складанні блок-схем програмного забезпечення і напрацювання алгоритмів я зіткнувся з масою проблем, які вимагали напрацювання процедур і функцій над основною проблематикою.

Для чого були створені додаткові класи, типи даних і константи, що забезпечило вирішення проблем.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

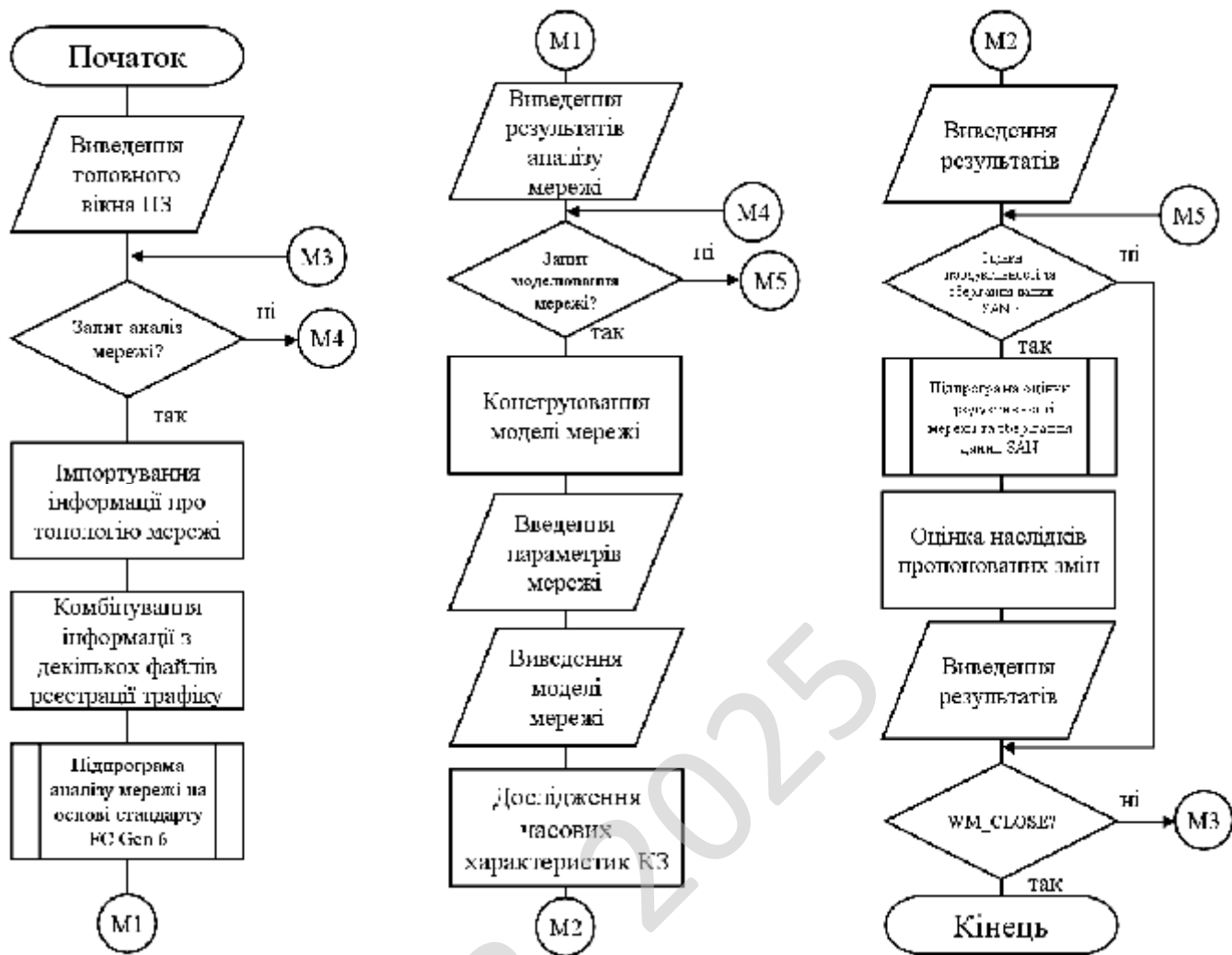


Рисунок 4.1 – Блок-схема основної програми

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції.

Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються.

Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

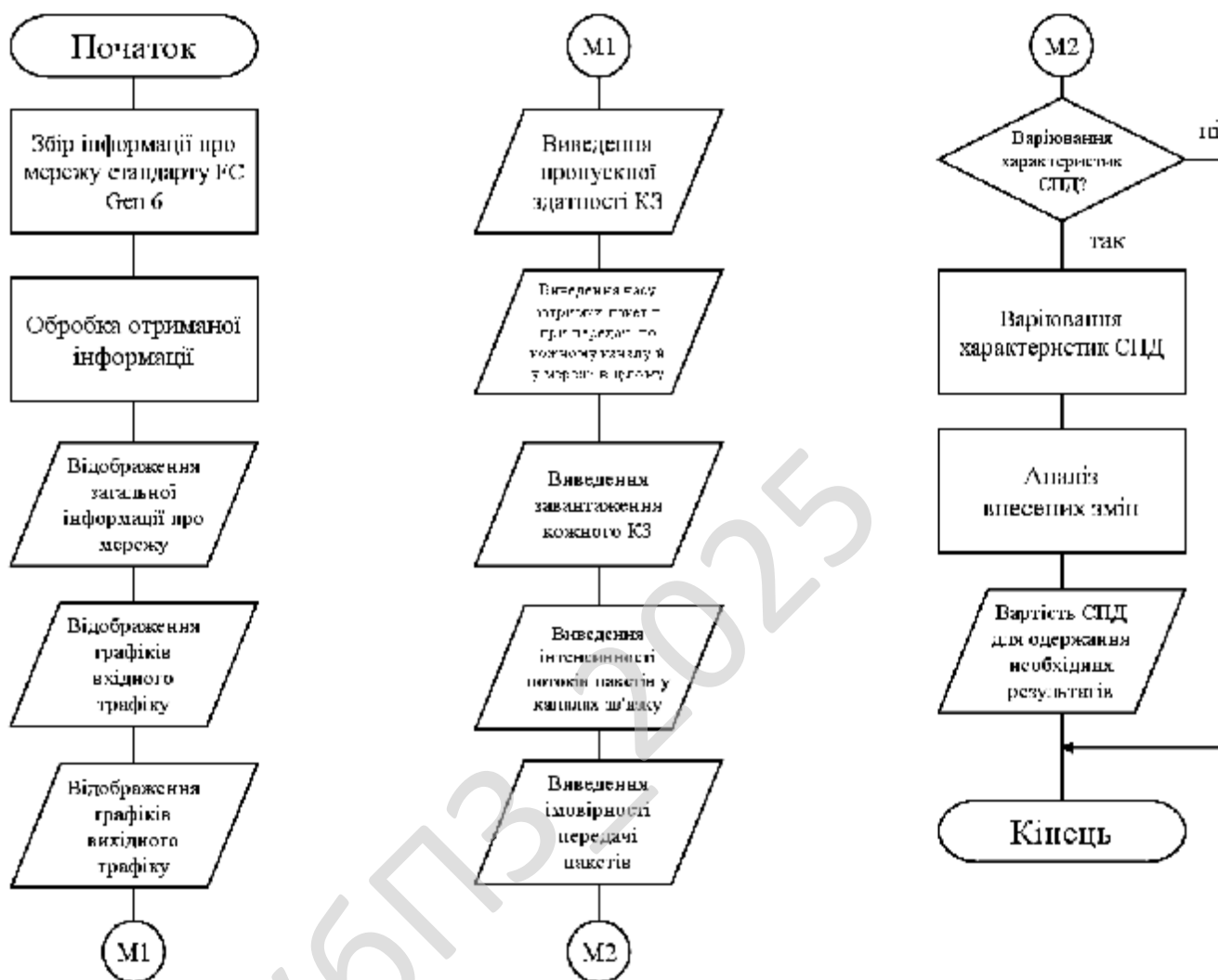


Рисунок 4.2 – Блок-схема роботи підпрограми

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

- діаграма класів;
- діаграма компонент.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- асоціації (association relationship);
- включення (include relationship);
- розширення (extend relationship);
- узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

Діаграма класів (class diagram) служить для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

В UML існують наступні типи зв'язків які використовуються у діаграмі класів: Асоціації; Агрегація; Композиція.

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовани ромбиком.

Діаграма компонент в UML це діаграма, на якій відображаються компоненти, залежності та зв'язки між ними.

Діаграма компонент відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись.

Модуль програмного забезпечення може бути представлено в якості компоненти. Деякі компоненти існують під час компіляції, деякі – під час компонування, а деякі під час роботи програми.

Діаграма компонент відображає лише структурні характеристики, для відображення окремих екземплярів компонент слід використовувати діаграму розгортання.

Компоненти об'єднуються разом використовуючи структурні зв'язки (assembly connector) щоб об'єднати інтерфейси двох компонент. Це ілюструє зв'язок типу «клієнт-сервер».

Структурна взаємодія – «зв'язок двох компонент, який передбачає, що один з них надає послуги, потрібні іншому компоненту».

При використанні діаграми компонент щоб показати внутрішню структуру компонента, клієнтські та серверні інтерфейси можуть утворювати пряме з'єднання з внутрішніми. Таке з'єднання називається з'єднанням делегації.

Опис програмної системи SAN на основі стандарту FC Gen

Програмна система на мові Python моделює сховище даних рівня SAN, яке працює на основі стандарту FC Gen. Система орієнтується на середовище магістерської випускної кваліфікаційної роботи та відображає логічну структуру типового FC SAN комплексу. Вона включає моделі дискового масиву, FC fabric, зонування, LUN-ів, знімків та реплікації, а також модулі моніторингу і збереження конфігурації у файлі.

Система працює як навчальний симулятор. Вона не взаємодіє з реальним обладнанням, але відтворює ключові сутності та процеси, які відповідають стандарту FC Gen. У центрі системи знаходиться клас SANController, який координує роботу всіх інших компонентів і надає єдиний інтерфейс для операцій розгортання, налаштування та моніторингу SAN.

Моделі предметної області

Система описує фізичний рівень зберігання за допомогою класу Disk. Кожен диск має ідентифікатор, обсяг в гігабайтах, значення швидкості обертання та ознаку працездатності. Таке подання дозволяє моделювати різні типи накопичувачів. Диски об'єднуються у логічні томи.

Логічний рівень представлений класом Volume. Том має власний ідентифікатор, ім'я, розмір у гігабайтах, номер LUN та перелік дисків, на яких він розміщується.

Додатково том містить перелік ідентифікаторів хостів, яким він призначається. Така структура відображає типову схему роботи SAN, коли адміністратор створює LUN й експортує його до одного або кількох серверів.

Сторона обчислювальних ресурсів моделюється класом Host. Хост має ідентифікатор, ім'я та WWN. Поле WWN відображає вимоги стандарту FC Gen

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

щодо глобально унікальних ідентифікаторів FC-портів та вузлів. Через WWN система пов'язує хости з FC портами та зонами.

Транспортний рівень FC SAN описується через клас FCPort. Порт має ідентифікатор, WWN, швидкість у Гбіт за секунду, ознаку доступності та посилання на підключений вузол. Така модель відповідає особливостям стандарту FC Gen, який регламентує швидкості поколінь каналів, формат WWN та поведінку портів у fabric.

Зонування у FC fabric представляє клас Zone. Зона має ідентифікатор, ім'я та список учасників. У якості учасників використовуються WWN FC портів. Це демонструє принцип, за яким адміністратор формує ізольовані групи портів у FC fabric, щоб забезпечити контроль доступу до LUN-ів та сегментацію трафіку.

Усі порти та зони об'єднуються в структуру Fabric. Fabric має ідентифікатор, ім'я, словник FC портів та словник зон. У магістерській роботі така модель дозволяє демонструвати різні варіанти побудови FC fabric, змінювати склад зон та експериментувати з конфігурацією без доступу до реального комутатора.

Рівень захисту та відновлення даних представлений класами Snapshot і ReplicationSession. Знімок містить ідентифікатор, посилання на том, час створення та опис. Реплікаційна сесія містить ідентифікатор, ідентифікатори вихідного і цільового томів, режим реплікації та час останньої синхронізації. Таким чином система моделює базові сервіси безперервності бізнесу, які характерні для промислових SAN.

Моделі моніторингу створюються через класи Metric та Event. Об'єкт метрики містить час, тип об'єкта, ідентифікатор, значення IOPS, затримку в мілісекундах і пропускну спроможність у Мбіт за секунду. Подія містить час, рівень важливості та текстове повідомлення. Такі структури відображають типові показники, які адміністратор відслідковує в реальній системі FC SAN.

Модуль збереження конфігурації

Клас `SANConfigRepository` відповідає за збереження та відновлення конфігурації SAN у форматі JSON. Він отримує шлях до файлу конфігурації при створенні екземпляру. Метод `save` приймає словник з описом поточного стану контролера і записує його у файл з відступами. Метод `load` читає файл, якщо він існує, і повертає стан у вигляді словника.

Такий підхід дозволяє фіксувати створені диски, томи, хости, `fabric`, зони, знімки і сесії реплікації, а потім відновлювати їх для подальших експериментів у лабораторних роботах. У структурі стану окремо зберігаються властивості сутностей та службове поле `next_lun_id`, що забезпечує коректну автогенерацію нових номерів LUN після відновлення конфігурації.

Модуль моніторингу

Клас `MonitoringService` збирає спрощені метрики та події. Він зберігає списки об'єктів `Metric` і `Event`. Метод `register_event` створює нову подію з поточним часом і додає її до журналу. Метод `collect_metric` генерує випадкові значення IOPS, затримки та пропускну здатність для заданого типу об'єкта і додає метрику до колекції.

Метод `get_latest_metrics` повертає останні метрики для відображення в інтерфейсі моніторингу або для аналізу продуктивності. Метод `get_events` дозволяє отримувати всі події або тільки події з певним рівнем важливості. У поєднанні з контролером SAN цей модуль моделює панель спостереження за станом SAN, характерну для промислових систем управління сховищами.

Центральний контролер SAN

Клас `SANController` є основою логіки системи. Він тримає посилання на об'єкт моніторингу, словники дисків, томів і хостів, екземпляр `fabric`, а також словники знімків і сесій реплікації. Поле `next_lun_id` відповідає за послідовне призначення номерів LUN.

Метод `create_disk` створює новий диск із заданою ємністю та швидкістю обертання, реєструє його у словнику дисків та додає подію з описом створення.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

Метод `create_host` створює хост з ім'ям і WWN, реєструє його у системі і записує відповідну подію.

Метод `create_fc_port` створює FC порт із заданим WWN та швидкістю, додає його до `fabric` і фіксує подію додавання порту. У таких методах використовується функція `generate_id`, яка формує унікальні ідентифікатори з використанням бібліотеки `uuid`.

Метод `create_zone` створює зону з ім'ям і переліком WWN учасників та додає її до `fabric`. Подія про створення зони потрапляє до журналу моніторингу. Такий механізм демонструє реалізацію зонування FC SAN, яке визначає, які порти мають право взаємодіяти один з одним у рамках стандарту FC Gen.

Метод `create_volume` формує том на основі списку дисків. Він перевіряє наявність коректних ідентифікаторів дисків, використовує внутрішній лічильник для призначення LUN, додає том до словника томів і реєструє подію створення. Студент бачить, як реалізація LUN відображається у програмній моделі.

Метод `map_volume_to_host` відповідає за призначення LUN конкретному хосту. Він додає ідентифікатор хоста до списку призначень у томі и записує в журнал подію щодо відображення LUN на сервер. У реальній системі така операція відповідає налаштуванню масиву для експорту LUN через FC `fabric` на обраний сервер.

Метод `create_snapshot` створює знімок тому, записує час створення та опис і зберігає його у словнику знімків. Подія формування знімку потрапляє до моніторингу. Метод `create_replication` налаштовує реплікацію між двома томами у заданому режимі, а метод `sync_replication` оновлює час останньої синхронізації та фіксує відповідну подію. У магістерській роботі це демонструє сценарії аварійного відновлення та дублювання даних.

Метод `collect_all_metrics` проходить по всіх дисках, томах і FC портах та викликає модуль моніторингу для генерації метрик. Такий підхід дозволяє з одного виклику отримати агрегований зріз продуктивності всієї SAN.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

Метод `export_state` формує словник з поточним станом усіх сутностей. Для томів додатково формуються списки ідентифікаторів дисків, щоб забезпечити коректне відновлення. Метод `import_state` виконує зворотну операцію – очищає поточні колекції, створює нові об'єкти на основі даних із словника та відновлює лічильник LUN. Це дозволяє зберігати та відновлювати лабораторні конфігурації SAN.

Демонстраційний сценарій

Функція `demo_scenario` демонструє типовий сценарій використання SAN системи у навчальній лабораторії. Вона створює екземпляри модулів моніторингу та центрального контролера, після чого формує набір дисків різного обсягу та швидкості.

Функція створює навчальний сервер з певним WWN, два FC порти з швидкістю 16 Гбіт за секунду, один порт для хоста і один для масиву. Далі порт хоста логічно прив'язується до створеного сервера. Формується зона, яка містить WWN портів хоста і сховища, що відображає типову конфігурацію FC Gen fabric для одного серверного вузла.

Потім сценарій створює два томи `os-datastore` і `data-datastore` різного розміру на основі наявних дисків. Обидва томи призначаються хосту, що моделює видимість LUN з боку гіпервізора. На обох томах створюються знімки початкового стану з текстовими описами. Для одного тому налаштовується асинхронна реплікація на інший том і виконується початкова синхронізація.

Сценарій збирає метрики продуктивності для всіх компонентів, експортує стан контролера у словник і зберігає його у файл через `SANConfigRepository`. Потім сценарій завантажує стан з файлу та відновлює його в новому екземплярі контролера.

Таким чином магістерська робота демонструє повний цикл – створення конфігурації SAN на основі стандарту FC Gen, моделювання роботи, моніторинг, збереження та відновлення.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54


```

ports: Dict[str, FCPort] = field(default_factory=dict)
zones: Dict[str, Zone] = field(default_factory=dict)

@dataclass
class Snapshot:
    id: str
    volume_id: str
    timestamp: float
    description: str

@dataclass
class ReplicationSession:
    id: str
    source_volume_id: str
    target_volume_id: str
    mode: str
    last_sync: float = 0.0

@dataclass
class Metric:
    timestamp: float
    object_type: str
    object_id: str
    iops: int
    latency_ms: float
    throughput_mbps: float

@dataclass
class Event:
    timestamp: float
    level: str
    message: str

class SANConfigRepository:
    #Клас відповідає за збереження та завантаження конфігурації SAN у форматі JSON
    def __init__(self, path: str):
        self.path = path

    def save(self, controller_state: dict) -> None:
        with open(self.path, "w", encoding="utf-8") as f:
            json.dump(controller_state, f, ensure_ascii=False, indent=2)
    def load(self) -> Optional[dict]:
        try:
            with open(self.path, "r", encoding="utf-8") as f:
                return json.load(f)
        except FileNotFoundError:
            return None

class MonitoringService:
    #Клас відповідає за збір спрощених показників продуктивності та подій
    def __init__(self):
        self.metrics: List[Metric] = []
        self.events: List[Event] = []
    def register_event(self, level: str, message: str) -> None:
        event = Event(timestamp=time.time(), level=level, message=message)
        self.events.append(event)
    def collect_metric(self, object_type: str, object_id: str) -> None:
        metric = Metric(
            timestamp=time.time(),
            object_type=object_type,
            object_id=object_id,
            iops=random.randint(100, 5000),
            latency_ms=round(random.uniform(0.3, 5.0), 2),
            throughput_mbps=round(random.uniform(50.0, 800.0), 2),
        )

```

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

```

        self.metrics.append(metric)
    def get_latest_metrics(self, limit: int = 10) -> List[Metric]:
        return self.metrics[-limit:]
    def get_events(self, level: Optional[str] = None) -> List[Event]:
        if level is None:
            return list(self.events)
        return [e for e in self.events if e.level == level]
class SANController:
#Клас координує роботу всіх логічних компонентів SAN
    def __init__(self, monitoring: MonitoringService):
        self.monitoring = monitoring
        self.disks: Dict[str, Disk] = {}
        self.volumes: Dict[str, Volume] = {}
        self.hosts: Dict[str, Host] = {}
        self.fabric = Fabric(id=generate_id("fab"), name="MainFabric")
        self.snapshots: Dict[str, Snapshot] = {}
        self.replications: Dict[str, ReplicationSession] = {}
        self.next_lun_id = 1
    def create_disk(self, capacity_gb: int, rpm: int) -> Disk:
        disk = Disk(id=generate_id("disk"), capacity_gb=capacity_gb, rpm=rpm)
        self.disks[disk.id] = disk
        self.monitoring.register_event("INFO", f"Створюється диск {disk.id}
обсягом {capacity_gb} ГБ")
        return disk
    def create_host(self, name: str, wwn: str) -> Host:
        host = Host(id=generate_id("host"), name=name, wwn=wwn)
        self.hosts[host.id] = host
        self.monitoring.register_event("INFO", f"Реєструється хост {name} з WWN
{wwn}")
        return host
    def create_fc_port(self, wwn: str, speed_gbps: int) -> FCPort:
        port = FCPort(id=generate_id("fc"), wwn=wwn, speed_gbps=speed_gbps)
        self.fabric.ports[port.id] = port
        self.monitoring.register_event("INFO", f"Додається FC порт {port.wwn} зі
швидкістю {speed_gbps} Гбіт/с")
        return port

    def create_zone(self, name: str, members: List[str]) -> Zone:
        zone = Zone(id=generate_id("zone"), name=name, members=members)
        self.fabric.zones[zone.id] = zone
        self.monitoring.register_event("INFO", f"Створюється зона {name} з
учасниками {members}")
        return zone

    def create_volume(self, name: str, size_gb: int, disk_ids: List[str]) ->
Volume:
        selected_disks = [self.disks[d] for d in disk_ids if d in self.disks]
        if not selected_disks:
            raise ValueError("Необхідно вказати хоча б один коректний диск")
        volume = Volume(
            id=generate_id("vol"),
            name=name,
            size_gb=size_gb,
            lun_id=self.next_lun_id,
            disks=selected_disks,
        )
        self.next_lun_id += 1
        self.volumes[volume.id] = volume
        self.monitoring.register_event("INFO", f"Створюється том {name} LUN
{volume.lun_id} розміром {size_gb} ГБ")
        return volume

    def map_volume_to_host(self, volume_id: str, host_id: str) -> None:

```

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

```

    volume = self.volumes[volume_id]
    host = self.hosts[host_id]
    if host_id not in volume.allocated_hosts:
        volume.allocated_hosts.append(host_id)
        self.monitoring.register_event(
            "INFO",
            f"Призначається LUN {volume.lun_id} тому {volume.name} хосту
{host.name}",
        )

    def create_snapshot(self, volume_id: str, description: str) -> Snapshot:
        if volume_id not in self.volumes:
            raise ValueError("Невідомий том")
        snap = Snapshot(
            id=generate_id("snap"),
            volume_id=volume_id,
            timestamp=time.time(),
            description=description,
        )
        self.snapshots[snap.id] = snap
        self.monitoring.register_event("INFO", f"Створюється знімок {snap.id} для
тому {volume_id}")
        return snap

    def create_replication(self, source_volume_id: str, target_volume_id: str,
mode: str) -> ReplicationSession:
        if source_volume_id not in self.volumes or target_volume_id not in
self.volumes:
            raise ValueError("Невідомий ідентифікатор тому для реплікації")
        repl = ReplicationSession(
            id=generate_id("repl"),
            source_volume_id=source_volume_id,
            target_volume_id=target_volume_id,
            mode=mode,
        )
        self.replications[repl.id] = repl
        self.monitoring.register_event(
            "INFO",
            f"Налаштовується реплікація {mode} між томами {source_volume_id} та
{target_volume_id}",
        )
        return repl

    def sync_replication(self, repl_id: str) -> None:
        repl = self.replications[repl_id]
        repl.last_sync = time.time()
        self.monitoring.register_event(
            "INFO",
            f"Виконується синхронізація реплікації {repl.id} між томами
{repl.source_volume_id} та {repl.target_volume_id}",
        )

    def collect_all_metrics(self) -> None:
        for disk_id in self.disks:
            self.monitoring.collect_metric("disk", disk_id)
        for vol_id in self.volumes:
            self.monitoring.collect_metric("volume", vol_id)
        for port_id in self.fabric.ports:
            self.monitoring.collect_metric("fc_port", port_id)

    def export_state(self) -> dict:
        state = {
            "disks": [vars(d) for d in self.disks.values()],
            "volumes": [
                {
                    "id": v.id,

```

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

```

        "name": v.name,
        "size_gb": v.size_gb,
        "lun_id": v.lun_id,
        "disk_ids": [d.id for d in v.disks],
        "allocated_hosts": v.allocated_hosts,
    }
    for v in self.volumes.values()
],
"hosts": [vars(h) for h in self.hosts.values()],
"fabric": {
    "id": self.fabric.id,
    "name": self.fabric.name,
    "ports": [vars(p) for p in self.fabric.ports.values()],
    "zones": [
        {"id": z.id, "name": z.name, "members": z.members}
        for z in self.fabric.zones.values()
    ],
},
"snapshots": [vars(s) for s in self.snapshots.values()],
"replications": [vars(r) for r in self.replications.values()],
"next_lun_id": self.next_lun_id,
}
return state

def import_state(self, state: dict) -> None:
    self.disks.clear()
    self.volumes.clear()
    self.hosts.clear()
    self.fabric = Fabric(id=state["fabric"]["id"],
name=state["fabric"]["name"])
    self.snapshots.clear()
    self.replications.clear()
    for d in state.get("disks", []):
        disk = Disk(**d)
        self.disks[disk.id] = disk
    for h in state.get("hosts", []):
        host = Host(**h)
        self.hosts[host.id] = host
    for p in state["fabric"].get("ports", []):
        port = FCPort(**p)
        self.fabric.ports[port.id] = port
    for z in state["fabric"].get("zones", []):
        zone = Zone(**z)
        self.fabric.zones[zone.id] = zone
    for v in state.get("volumes", []):
        disks = [self.disks[d_id] for d_id in v.get("disk_ids", []) if d_id in
self.disks]
        volume = Volume(
            id=v["id"],
            name=v["name"],
            size_gb=v["size_gb"],
            lun_id=v["lun_id"],
            disks=disks,
            allocated_hosts=v.get("allocated_hosts", []),
        )
        self.volumes[volume.id] = volume
    for s in state.get("snapshots", []):
        snap = Snapshot(**s)
        self.snapshots[snap.id] = snap

    for r in state.get("replications", []):
        repl = ReplicationSession(**r)
        self.replications[repl.id] = repl

```

					БКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

```

        self.next_lun_id = state.get("next_lun_id", 1)

def demo_scenario() -> None:
#Функція демонструє типовий сценарій розгортання та використання SAN у навчальній
лабораторії
    monitoring = MonitoringService()
    controller = SANController(monitoring=monitoring)
    disk1 = controller.create_disk(1000, 7200)
    disk2 = controller.create_disk(1000, 7200)
    disk3 = controller.create_disk(2000, 10000)
    host = controller.create_host("lab-esxi-01", "10:00:00:00:aa:bb:cc:dd")
    fc_port_host = controller.create_fc_port("20:00:00:00:aa:bb:cc:dd", 16)
    fc_port_storage = controller.create_fc_port("30:00:00:00:aa:bb:cc:dd", 16)

    fc_port_host.connected_node_id = host.id

    zone = controller.create_zone("zone-lab-esxi-01", [fc_port_host.wwn,
fc_port_storage.wwn])

    volume_os = controller.create_volume("os-datastore", 500, [disk1.id,
disk2.id])
    volume_data = controller.create_volume("data-datastore", 1500, [disk2.id,
disk3.id])

    controller.map_volume_to_host(volume_os.id, host.id)
    controller.map_volume_to_host(volume_data.id, host.id)
    snap1 = controller.create_snapshot(volume_os.id, "Початковий стан перед
встановленням ОС")
    snap2 = controller.create_snapshot(volume_data.id, "Початковий стан сховища
віртуальних машин")

    repl = controller.create_replication(volume_data.id, volume_os.id, "async")
    controller.sync_replication(repl.id)
    controller.collect_all_metrics()
    state = controller.export_state()
    repository = SANConfigRepository("san_state.json")
    repository.save(state)
    loaded_state = repository.load()
    if loaded_state:
        restored_controller = SANController(monitoring=MonitoringService())
        restored_controller.import_state(loaded_state)
if __name__ == "__main__":
#Точка входу для ручного тестування сценарію
    demo_scenario()

```

4.2 Захист розробленого програмного забезпечення

Дані у системі захищаються за допомогою алгоритму обміну ключа Діффі-Хеллмана.

Ціль алгоритму полягає в тому, щоб два учасники могли безпечно обмінятися ключем, що надалі може використовуватися в якому-небудь

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

алгоритмі симетричного шифрування. Сам алгоритм Діффі-Хеллмана може застосовуватися тільки для обміну ключами.

Алгоритм заснований на труднощі обчислень дискретних логарифмів. Дискретний логарифм визначається в такий спосіб. Уводиться поняття примітивного кореня простого числа Q як числа, чії ступені створюють всі цілі від 1 до $Q - 1$. Це означає, що якщо A є примітивним коренем простого числа Q , тоді числа:

$$A \bmod Q, A^2 \bmod Q, \dots, A^{Q-1} \bmod Q,$$

є різними й складаються із цілих від 1 до $Q - 1$ з деякими перестановками. У цьому випадку для будь-якого цілого $Y < Q$ і примітивного кореня A простого числа Q можна знайти єдину експоненту X , таку, що:

$$Y = A^X \bmod Q,$$

де $0 \leq X \leq (Q - 1)$.

Експонента X називається дискретним логарифмом, або індексом Y , по підставі $A \bmod Q$. Це позначається як $\text{ind}_Q(Y)$.

Тепер опишемо алгоритм обміну ключів Діффі-Хеллмана.

Загальновідомі елементи

Q – просте число.

A – $A < Q$ і A є примітивним коренем Q .

Створення пари ключів користувачем I

Вибір випадкового числа X_i (закритий ключ):

$$X_i < Q.$$

Обчислення числа Y_i (відкритий ключ):

$$Y_i = A^{X_i} \bmod Q.$$

Створення відкритого ключа користувачем J

Вибір випадкового числа X_j (закритий ключ):

$$X_j < Q.$$

Обчислення випадкового числа Y_j (відкритий ключ):

$$Y_j = A^{X_j} \bmod Q.$$

Створення загального секретного ключа користувачем I

$$K = (Y_j)^{X_i} \bmod Q.$$

Створення загального секретного ключа користувачем J

$$K = (Y_i)^{X_j} \bmod Q.$$

Передбачається, що існують два відомих усім числа: просте число Q і ціле A , що є примітивним коренем Q .

Тепер припустимо, що користувачі I і J хочуть обмінятися ключем для алгоритму симетричного шифрування.

Користувач I вибирає випадкове число $X_i < Q$ і обчислює:

$$Y_i = A^{X_i} \bmod Q.$$

Аналогічно користувач J незалежно вибирає випадкове ціле число $X_j < Q$ і обчислює:

$$Y_j = A^{X_j} \bmod Q.$$

Кожна сторона тримає значення X у секреті й робить значення Y доступним для іншої сторони.

Тепер користувач I обчислює ключ як $K = (Y_j)^{X_i} \bmod Q$, і користувач J обчислює ключ як $K = (Y_i)^{X_j} \bmod Q$. У результаті обоє одержать те саме значення:

$$\begin{aligned} K &= (Y_j)^{X_i} \bmod Q = (A^{X_j} \bmod Q)^{X_i} \bmod Q = (A^{X_j})^{X_i} \bmod Q = \\ &\quad \text{за правилами модульної арифметики} \\ &= A^{X_j X_i} \bmod Q = (A^{X_j})^{X_i} \bmod Q = (A^{X_i})^{X_j} \bmod Q = (Y_i)^{X_j} \bmod Q \end{aligned}$$

Таким чином, дві сторони обмінялися секретним ключем. Так як X_i і X_j є закритими, супротивник може одержати тільки наступні значення: Q , A , Y_i і Y_j . Для обчислення ключа атакуючий повинен зламати дискретний логарифм, тобто обчислити:

$$X_j = \text{ind}_{a, q}(Y_j).$$

Безпека обміну ключа в алгоритмі Діффі-Хеллмана впливає з того факту, що, хоча відносно легко обчислити експоненти по модулю простого числа, дуже важко обчислити дискретні логарифми. Для великих простих чисел задача вважається нерозв'язною.

Варто помітити, що даний алгоритм уразливий для атак типу "in-the-middle". Якщо супротивник може здійснити активну атаку, тобто має можливість не тільки перехоплювати повідомлення, але й замінити їх іншими, він може перехопити відкриті ключі учасників Y_i і Y_j , створити свою пару відкритого й закритого ключа $(X_{оп}, Y_{оп})$ і послати кожному з учасників свій відкритий ключ. Після цього кожний учасник обчислить ключ, що буде загальним із супротивником, а не з іншим учасником. Якщо немає контролю цілісності, то учасники не зможуть виявити подібну підміну.

КБПЗ_2025

					VKPM-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено інтерфейс програмного забезпечення, розробленого у результаті виконання магістерської роботи.

Розроблене програмне забезпечення мереж зберігання даних SAN на основі стандарту FC Gen 6 складається з наступних функціональних блоків:

– Навігаційне меню: Файл; Аналізатор мережі; Моделювання мережі; Оцінка продуктивності; Параметри; Довідка.

– Функції представлені у графічному вигляді.

– Розділу вікна підключення мережного інтерфейсу.

– Розділу вікна Інформації про поточний мережний інтерфейс.

– Вікно виведення результату роботи системи.

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Деревовидного представлення зберігання даних SAN на основі стандарту FC Gen 6.

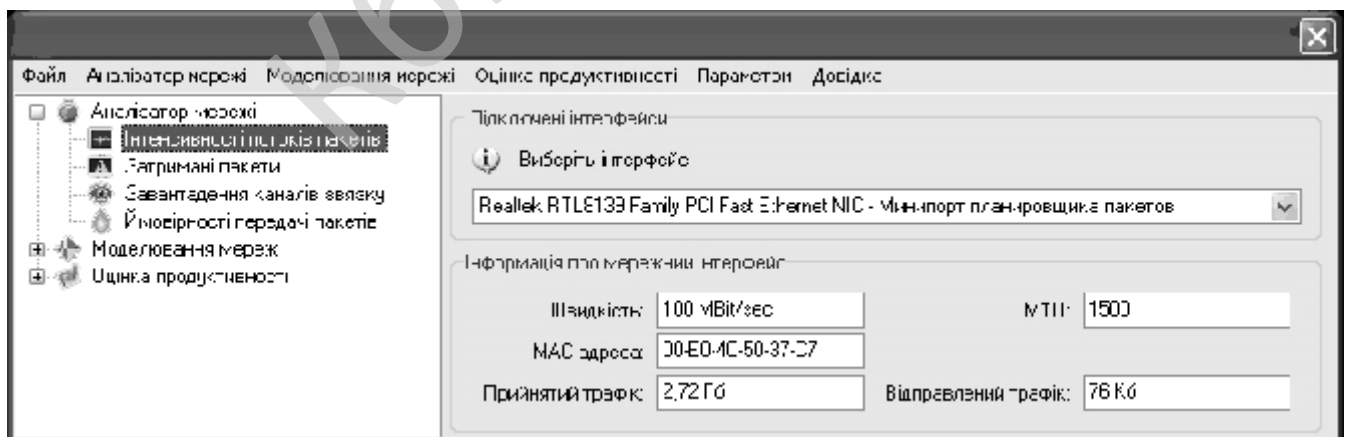


Рисунок 5.1 – Головне вікно розробленого ПЗ

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

- Гарантується перевірка всіх незалежних маршрутів програми.
- Знаходяться гілки True, False для всіх логічних рішень.
- Виконуються всі цикли (у межах їхніх кордонів та діапазонів).
- Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

- Кількість незалежних маршрутів може бути дуже велика.
- Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.
- У програмі можуть бути пропущені деякі маршрути.
- Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

- Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.
- Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.

- При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Обрано умови розповсюдження – commercial software.

Програмне забезпечення, створене комерційною організацією з метою отримання прибутку від його використання іншими, наприклад, шляхом продажу копій.

Найважливішою особливістю комерційних програмних продуктів є підтримка великих компаній, прямо зацікавлених у поширенні програм. Багато організацій надають виключно платну підтримку своїх продуктів, такий підхід, як правило, використовують організації надають відкриті вихідні коди. Для продуктів, що розповсюджуються на комерційній основі діють зазвичай безкоштовні служби підтримки, покликані збільшити рівень довіри у клієнтів і потенційних покупців.

Далеко не завжди, але як правило терміни критично важливих змін в комерційних продуктах значно менше, ніж у некомерційних проектів. Це пов'язано з тим, що над комерційним продуктом працюють цілі групи розробників і ця робота є їх основним заняттям. Розробникам-початківцям як правило доводиться шукати додаткові способи заробітку, і це збільшує час, що витрачається на доповнення і зміни програм. Так як основним рушійним фактором створення комерційного ПЗ є одержання прибутку, то комерційні програмні продукти першими заповнюють вільні ніші та пропонують варіанти вирішення завдань відразу по мірі виявлення вакууму в будь-якому секторі ринку.

Окремий вид комерційних програм, коли їх розробка оплачується безпосередньо замовником. Такі програми найчастіше позбавлені всіх переваг комерційних продуктів, оскільки мають обмежений бюджет, але більш адаптовані до вимог замовника, ніж аналоги.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи зберігання даних SAN на основі стандарту FC Gen.

Метою розробки є дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen.

Об'єктом дослідження є процес зберігання даних SAN на основі стандарту FC Gen.

Предметом дослідження є методи зберігання даних SAN на основі стандарту FC Gen.

Методи дослідження базуються на методах теорії побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод зберігання даних SAN на основі стандарту FC Gen.
- Розроблено вітчизняний продукт зберігання даних SAN на основі стандарту FC Gen, який має більш широкі можливості, на відміну від існуючих аналогів.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та реалізації системи SAN на основі Fibre Channel Gen можуть бути насамперед цікавими великим підприємствам, які щодня працюють з великими обсягами даних – це фінансові установи, державні організації, виробничі корпорації та ІТ-компанії. Для них швидкість доступу, стабільність і безпека інформації є критично важливими, адже будь-яка затримка або втрата даних може спричинити значні збитки. Такі компанії шукають ефективні способи централізованого зберігання даних, що дозволяють скоротити час обробки запитів і забезпечити безперервність бізнес-процесів.

Крім того, дослідження може зацікавити навчальні заклади технічного профілю, адже впровадження подібних систем у лабораторному форматі створює чудову базу для підготовки студентів до реальної роботи в сфері інформаційних технологій та адміністрування мережевих сховищ. Такі проєкти допомагають формувати практичні навички роботи з високошвидкісними інтерфейсами передачі даних, а також розуміння архітектури корпоративних ІТ-рішень.

Водночас результати можуть бути корисними й для малого та середнього бізнесу, який поступово переходить до цифрових рішень. Для таких компаній важливо знайти баланс між вартістю впровадження й отриманими перевагами, а SAN на базі FC Gen дає можливість масштабувати сховище поступово, без необхідності повної заміни інфраструктури.

Також інтерес до таких систем проявляють інтегратори та постачальники ІТ-послуг, які можуть використовувати результати дослідження як основу для створення власних комерційних продуктів або для розробки сервісних пропозицій у сфері корпоративного зберігання даних.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для оцінки привабливості системи SAN на основі Fibre Channel Gen можна застосувати метод експертних оцінок, який передбачає залучення фахівців у сфері мережевих технологій, адміністрування серверів і систем зберігання. Експертам пропонується оцінити систему за кількома критеріями – продуктивність, масштабованість, безпека, сумісність, простота інтеграції та загальна вартість володіння. Кожен експерт виставляє оцінку за шкалою від 1 до 10, після чого вираховується середній бал.

У результаті, наприклад, система може отримати середній показник 8,7 бала. Найвищі оцінки – за продуктивність і стабільність (9,5 бала), трохи нижчі – за складність налаштування та необхідність спеціалізованого обладнання (7,5 бала). Такий результат демонструє, що рішення є технічно привабливим, але потребує високої кваліфікації персоналу для впровадження та підтримки.

Крім кількісної оцінки, експерти також можуть надати якісні коментарі, наприклад, відзначити, що перехід на SAN FC Gen значно підвищує швидкість обміну між серверами, але для максимального ефекту потрібно модернізувати частину комутаційного обладнання. Саме такі зауваження допомагають сформуванню більш повне розуміння сильних і слабких сторін проєкту.

Таким чином, метод експертних оцінок дозволяє не лише визначити загальну привабливість технології, але й отримати практичні рекомендації щодо її впровадження, враховуючи досвід людей, які безпосередньо працюють у цій галузі.

7.3 Вибір методу оцінки вартості ПЗ

Оптимальним методом оцінки вартості системи SAN є витратний підхід, який враховує всі прямі та непрямі витрати, пов'язані з її впровадженням і

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

подальшою експлуатацією. У ньому детально розраховується вартість обладнання (сервери, FC-комутатори, системи зберігання), програмного забезпечення, ліцензій, монтажних робіт, а також витрати на навчання персоналу. Такий підхід дозволяє сформувати реалістичну оцінку початкових інвестицій і планових операційних витрат.

Поряд із цим доцільно застосувати метод аналізу загальної вартості володіння (TCO – Total Cost of Ownership). Він включає в себе не лише стартові інвестиції, а й довгострокові витрати на підтримку, оновлення, енергоспоживання та заміну компонентів упродовж життєвого циклу системи. Завдяки цьому можна визначити, чи є інвестиції економічно обґрунтованими у порівнянні з традиційними рішеннями DAS або NAS.

Також ефективним буде порівняльний метод, який дозволяє зіставити вартість системи SAN на FC Gen з альтернативними технологіями, наприклад, iSCSI або NVMe over Fabrics. Це допомагає обґрунтувати вибір саме FC Gen як оптимального варіанту з точки зору співвідношення "ціна–продуктивність–надійність".

У поєднанні ці методи дають повну картину фінансових аспектів проєкту, допомагаючи не лише визначити його вартість, а й передбачити майбутню економію та вигоди для підприємства.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Підприємство використовує традиційну архітектуру зберігання даних – окремі сервери з локальними жорсткими дисками (DAS). Така система створює проблеми з масштабуванням, резервуванням, продуктивністю та централізованим управлінням. З метою оптимізації прийнято рішення впровадити систему SAN на основі FC Gen 6, що забезпечує високу швидкість передачі (до 32 Гбіт/с), низьку затримку та надійну інфраструктуру для зберігання критичних корпоративних

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

530 000 грн/рік, чистий річний ефект – 1 330 000 грн, термін окупності – 1,8 року (~22 місяці), рентабельність інвестицій – 55 %.

Додаткові нефінансові вигоди: зменшення часу доступу до даних у 8 разів, що підвищує ефективність бізнес-процесів, висока відмовостійкість: автоматичне дублювання зберігання (RAID, multipathing), гнучке масштабування: можна додавати нові сервери без зупинки роботи системи, безперервність бізнесу: відновлення після збоїв відбувається за хвилини, а не години, централізоване управління: усі дані контролюються з єдиної консолі, що знижує ризики людських помилок.

Таким чином, перехід до SAN на FC Gen 6 – це не просто оновлення обладнання, а інвестиція у надійність, швидкість та стабільність інформаційної інфраструктури, що дозволяє компанії працювати ефективніше й упевнено конкурувати на ринку.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування проєкту впровадження SAN має починатися з формування чіткої ціннісної пропозиції для потенційних клієнтів. Потрібно підкреслити, що FC Gen – це не просто чергова технологічна новинка, а реальний інструмент підвищення ефективності бізнесу. Першим кроком може бути створення пілотного проєкту або демонстраційної лабораторії, де клієнти можуть побачити реальні показники продуктивності та стабільності системи.

Наступним етапом є інформаційна кампанія. Вона може включати участь у профільних IT-конференціях, вебінарах, публікації статей і кейсів з результатами тестувань. Особливо важливо донести до аудиторії не технічні деталі, а економічні вигоди – швидку окупність, зниження витрат на обслуговування та зменшення ризику втрати даних.

Також доцільно налагодити партнерство з виробниками обладнання, такими як Dell, HPE, IBM чи NetApp, що дозволить підвищити довіру клієнтів до

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

рішень на основі FC Gen. Крім того, важливо сформувати команду технічних консультантів, які можуть проводити індивідуальні презентації для потенційних замовників.

На завершальному етапі просування слід збирати зворотний зв'язок від перших користувачів, аналізувати їхні результати та формувати рекомендації для подальших проєктів. Такий підхід дозволить не лише розширити ринок, а й підвищити репутацію розробника як надійного постачальника сучасних рішень у сфері зберігання даних.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту важливо використовувати комбінований підхід. Прямий продаж великим корпоративним клієнтам дозволить забезпечити персональний супровід і адаптацію рішення під конкретні бізнес-процеси. Паралельно можна створити партнерську мережу інтеграторів, які пропонуватимуть SAN-рішення у складі своїх ІТ-послуг. Це дозволить охопити ширший ринок без значного збільшення власних витрат.

Не менш ефективним буде використання хмарних моделей розгортання SAN, де клієнт може протестувати або орендувати ресурси зберігання даних, не купуючи обладнання. Це значно спрощує процес прийняття рішення, оскільки компанія одразу бачить практичні переваги технології без великих інвестицій.

Варто також звернути увагу на побудову цифрових каналів комунікації – сайт, онлайн-платформи для демонстрацій, інтерактивні калькулятори ефективності. Це дозволяє потенційному замовнику самостійно оцінити вигоди ще до контакту з відділом продажу.

Таким чином, оптимізація збуту полягає не лише в розширенні каналів, а й у створенні максимально прозорого й комфортного процесу взаємодії з клієнтом – від першого знайомства до впровадження рішення.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		74

7.7 Визначення ключових факторів успіху конкретного проєкту

Успіх реалізації проєкту SAN визначається поєднанням технологічних, організаційних та людських факторів. Насамперед система повинна бути надійною, стабільною та демонструвати реальне підвищення продуктивності порівняно з попередньою інфраструктурою. Це створює довіру клієнтів і підтверджує ефективність обраного рішення.

Другим важливим фактором є професіоналізм команди впровадження. Від якості проєктування, налаштування та підтримки залежить не лише швидкість роботи системи, а й загальна задоволеність користувачів. Навіть найкраще обладнання не дасть результату без грамотного адміністрування.

Крім того, важливу роль відіграє післяпродажна підтримка. Якщо клієнт відчуває, що його супроводжують на всіх етапах, він охочіше рекомендує рішення іншим компаніям. Це створює ефект лояльності та органічного росту ринку.

Нарешті, визначальним чинником є здатність системи масштабуватися й адаптуватися до майбутніх потреб. FC Gen відкриває можливості для подальшого розвитку – інтеграції з хмарними сховищами, системами віртуалізації та новими стандартами зберігання, що робить її стратегічним рішенням на роки вперед.

					VKPM-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Електронно-обчислювальна машина (ЕОМ) відіграє важливу роль у житті сучасної людини. Кожного дня мільйони людей використовують ЕОМ для пошуку необхідної інформації, спілкуванні у соціальних мережах, перегляду новин, роботи тощо. Багато людей користуються ЕОМ у професійних цілях, оскільки завдяки ЕОМ з'явилося багато нових професій.

Аналізуючи умови працівників ІТ-сфери, на перший погляд, може здатися, що працівники сфери інформаційних технологій не схильні до ризиків на виробництві, та якщо більш глибоко розглянути умови і специфіку праці фахівців сфері ІТ-індустрії, можна виявити ряд факторів які будуть мати негативний вплив на стан охорони праці, та на самого ІТ-фахівця зокрема. Сюди можна віднести як невідповідність освітлення, так і високий рівень шуму, що негативно позначатимуться як на емоційному так і на фізичному стані фахівця, призводитимуть до зниження ефективності праці та виробничих травм. Також, важливим моментом охорони праці ІТ-фахівця є врахування його психологічних можливостей (швидкість реакції, особливості пам'яті та уваги, емоційний стан тощо). Для того, щоб забезпечити ефективну роботу ІТ-фахівця, потрібно враховувати та максимально компенсувати такі негативні фактори як: надмірне нервово-емоційне навантаження, довготривалі статичні перевантаження, обмежена рухова активність. Всі ці чинники призводить до різноманітних відхилень у стані здоров'я, зокрема до перевтоми, зниження фізичної та розумової працездатності, неврозів, захворювань серцево-судинної системи тощо. Метою даного розділу є огляд конкретних умов праці спеціаліста у сфері ІТ-індустрії. Завданнями для даного розділу є: аналіз умов праці на робочому місці фахівця ІТ-індустрії, розробка конкретних рекомендацій щодо покращення умов

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

праці фахівців ІТ-індустрії, огляд пожежної безпеки на ІТ-підприємстві та розрахунок системи загального штучного освітлення виробничого приміщення де працюють ІТ-фахівці.

8.2 Аналіз умов праці на робочому місці ІТ-фахівця

На робочому місці ІТ-фахівця (або програміста) виникають небезпечні та шкідливі для безпечної життєдіяльності фактори:

- підвищений рівень шуму;
- несприятливі мікрокліматичні умови;
- недостатній рівень освітленості;
- шкідливі речовини;
- підвищений рівень електромагнітних випромінювань радіочастот;
- висока напруга електричної мережі;
- статична електрика та інші.

Робота програміста супроводжується також підвищеним ступенем напруженості трудового процесу. При систематичному впливі виробничих факторів, які не відповідають нормативним показникам, зростає рівень професійно зумовленої захворюваності працюючих та можуть виникнути професійні захворювання органів зору, руху, нервової системи. Таким чином, вивчення умов праці на робочому місці програміста є необхідною умовою запобігання негативних наслідків впливу небезпечних та шкідливих факторів. Робоче місце, добре пристосоване до трудової діяльності інженера, правильно і доцільно організоване, щодо простору, форми, розміру забезпечує йому зручне положення при роботі і високу продуктивність праці при найменшому фізичному і психічному напруженні.

Нормування параметрів проводиться в залежності від періоду року та категорії важкості виконуваних робіт. Для постійних робочих місць, якими є робочі місця ІТ-фахівців, встановлені оптимальні параметри мікроклімату, а за

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

комп'ютерних класів.

Для підвищення розумової працездатності то зорової роботи повинна здійснюватися ергономічна оптимізація в рамках системи «оператор-термінал», яка сприятиме результативній фізичній та інтелектуальній працездатності і відновленню психосоматичного здоров'я фахівців ІТ-індустрії. Всі наведені заходи щодо вдосконалення охорони праці фахівців ІТ-індустрії повинні контролюватися службою охорони праці та комісією з охорони праці підприємства. Особливе значення у соціальному захисті цієї категорії працівників належить прийняття комплексного договору, який може забезпечити фахівців додатковими пільгами та компенсаціями.

Для більшого розуміння, пропозиції щодо підвищення працездатності ІТ-фахівців, розіб'ємо на декілька категорій:

1 Середовище і розпорядок праці. Для мінімізації негативних ефектів, що пов'язані з перевтомленням ІТ-фахівців, потрібно чітко прописати і реалізувати графік періодів праці-відпочинку, щоб фахівець міг можливість переключити увагу, дати можливість відпочити очам, мозку, елементарно, встати розім'яти ноги. Також потрібно зробити максимально комфортними умови мікроклімату у офісному приміщенні, де працюють ІТ-фахівці. Мається на увазі встановлення і експлуатація, коли виникає необхідність, кондиціонерів, опалення, та системи вентиляції, задля попередження перегрівання, переохолодження ІТ-фахівців, і подальшої неможливості ними виконувати свої функції. Також, за можливості, нами пропонується введення практики віддаленої праці ІТ-фахівцями, якщо роботодавець не може забезпечити оптимальні і безпечні умови в офісному приміщенні, або якщо фахівця вони не влаштовують із певних причин.

2 Фізичні і психоемоційні чинники. Першим і найважливішим чинником, що впливає на працездатність ІТ-фахівців є робоче місце, і саме тому, роботодавець має забезпечити максимальний його комфорт і безпеку.

Гарантією цих факторів може слугувати сертифікація меблів, що використовуються на підприємстві ІТ-галузі. Тому нами пропонується закупівля

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		80

тільки меблів, які пройшли сертифікацію на відповідність. Під психоемоційними чинниками ми розуміємо гарне самопочуття фахівців, позитивний настрій, гарний психологічний клімат у колективі, тощо. Задля того, щоб психоемоційні чинники мали максимально позитивний ефект, керівництву слід поводити заходи, які сприятимуть укріпленню і покращенню міжособистісних стосунків у колективі, таких як психологічні тренінги, таймбілдінг, спортивні змагання і естафети. Також, сюди можна віднести розробку і впровадження системи мотивації працівників, як фінансової, так моральної і адміністративної.

8.4 Розрахунок системи загального штучного освітлення виробничого приміщення де працюють ІТ-фахівці

Приміщення з ПК повинні мати природне і штучне освітлення, яке відповідало б вимогам ДБН В.2.5-28-2006 «Природне і штучне освітлення» [1], ДСанПІН 3.3.2.007-98 «Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [2].

Віконні прорізи повинні бути орієнтовані на північ або на північний схід, забезпечувати коефіцієнт природної освітленості (К.П.О.) не менш 1,5% і мати жалюзі або штори. Віконні прорізи повинні мати регульовані пристрої для відкривання, а також жалюзі, завіски, зовнішні козирки тощо. Приміщення із ПЕОМ повинні бути обладнані системою загального рівномірного освітлення. У виробничих і адміністративно-суспільних приміщеннях, де переважно ведеться робота з документами, допускається комбінована система штучного освітлення.

Штучне освітлення має здійснюватися системою загального рівномірного освітлення, яка включає суцільні або такі, що перериваються лінії світильників, розташованих збоку робочих місць (переважно ліворуч), паралельно лінії зору користувачів ПК. Світильники повинні мати розсіювачі світла. У світильниках місцевого освітлення можна використовувати лампи накаливання. При розміщенні ПК по периметру приміщення лінії світильників штучного освітлення

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		81

Що стосується розподілу яскравості в полі зору працюючих за дисплеями ПК, то відношення значень яскравості робочих поверхонь не повинно перевищувати 3:1.

Проведемо розрахунок штучного освітлення за методом коефіцієнта використання світлового потоку для приміщення ширина якого складає 6 м, довжина – 7 м, висота – 2,9 м. У зазначеному приміщенні працює 7 людей.

Для того, щоб визначити потрібну кількість світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою [1]:

$$F = E \cdot S \cdot K \cdot Z / n,$$

де F – світловий потік, що розраховується, Лм; E – нормована мінімальна освітленість, Лк; $E = 300$ Лк; S – площа освітлюваного приміщення (у нашому випадку $S = 6 \times 6,8 = 40,8$ м²); K – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників у процесі експлуатації (його значення залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку $K = 1,5$); Z – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1... 1.2, в нашому випадку $Z = 1,1$); n – коефіцієнт використання світлового потоку, (відношення світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп і обчислюється в долях одиниці [7]; залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ($\rho_{\text{стін}}$) і стелі ($\rho_{\text{стелі}}$), значення коефіцієнтів дорівнюють $\rho_{\text{стін}} = 50\%$ і $\rho_{\text{стелі}} = 50\%$.

Обчислимо індекс приміщення за формулою:

$$i = S / (h \cdot (A + B)),$$

де S – площа приміщення, $S = 40,8$ м²; h – розрахункова висота підвісу, $h = 3$ м (співпадає з висотою стелі, оскільки лампи освітлення закріплюються на стелі); A – ширина приміщення, $A = 6$ м; B – довжина приміщення, $B = 6,8$ м.

Підставимо всі значення у формулу та визначимо індекс приміщення:
 $i = 1,1$.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

Знаючи індекс приміщення, за знаходимо $n = 0,46$ (з табличних даних коефіцієнтів використання світлового потоку (n) світильників з відповідним типом ламп) [7]. Підставимо всі значення у формулу, визначимо світловий потік: $F=43904$ Лм.

Для розрахунку будемо використовувати *світлодіодні стельові панелі Lezard 6400K 72 Вт.*, світловий потік яких $F_{л} = 7200$ Лм.

Число ламп визначається по формулі:

$$N=F/F_{л}$$

де: F – світловий потік, $F_{л}$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначимо потрібну кількість світильників

$$N= 43904 / 7200=6,09 \text{ шт.}$$

Приймаємо необхідну кількість *світлодіодних світильників 7 шт.*

8.5 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці на робочому місці ІТ-фахівця, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з умов поліпшення охорони праці.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи зберігання даних SAN на основі стандарту FC Gen.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів зберігання даних SAN на основі стандарту FC Gen.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем зберігання даних SAN на основі стандарту FC Gen.

– Досліджена система зберігання даних SAN на основі стандарту FC Gen.

– На основі отриманих результатів досліджень створена програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання зберігання даних SAN на основі стандарту FC Gen.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		85

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Діффі-Хеллмана.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проєкту.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		86

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Семенюк О.А. Дослідження та програмна реалізація системи зберігання даних SAN на основі стандарту FC Gen // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Ramon Nastase «Computer Networking: The Beginner’s guide for Mastering Computer Networking, the Internet and the OSI Model». 2018. – 186 p.

3. Russ White & Ethan Banks «Computer Networking Problems and Solutions: An Innovative Approach to Building Resilient, Modern Networks». 2017. – 832 p.

4. Вінтенко Б., Смірнов О., Миронець І., Смірнова Т., Смірнов С. «Імітаційна модель шляхів вхідних даних комп’ютерної інтелектуальної системи підтримки оператора енергоблоку АЕС». Комбінаторні конфігурації та їхні застосування: Матеріали XXVII Міжнародного науково-практичного семінару, присвяченого 125-річчю Національного університету «Запорізька політехніка» (Запоріжжя-Кропивницький-Київ, 4-6 червня 2025 р.). Запоріжжя: НУ «Запорізька політехніка», 2025. С.82-91.

5. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Drieiev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». International Review on Modelling and Simulations 18 (1), 2025. pp. 32-42.

6. Смірнова Т.В., Коноплицька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». Кібербезпека: освіта, наука, техніка. 2024. №4(24), С. 6-27.

7. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». Підводні технології, 2024, № 13, с. 28-35.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		87

8. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». CEUR Workshop Proceedings, 2023, 3628, pp. 106-115.

9. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». Advanced Information Systems, 2023, 7(2), pp. 49-56.

10. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yanchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». CEUR Workshop Proceedings, Volume 3530, 2023, pp. 256-265.

11. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». Lecture Notes on Data Engineering and Communications Technologies, 2023, 178, pp. 208–223.

12. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». Сучасні інформаційні системи, 2023, том 7, № 2, С. 49-56.

13. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». CEUR Workshop Proceedings Volume 3156, 2022, Pages 390-399.

14. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». Проблеми інформатизації та управління, № 2(70). 2022. С. 28-37.

15. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		88

захисту в інформаційно-комунікаційних системах» Системи управління, навігації та зв'язку, 2022, № 3(69). С. 93-98.

16. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» Вісник Хмельницького національного університету. Серія: «Технічні науки», № 2 (307). С. 46-52. 2022.

17. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» Системи управління, навігації та зв'язку, 2022, № 1(67). С. 84-89.

18. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». Communications in Computer and Information Science, 2021, vol 1486. Springer, Cham. pp 169-184.

19. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

20. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». Journal of theoretical and applied information technology Vol.98. No 21, 2020, P. 3334-3346.

21. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

22. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

23. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». CEUR Workshop Proceedings Volume 2616, 2020, Pages 125-136.

24. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.

25. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

26. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

27. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

28. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019. P.517-522.

29. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyz, M., «QoE optimization technique for media delivery in 5G networks». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.

30. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

31. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

32. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 353-358.

33. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.

34. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

35. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

36. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». Сучасні інформаційні системи. 2021. Т. 5, № 4. С. 79-95

37. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» Вісник Черкаського державного технологічного університету. Технічні науки. №4. С. 103-110. 2020.

38. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.

39. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В., Поліщук Л.І. Інформаційна безпека в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2020. – 294 с.

40. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у Кібербезпека та інформаційні технології: монографія. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

41. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральньоукраїнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.

42. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

43. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		92

D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

44. Смірнов О.А., Дреєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

45. Дреєва Г.М., Смірнов О.А., Дреєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральнуукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

46. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральнуукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

47. Смірнов О.А., Смірнов С.А., Поліщук Л.І., Смірнова Т.В., Коноплицька-Слободенюк О.К. Метод формування антивірусного захисту даних з використанням безпечної маршрутизації метаданих. Кібербезпека: освіта, наука, техніка. – Том 3 № 3. – Київ: КУ ім. Бориса Грінченка. – 2019. – С. 63-87.

48. Смірнов О.А., Гнатюк С.О., Кавун С.В., Терейковський І.А., Жмурко Т.О., Смірнов С.А., Коваленко А.С. Основи безпеки в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2018. – 177 с.

49. Смірнов О.А., Котелянець В.В. Стійкі до колізій стохастичні моделі функціонування безпроводових сенсорних мереж. Вісник інженерної академії України, №3, с. 145-152, 2018

50. Смірнов О.А., Смірнов С.А., Дідик А.К., Дреєв А.М. Алгоритми формування безлічі маршрутів передачі метаданих у антивірусні хмарні системи. Збірник наукових праць "Системи обробки інформації". – Випуск 5 (142). – Х.: ХУПС – 2016. – С. 148-152.

					ВКРМ-123.25.0060.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		93