

Хмельницький національний університет
Факультет програмування
та комп'ютерних і телекомунікаційних систем
Кафедра комп'ютерної інженерії та системного програмування

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень


Кластерна система загального призначення на основі топології «Решітка»
Назва теми

КвРКІ. 170173.17.01.13 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»
Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»
Шифр, назва

Освітня програма «Комп'ютерна інженерія»
Назва

Виконав: студент IV курсу, група КІ-17-1  Ю.В.Коротков
Підпис Ініціали, прізвище

Керівник  Д.М. Медзатий
Підпис, дата Ініціали, прізвище

Нормоконтролер  С.М. Лисенко
Підпис, дата Ініціали, прізвище

До захисту допускаю:
Зав. кафедри комп'ютерної
інженерії та системного
програмування  Т.О. Говоруценко
Підпис Ініціали, прізвище

« 11 » червня 2021 р.

Хмельницький 2021
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ПРОГРАМУВАННЯ ТА КОМП'ЮТЕРНИХ І ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМОГО ПРОГРАМУВАННЯ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЯ ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

“ 11 ” 01 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Короткову Юрію Володимировичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Кластерна система загального призначення на основі топології «Решітка»

Керівник проекту (роботи) Мелзатий Д.М. д.т.н., проф.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 05.02.2021 р. № 11

2. Строк подання студентом проекту (роботи) на кафедру 07.06.2021 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____

Загальні відомості

Проектування програмно-технічного засобу

Програмно-апаратна реалізація та тестування програмно-технічного засобу





5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) _____

Логічна топологія кластерної комп'ютерної системи

Фізична топологія кластерної комп'ютерної системи

Схема розподілу пам'яті

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Лисенко С.М., професор кафедри КІСП		
Антиплагіат	Нічепорук А.О., доцент кафедри КІСП		

7. Дата видачі завдання « 11 » 01 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	11.01.2021	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2021	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2021	виконано
4	Робота над розділом 2 – моделювання та проектування кластерної комп'ютерної мережі	01.04.2021	виконано
5	Робота над розділом 3 – реалізація комп'ютерної кластерної мережі	30.04.2021	виконано
6	Оформлення пояснювальної записки згідно вимог	31.05.2021	виконано
7	Попередній захист ВКР	02.06.2021	виконано
8	Захист ВКР на засіданні ЕК	Червень 2021 року	

Студент


Підпис

Ю.В.Коротков
Ініціали, прізвище





Керівник проекту (роботи)


Підпис

Д.М. Медзатий
Ініціали, прізвище

№ р я д к а	Ф о р м а т	Позначення	Найменування	К і л - л и с т і в	№ ек з	П р и м і т к а
			<u>Текстові документи</u>			
1		КвРКІ 170173.17.01.13 ПЗ	Пояснювальна записка	60		
			<u>Графічні матеріали</u>			
2		КвРКІ 170173.17.01.13 Е8	Логічна топологія кластерної комп'ютерної мережі	1		
		КвРКІ 170173.17.01.13 Е8	Фізична топологія кластерної комп'ютерної мережі	1		
3		КвРКІ 170173.17.01.13 Е8	Схема розподілу пам'яті	1		

КвРКІ 170173.17.01.13 ВП

Зм	Арк	№ докум	Підпис	Дата	Літера	Аркуш	Аркушів
Розробив		Коротков			У	1	1
Перевір.		Медзатий					
Н. конпр.		Лисенко					
Затв.		Говоруханко		11.06.21			

Відомість проекту

ХНУ, КІ-17-1

АНОТАЦІЯ

Тема кваліфікаційної роботи: «КЛАСТЕРНА СИСТЕМА ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ НА ОСНОВІ ТОПОЛОГІЇ «РЕШТКА».

Автор роботи: Коротков Юрій Володимирович.

Керівник роботи: Савенко Олег Станіславович.

Пояснювальна записка: 60 с., 17 рис., 10 табл., 3 дод., 51 джерел.

Графічна частина: 7 презентаційних слайдів.

КЛАСТЕРНА СИСТЕМА ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ НА ОСНОВІ ТОПОЛОГІЇ «РЕШТКА».

Метою роботи є створення кластерної системи загального призначення

Об'єктом дослідження є програмно-технічні засоби для створення кластерної мережі

Предметом дослідження є кластерна система її можливості та її недоліки.

Практичне значення має створення незалежної кластерної системи для прискорення обробки інформації та за для забезпечення втрати файлів та збільшення їх надійності.


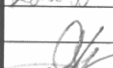



Підпис студента

10.06.2021

Дата

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	4
ВСТУП.....	5
1 ЗАГАЛЬНІ ВІДОМОСТІ	8
1.1 Типи мультипроцесорів	8
1.2 Переваги багатопроцесорних систем	9
1.3 Недоліки багатопроцесорних систем	9
1.4 Підсумки багатопроцесорних систем	10
1.5 Кластерна система.....	10
1.6 Управління кластерами	11
1.7 Переваги та недоліки кластерних систем	12
1.8 Як влаштована робота кластера.....	15
2 МОДЕЛЮВАННЯ ТА ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ	17
2.1 Як побудувати кластер	17
2.2 Типи кластерів	21
2.3 Кластерне обладнання	22
2.4 Налаштування мережі.....	25
2.5 Вузол зберігання.....	26
2.6 RAID-масиви.....	27
2.7 Збірка	30
2.8 Технічне обслуговування	31
2.9 Програмне забезпечення	32
2.10 NLB-кластери	32
2.11 Висновки	37

					КвРКІ 170173.17.01.13 ПЗ			
Зм.	Арк.	№докум.	Підпис	Дата	Кластерна комп'ютерна система загального призначення на основі топології «Решітка»	Літера	Арквш	Арквшів
Виконав		Коротков Ю.В.				у		68
Перевір.		Медзатий Д.М.						
Н.контр.		Лисенко С.М.						
Затвер.		Говорущенко Т.О.		11.06.21				ХНУ КІ-17-1

ДОДАТОК А Копія креслення логічної топології кластерної комп'ютерної мережі	66
ДОДАТОК Б Копія креслення фізичної топології кластерної комп'ютерної мережі	67
ДОДАТОК В Копія креслення схеми розподілу пам'яті	68

					КВРКІ 170173.17.01.13 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		3

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ПК– персональний комп'ютер

WS2016-windows server 2016

IP –Intellectual property (інтелектуальна власність)

Вузол – ПК який підключений до мережі

					КВРКІ 170173.17.01.13 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

ВСТУП

За всю історію обчислювальної техніки не було такого моменту, щоб рівня розвитку обчислювальної техніки було досить для вирішення всіх поставлених перед людством завдань. Постійно ставляться нові, все більш складні завдання, що вимагають все більш потужних обчислювальних ресурсів для свого рішення. І сучасні технології обчислювальної техніки підійшли до рубежу, коли подальший її розвиток роботи індивідуальних пристроїв стає практично неможливим. У зв'язку з цим розвиток обчислювальної техніки пішло по шляху, заснованому на дублюванні обчислювальних пристроїв, які в паралелі можуть працювати над спільним завданням. Разом з цим народилося паралельне програмування, покликане дати можливість ефективно використовувати паралельні архітектури. І сьогодні розробники програмних систем використовують паралелізм на всіх рівнях, починаючи від декількох конвеєрів суперскалярних процесорів, і закінчуючи паралельно працюючими обчислювальними вузлами. Окремий клас паралельних архітектур представляють кластерні системи.

Кластер - це сукупність обчислювальних вузлів, об'єднаних мережею. паралельно кластерна системи являє собою кілька процесів, які спілкуються один з одним по мережі. Таким чином, якщо користувач зможе ефективно розподілити свою задачу між декількома процесорами на вузлах кластера, тобто він отримує вигоду в швидкості роботи, пропорційній числу процесорів. Як правило, кластерні системи вкрай інтенсивно використовуються для проведення обчислень. Підприємства та організації найчастіше купують кластери для вирішення потоку завдань. І найчастіше потреби бажаючих скористатися обчислювальними ресурсами перевершують доступний обсяг ресурсів, тому до кластерів можна спостерігати черги. Ситуація дуже схожа на ту, що існувала з мейнфреймами. У той час для ефективного управління потоками завдань створювалися так звані системи пакетної обробки. Користувачі поміщали свої завдання в чергу цих систем, а за результатом приходили через декількох годин, а іноді і днів. Приблизно те ж саме

					КВРКІ 170173.17.01.13 ПЗ	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

відбувається зараз на кластерах, тому правильний розподіл навантаження по обчислювальним вузлам кластера має дуже велике значення. Це питання набуває ще більшої важливості в разі, якщо кластер має неоднорідну структуру: розрізняється потужність центральних процесорів, обсяг оперативної пам'яті, швидкість ділянок локальної мережі. Якщо не враховувати особливості апаратури, то можна спостерігати, як додаток простоює, чекаючи процесу, який був розподілений на найповільніший обчислювальний вузол. Крім ефективного планування запуску завдань на кластері, необхідно також автоматизувати процеси прийому призначених для користувача завдань, постановки їх в чергу, запуску і збору результатів. Важливо забезпечити безпеку використання кластера, та його відмовостійкість, зробивши при цьому роботу з кластером максимально простий, позбавляючи користувачів від зайвих технічних подробиць. Всі ці фактори призводять до необхідності створення спеціалізованих систем управління кластерами, основна мета яких - надати зручні засоби ефективного використання кластера.

У дінній науковій роботі висвітлюється потреба у розробці кластерної системи загального призначення для масового використання. Розробка кластерної багатопроцесорної системи полягає у потребі збільшення надійності робочої ситеми та прискорення її роботи. Проаналізувавши доступні кластерні системи, мною було вирішено створити кластерну багатопроцесорну систему для загального призначення на основі топології «решітка». З назви даної топології ми розуміємо, що комп'ютери в даній системі будуть з'єднуватись у вигляді решітки, тобто один комп'ютер з'єднується зі всіма сусідніми (приклад зображено на рисунку 1.1).

					КвРКІ 170173.17.01.13 ПЗ	Арк.
						6
Зм.	Арк.	№ докум.	Підпис	Дата		

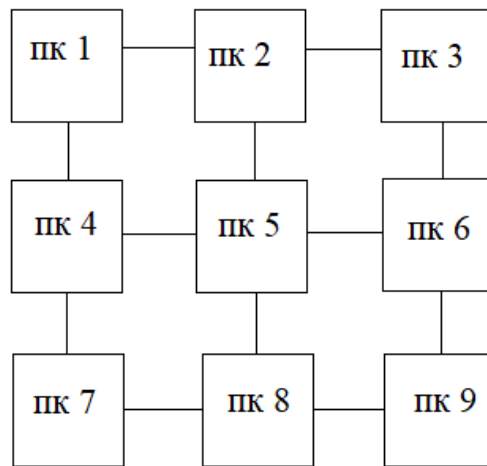


Рисунок 1.1– Схематичний вигляд кластерної системи на топології «решітка»

Дана кластерна система розраховується на не велике підприємство, та буде використовуватись лише у одній фірмі. Вона створюється для запобігання втрати файлів, стабільної роботи та прискорення роботи як програмного забезпечення так і персоналу працівників, що дозволить швидко та якісно виконувати роботу.

В ході чого затрати які були б витрачені на побудову та налаштування багатокомп'ютерної системи у вигляді кластера швидко повернуться, та дозволять в коротші терміни перейти від відпрацювання коштів до заробітку. Для побудови своєї кластерної системи мені знадобляться 9 комп'ютерів які працюють на операційній системі "ос Windows". Ще головних критерієм являється те, що кожен з цих 9 комп'ютерів повинен працювати на процесорі однієї фірми AMD або Intel. Використання процесорів однієї фірми, мінімалізує втрати загальної потужності, полегшить налаштування та контроль. Також це вигідно в плані взаємо замінення компонентів, якщо при виході з ладу 1 або декількох ком-п'ютерів, можна буде їх відремонтувати за рахунок іншого комп'ютера, наприклад, вихід з ладу оперативної пам'яті , через це комп'ютер не буде працювати, тоді ми можемо витягти по одній планці озу та вставити у не працюючий комп'ютер. Обсяг усієї пам'яті не зміниться в системі, але обчислювані потужності системи виростуть.

1 ЗАГАЛЬНІ ВІДОМОСТІ

Більшість комп'ютерних систем є однопроцесорні, тобто вони мають лише один процесор для обробки задач. Але в даний момент часу зростає потреба у багатопроцесорній або паралельній системі. Ці системи мають кілька процесорів що працюють паралельно, які поділять годинник комп'ютера, пам'ять, шину, периферійні присторої. Багатопроцесорні системи використовують для прискорення обробки поставленої задачі, при цьому забезпечуючи єдиний доступ до кожного процесора системи, оптимізуючи використання ресурсів.(Приклад архітектури багатопроцесорної системи зображено на рисунку 1.2).

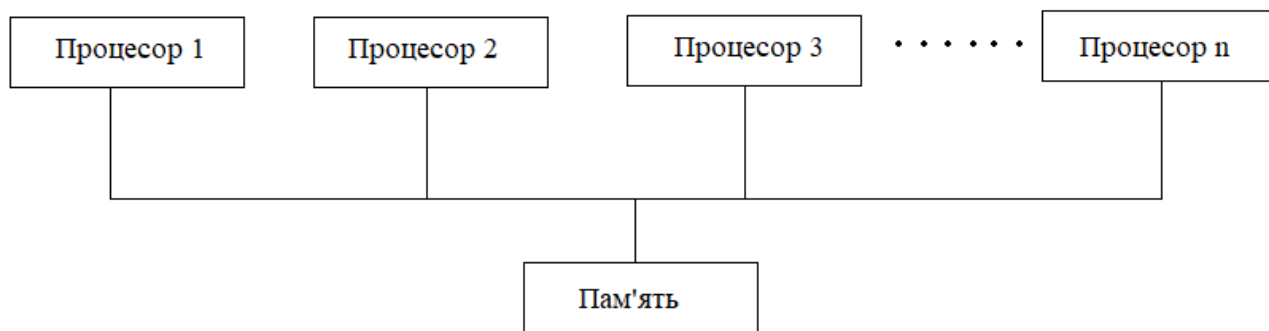


Рисунок 1.2 – Архітектура багатопроцесорної системи

1.1 Типи мультипроцесорів

Існує в основному два типи мультипроцесорів симетричні та асиметричні.

Симетричні системи : у симетричних типах систем кожен процесор містить однакову копію операційної системи і вони спілкуються один з одним. Всі процесори знаходяться в однорангових відносинах, тобто між ними не існує ніяких зв'язків типу головний-рабський. Прикладом симетричної багатопроцесорної системи виступає версія Unix для multimax комп'ютера.

Асиметричні системи: у асиметричних системах, кожному процесору заздалегідь визначається завдання. В цій системі існує головний-процесор який дає інструкції всім іншим процесорам мережі. Асиметрична багатопроцесорна система

містить зв'язок підпорядкованого шаблону, тобто всі процесори системи підпорядковуються одному головному процесору.

1.2. Переваги багатопроцесорних систем

У порівнянні багатопроцесорної та однопроцесорної систем, перша має переваги у швидкості роботи та надійності над іншою. Надійність багатопроцесорної системи полягає у тому, що при виході з ладу одного процесора система продовжує функціонувати, та виконувати поставлену задачу, при цьому звісно швидкість системи падає, але все ж таки вона залишається швидкою за однопроцесорну систему. З данною переваги ми зразу знаходимо ще одну, яка виводить потребу в багатопроцесорній системі виводить потребу в багатопроцесорній системі на інший рівень – це швидкодія. При використанні ресурсів кількох процесорів поставлена задача виконується в кілька разів швидше, що дає змогу раціонально використовувати час для виконання задачі, використання багатопроцесорної системи збільшує продуктивність роботи. За збільшення швидкодії відповідає пропусна здатність. Вона збільшується в стільки раз скільки процесорів працює в тандемі, тобто пропусна здатність кожного процесора сумується, і це сумування призводить до прискорення обробки інформації. Також перевагою багатопроцесорної системи є ціна у довгостроковій перспективі, оскільки процесори поділяють між собою периферійні пристрої, блоки живлення, пам'ять. Також данна система буде мати потужності які будуть задовільняти потреби більшу кількість часу ніж однопроцесорна, це й параметр мінімізує подальші витрати на обслуговування та оновлення заліза.

1.3. Недоліки багатопроцесорних систем

Головним недоліком багатопроцесорної системи є ціна, хоча в довгостроковій перспективі багатопроцесорна система є дешевша, але початкова ціна є

					КвРКІ 170173.17.01.13 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

висока і в цьому плані покупка онопроцесорної системи виглядає більш привабливою.

1.4. Підсумки багатопроцесорних ком'ютерів

При перегляді параметрів багатопроцесорної системи було прийнято рішення, що для складної обчислювальної роботи доцільніше використовувати систему з багатьма процесорами, а системи на базі одного процесора доцільніше використовувати для нескладних задач та задач які базуються на використанні для повсякденного життя, тобто проста офісна робота, використання для входу в мережу інтернет, перегляду відтворення відео аудіо файлів та розваг по типу ігор тому що розробники краще оптимізують свої проекти під однопроцесорні комп'ютери адже вони складають більшу частину ринка комп'ютерів.

1.5. Кластерна система

Існує ще один спосіб пришвидчити та вдосконалити процеси роботи це з'єднання комп'ютерів в кластер. Комп'ютерний кластер відноситься до групи комп'ютерів, які працюють разом над подібним завданням. Комп'ютерний кластер або просто кластер — це декілька незалежних обчислювальних машин, що використовуються спільно і працюють як одна система для вирішення тих чи інших задач. Двома основними причинами кластеризації комп'ютерів є:

1. Підвищення продуктивності або пропускної здатності.
2. Покращення часу безвідмовної роботи, надавши резервний комп'ютер на випадок, якщо основний комп'ютер вийде з ладу.

Наприклад: ми маємо офіс з однопроцесорних комп'ютерів які працюють під управлінням однієї операційної системи, допустимо ос Windows, та виконують відео рендер та рендер складного 3D моделювання. Для обробки одного проекту витрачається приблизно 90% всієї потужності комп'ютера та затрачає в середньому від 1 години до 3. Якщо ми об'єднаємо кожен окремий однопроцесорний комп'ютер

					КВРКІ 170173.17.01.13 ПЗ	Арк. 10
Зм.	Арк.	№ докум.	Підпис	Дата		

у єдиний кластер ми створемо подобу одного багатопроцесорного супер комп'ютера, з єдиною пам'яттю який пришвидчує обчислення, збільшує швидкість обробки даних у пам'яті у стільки стільки разів скільки ком'ютерів під'єднано до кластера, тобто якщо в кластері знаходиться 9 ком'ютерів то швидкість системи виросте близько 9 раз, бо втрати будуть присутні у будь-якому випадку. Звичайно ми не зможемо повноцінно досягти таких же результатів як при використанні багатопроцесорного компютера, але ми прискоримо та та краще захистимо файли, за менший кошт без значних додаткових витрат.

1.6 Управління кластерами

Системи управління кластерами вирішують задачу ефективного управління ресурсами кластера, але це можна робити різними способами, і в даний час використовуються два основних підходи:

1. Створення розподілених оточень для кластерних обчислень (Distributed Cluster Computing Environments, DCCE).
2. Створення систем управління кластерами (Cluster Management Systems, CMS), або систем управління ресурсами і планування (Resource Management and Scheduling, RMS).

Ідея першого підходу полягає в створенні «кластерної» операційної системи, яка б взяла на себе всі питання, пов'язані з ефективним управлінням кластером. Так, додаток не обов'язково виконується на тому вузлі, звідки був зроблений запуск, а може мігрувати на менш завантажений вузол. Також на всіх кластерах використовується загальна файлова система, і користувач може навіть не знати, на якому саме вузлі кластера знаходяться дані. Існує кілька реалізацій подібних оточень: OpenMosix, Kerrighed, OpenSSI, Gluster. ці системи вельми зручні для користувачів, тому як робота в них фактично не відрізняється від роботи в звичайній операційній системі. Недолік полягає в тому, що він практично не

					КвРКІ 170173.17.01.13 ПЗ	Арк. 11
Зм.	Арк.	№ докум.	Підпис	Дата		

реалізується без корекції операційної системи. Саме тому всі подібні оточення існують тільки для UNIX-подібних систем з відкритим кодом (Linux, FreeBSD).

Існує альтернативний спосіб, не пов'язаний з модифікацією операційної системи, це - створення спеціалізованих систем управління кластером. Як правило, використовується наступний порядок роботи: користувач передає системі свій додаток, вказуючи параметри запуску, а система вже сама визначає в який момент і на яких вузлах буде запущено прикладна програма. Тобто системи управління кластерами встановлюються поверх операційної системи і є додатками призначеного для користувача рівня. Обидва викладених підходи мають свої переваги і недоліки. Зауважимо, що перший з них робить ставку на динамічне планування та перепланування (міграція процесів), в той час як в рамках другого підходу використовуються складні алгоритми планування завдань. Варто зауважити, що в даний час більш широко використовується другий підхід. Причина, по всій видимості, полягає в тому, що він не вимагає зміни операційної системи. Розподілені оточення в масі своїй представляють експериментальні зразки, на яких можливі проблеми, пов'язані з несумісністю програмного забезпечення. У такій ситуації зрозумілий вибір користувачів, які віддають перевагу системам під управління кластерами, які працюють на базі звичайних операційних систем.

1.7 Переваги та недоліки кластерних систем

Основне призначення кластера полягає в забезпеченні високого, порівняно з розрізненим набором комп'ютерів або серверів, рівня доступності (High Availability, HA), інакше названим рівнем готовності, а також високого рівня масштабованості і зручності адміністрування. Підвищення готовності системи забезпечує роботу критичних для бізнесу додатків протягом максимально тривалого проміжку часу. До критичних можна віднести всі програми, від яких безпосередньо залежить здатність компанії отримувати прибуток, надавати сервіс

					КвРКІ 170173.17.01.13 ПЗ	Арк. 12
Зм.	Арк.	№ докум.	Підпис	Дата		

або забезпечувати інші життєво важливі функції. Як правило, використання кластера дозволяє гарантувати, що в разі, якщо сервер або який-небудь додаток перестає нормально функціонувати, інший сервер в кластері, продовжуючи виконувати свої завдання, візьме на себе роль несправного сервера (або запустить у себе копію несправного додатка) з метою мінімізації простою користувачів через несправність в системі.

Готовність зазвичай вимірюється у відсотках часу, проведеному системою в працездатному стані, від загального часу роботи. Природно, різні додатки вимагають різної готовності. Готовність системи може бути збільшена різними методами. Який з них вибрати, вирішується в залежності від вартості системи і вартості часу простою. Як правило, дешевші рішення фокусуються в основному на зниженні часу простою після виникнення несправності. Дорожчі дозволяють системі продовжувати функціонувати і надавати сервіс користувачам навіть в тому випадку, коли один або кілька її компонентів вийшли з ладу. Кажуть, що в міру зростання готовності системи її ціна збільшується нелінійно; точно так же, нелінійно збільшується і вартість її підтримки. Відносно низька вартість обертається не найвищим рівнем відмовостійкості - не більше 99%. Це означає, що близько чотирьох днів в році інформаційна структура підприємства буде непрацездатна. На перший погляд, це не так вже й багато, якщо врахувати, що сюди входять і планові простої, пов'язані з проведенням профілактичних робіт або реконфігурацією. Але клієнтові, наприклад, користувачеві системи оплати по кредитних картках, байдуже, з якої причини він буде позбавлений обслуговування. Він залишиться незадоволеним і буде шукати іншого оператора. Висока доступність (готовність) має на увазі рішення, здатне продовжувати функціонувати або відновлювати функціонування після виникнення більшості помилок без втручання оператора. Дорогі відмовостійкі рішення здатні забезпечити заповітні «п'ять дев'яток» - 99,999% надійності системи, що означає не більше 5 хвилин простою на рік.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 13
Зм.	Арк.	№ докум.	Підпис	Дата		

Золоту середину між єдиними серверними системами з віддзеркаленими дисковими підсистемами (або дисковими масивами RAID) і відмовостійкими системами забезпечують кластерні рішення. За рівнем доступності вони наближаються до відмовостійких систем, при зрівнянні, меншою вартістю. Такі рішення ідеальні для випадків, коли можна допустити лише дуже незначні незаплановані простої.

У разі збою кластерної системи відновленням управляє спеціальне програмне і апаратне забезпечення. Зокрема, кластерне програмне забезпечення дозволяє автоматично визначити одиничний апаратний або програмний збій, ізолювати його і відновити систему. Спеціально розроблені підпрограми здатні вибрати найшвидший спосіб відновлення і за мінімальний час забезпечити працездатність служб. За допомогою вбудованого інструментального середовища розробки та програмного інтерфейсу можна створювати спеціальні програми, що виявляють, ізолюють і усувають збої, які виникають в додатках, розроблених користувачем. Інша перевага кластеризації - забезпечення масштабованості. Кластер дозволяє гнучко збільшувати обчислювальну потужність системи, додаючи в нього нові вузли і не перериваючи при цьому роботи користувачів. Сучасні кластерні рішення передбачають автоматичний розподіл навантаження між вузлами кластера, в результаті чого один додаток може працювати на декількох серверах і використовувати їх обчислювальні ресурси.

Типові додатки які експлуатуються на кластерах, це:

- 1) бази даних;
- 2) системи управління ресурсами підприємства (ERP);
- 3) засоби обробки повідомлень і поштові системи;
- 4) засоби обробки транзакцій через Web і Web-сервери;
- 5) системи взаємодії з клієнтами (CRM);
- 6) системи поділу файлів і друку.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 14
Зм.	Арк.	№ докум.	Підпис	Дата		

1.8 Як влаштована робота кластера

Кластер об'єднує кілька серверів, з'єднаних між собою спеціальним комунікаційним каналом, часто названим системною мережею. Вузли кластера контролюють працездатність один одного і обмінюються специфічною інформацією, наприклад, про конфігурацію кластера, а також передають дані між загальними накопичувачами і координують їх використання. Контроль працездатності здійснюється за допомогою спеціального сигналу heartbeat («серцебиття», або «пульс»), в якому вузли кластера передають його один одному, щоб підтвердити своє нормальне функціонування. Якщо в невеликих кластерах heartbeat-сигнали передаються по тих же каналах, що і дані, то в великих системах для цього виділяються спеціальні лінії, так як кластерне програмне забезпечення повинно отримувати сигнал «серцебиття» кожного сервера з певним часовим інтервалом - в разі його неотримання сервер вважається непрацюючим і кластер автоматично переконфігурується. Також автоматично вирішуються конфлікти між серверами, коли при запуску кластера виникає проблема вибору «ведучого» сервера або групи серверів, завдання яких - сформувати новий кластер. В якості комунікаційного каналу кластера можуть використовувати звичайні мережеві технології (Ethernet, Token Ring, FDDI, ATM), що розділяються на шини введення / виводу (SCSI або PCI), високошвидкісний інтерфейс Fibre Channel або спеціалізовані технології CI (Computer Interconnect), DSSI (Digital Storage System Interconnect) або Memory Channel. DSSI-інтерфейс призначений для доступу до накопичувачів і для взаємодії систем між собою. Він схожий на мультихостовий протокол SCSI-2, але володіє більшою продуктивністю і можливістю організації взаємодію комп'ютерів. DSSI-кластери підтримують засоби підвищення надійності системи, поділ ресурсів, розподілену файлову систему і прозорість. З точки зору управління і забезпечення безпеки DSSI-кластер видається єдиним доменом. CI-інтерфейс - це подвійна послідовна шина зі швидкістю обміну до 70 Мбіт / с. Він підключений до системи введення-виведення комп'ютера за допомогою

					КВРКІ 170173.17.01.13 ПЗ	Арк. 15
Зм.	Арк.	№ докум.	Підпис	Дата		

інтелектуального контролера, здатного підтримувати роботу як з подвійної, так і з одинарної шини, в залежності від вимог до надійності доступу для конкретного комп'ютера. Всі лінії зв'язку СІ-інтерфейсу одним кінцем з'єднані з СІ-інтегратором - спеціальним пристроєм, який відслідковує з'єднання з вузлами і конфігурації кластера. Високоєфективна комунікаційна технологія Memory Channel забезпечує високошвидкісний (до 100 Мбайт / с) обмін повідомленнями між серверами в кластері. Вимоги, що пред'являються до швидкодії комунікаційного каналу, залежать від ступеня інтеграції вузлів кластера і характеру роботи додатків. Скажімо, якщо додатки в різних вузлах не взаємодіють один з одним і не здійснюють одночасний доступ до дискових накопичувачів, то вузли обмінюються між собою тільки контрольними повідомленнями, що підтверджують їх працездатність, а також інформацією про зміну конфігурації кластера. Додавання нових вузлів, перерозподілі дискових томів і тому подібне.. Такий тип обміну не жадає значних ресурсів міжмережного з'єднання і цілком може задовольнитися простим 10-мегабітним каналом Ethernet.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 16
Зм.	Арк.	№ докум.	Підпис	Дата		

2 ПРОЕКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ

2.1 Як побудувати кластер

Розгортання кластера як такого – завдання досить просте. Причому, для цього підійде будь-який дистрибутив Linux за вашим вибором. Який саме з дистрибутивів Linux ставити в якості базової операційної системи - не має значення, Ubuntu, Mandriva, Alt Linux, Red Hat, SuSE. Вибір залежить тільки від уподобань. Покрокове розгортання кластера, використовуючи дистрибутив загального призначення:

1. Встановити операційну систему на комп'ютер, який буде виступати в ролі консолі кластера. Тобто на цьому комп'ютері будуть компілюватися і запускатися паралельні програми. Іншими словами, за цим комп'ютером буде сидіти людина, запускати програми і дивитися, що вийшло.

2. Після інсталяції базової ОС на консолі кластера, якщо це не зроблено в процесі початкової установки, ми повинні будемо встановити необхідні компілятори (фортран, С) і всі необхідні бібліотеки, desktop environment (GNOME або KDE за вашим вибором), текстові редактори та ін., Тобто перетворити цей комп'ютер в робочу станцію розробника.

3. Встановити з репозиторію або з початкових кодів пакет MPICH або OpenMPI.

4. Описати в / etc / hosts майбутні вузли вашого кластера, в тому числі і консолькластера.

5. Встановити NFS і розширити для всіх вузлів кластера якусь директорію, в якій будуть розміщуватися виконувані модулі паралельних програм і.

6. Файли даних, якими ці програми будуть користуватися в процесі своєї роботи.

7. Встановити на консолі кластера ssh-клієнт (обов'язково) та ssh-сервер, якщо будемо давати доступ до консолі кластера по мережі).

					КВРКІ 170173.17.01.13 ПЗ	Арк. 17
Зм.	Арк.	№ докум.	Підпис	Дата		

8. На всіх вузлах кластера встановити операційну систему, бібліотеки, необхідні для виконання призначених для користувача паралельних програм.

9. Встановити MPICH, NFS-client, ssh-server. Вузли кластера з метою економії ресурсів повинні завантажуватися в runlevel 3, так що ставити туди GNOME або KDE не треба. Максимум - поставити туди гномовські або кдешні бібліотеки, якщо вони потрібні для користувача програми.

10. Описати в / etc / hosts всіх вузлів кластера майбутні вузли нашого кластера, в тому числі і консоль кластера.

11. На всіх вузлах кластера необхідно автоматом при завантаженні монтувати розшарений в п. 5 ресурс. Причому, шлях до цього ресурсу повинен бути однаковий, як на консолі кластера, так і на його вузлах. Наприклад, якщо на консолі кластера ми розширюємо каталог / home / mpiuser / data, то на вузлах кластера цей ресурс також повинен бути змонтований в / home / mpiuser / data.

12. На всіх вузлах кластера забезпечити безпарольний доступ по ssh для консолі кластера.

Кластер зібраний і повністю готовий до використання. Фактично для розгортання кластера нам треба було встановити ОС, зайти в так би мовити в "Установку і видалення програм", відзначити для установки пакети SSH і MPICH, заборонити запит пароля віддаленого доступу до вузлів кластера, розшарити на центральному вузлі каталог, де будуть зберігатися наші паралельні програми і дані і поставити на вузлах кластера автоматичне підключення до цього каталогу при завантаженні. Як компілювати і запускати на виконання паралельні програми Ви можете подивитися в інших розділах цього сайту і в документації до MPICH. Все дуже просто і нічого, крім дистрибутива з репозиторіями не потрібно.

Тепер я хотів би обговорити інше питання, а саме побудову мережі кластера. Оскільки мережа - найвужче місце і від неї прямо залежить ефективність роботи кластера, то хотілося б зробити наступне. Хотілося б, щоб функціонування мережевої файлової системи NFS не заважало обміну даними, які виконують між

					КВРКІ 170173.17.01.13 ПЗ	Арк. 18
Зм.	Арк.	№ докум.	Підпис	Дата		

собою частини паралельної програми, що працюють на різних вузлах. Щоб це здійснити, необхідно фізично розділити мережу на два сегменти. В одному сегменті буде працювати NFS, в іншому - буде відбуватися обмін даними між частинами програми. Таким чином і в консолі кластера і в його вузлах необхідно мати два мережевих інтерфейса (Дві мережеві карти). Відповідно, потрібно два набору світчей, не пов'язаних один з одним, і два набори мережевих реквізитів для цих інтерфейсів. Тобто, NFS працює, наприклад, в мережі 192. 168.1.0/24, а обмін даними відбувається в мережі 192.168.2.0/24. І відповідно, в файлах / etc / exports і / etc / fstab повинні будуть бути прописані адреси з першої мережі, а в файлах / etc / hosts і в файлу machines.LINUX, що описують кластер – адреси з другої. Файли, що зберігаються на диску, в умовах паралельного завдання, що виконується на кластері, можуть знадобитися тільки для збереження стану завдання в контрольних точках. Звичайно, дискові ресурси можна використовувати і для організації віртуальної пам'яті, довантажуючи в міру необхідності дані в оперативну пам'ять, збільшуючи тим самим розмір сітки. Однак при наявності кластера, збільшення розміру сітки, логічніше й ефективніше може бути виконано використання додаткових обчислювальних вузлів кластера. Якщо ж дискові ресурси використовуються тільки для збереження контрольних точок і ці контрольні точки розташовані не в кожній ітерації (а в кожній десятій або сотій), то поділ локальної мережі кластера на два незалежних сегмента (NFS і мережу міжпроцесорного обміну даних) є не обов'язковою. Цілком можна обійтися всього одним сегментом, використовуючи його і для NFS і для обміну даними. Оскільки NFS буде використовуватися досить рідко, то і негативний вплив її на ефективність кластера буде мінімальний.

Настійно рекомендується використовувати гігабітну мережу, як найбільш доступну для університету (з точки зору фінансів). Точніше кажучи, Gigabit Ethernet – НЕ кращий вибір в якості мережі кластера в силу того, що ця мережа має досить велику латентність. Але це доступне рішення. Якщо ж фінанси дозволять, то краще звичайно звернути увагу на Myrinet або 10Gbit Ethernet.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 19
Зм.	Арк.	№ докум.	Підпис	Дата		

Параметри мережі ніколи не бувають надто хороші. Тому, якщо є можливість, треба намагатись їх поліпшити. Якщо Myrinet або 10GbE будуть недоступні, то можна спробувати поліпшити характеристики гигабітної мережі. власне, зробити з неї двухгігабітну. Суть справи в тому, що замість однієї мережевої карти ми використовуємо дві, об'єднавши їх спеціальним драйвером в єдиний віртуальний канал з подвійною пропускною спроможністю. В цьому випадку карти повинні бути підключені до двох окремим свічів, тобто потоки по цих картах ми поділяємо так само, як ми це робили раніше, розділяючи NFS і передачу даних. Створення такого каналу довге заняття. В принципі, це не обов'язково, хоча ефект буде помітний. Тепер власне про те, а навіщо взагалі потрібен кластер. Справа в тому, що твердження «Чим більше вузлів в кластері, тим швидше він працює» - в загальному випадку не вірно. Давайте подивимося, в яких випадках нам захочеться завантажувати наші програми на кластері. Існує тільки дві мети використання кластера:

1. Є сітка розміру R , обчислення на якій при використанні звичайного комп'ютера займають час T . Час T - критичний параметр. Нам хочеться істотно зменшити час обчислень, маючи R як константу.

2. Є сітка розміру R , обчислення на якій при використанні звичайного комп'ютера займають час T . Час T - не критично. Нас цікавить збільшення розміру сітки понад наявної. Всі обчислення на сітці мають один загальний і важливий для нас параметр: час однієї ітерації. У разі використання кластера цей час складається з двох частин: час рахунку на сітці T_{iter} і час обміну даними між вузлами T_{exch} . T_{iter} залежить тільки від потужності процесора. А ось T_{exch} залежить вже, від розміру сітки, кількості вузлів кластера і пропускної властивості.

					КВРКІ 170173.17.01.13 ПЗ	Арк.
						20
Зм.	Арк.	№ докум.	Підпис	Дата		

2.2 Типи кластерів

Комп'ютерні кластери скомпоновані таким чином, щоб забезпечити різне призначення: від бізнес-потреб загального призначення, таких як підтримка веб-служб, до інтенсивних наукових обчислень. В основному є три типи кластерів:

1) Кластер балансування навантаження - кластеру потрібна ефективна можливість балансування навантаження між доступними комп'ютерами. При цьому вузли кластера спільно використовують обчислювальне навантаження, щоб підвищити загальну продуктивність. Наприклад, високопродуктивний кластер, який використовується для наукових розрахунків, буде врівноважувати навантаження від різних алгоритмів з кластера веб-сервера, який може просто використовувати метод циклічного перебору, призначаючи кожен новий запит іншого вузла. Використовується цей тип кластера на фермах веб-серверів (web farm).

2) Відмовостійкі кластери. Функція перемикання додатків і ресурсів даних з відмовившої системи на альтернативну систему в кластері називається відмовостійкою. Ці типи використовуються для кластеризації бази даних критичних серверів, пошти, файлів і додатків.

3) Кластери високої доступності - вони пропонують високу ймовірність того, що всі ресурси будуть в експлуатації. Якщо відбувається збій, через збій системи або втрати тому диска, запити втрачаються. Будь який втрачений запит, якщо він буде повторений, буде обслуговуватися іншим комп'ютером в кластері. Цей тип кластера широко використовується в мережі, електронній пошті, новинах або FTP-серверах.

Переваги:

1. Абсолютна масштабованість - можна створювати великі кластери, що перевершують потужність навіть найбільших автономних машин. Кластер може мати десятки багатопроцесорних машин.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 21
Зм.	Арк.	№ докум.	Підпис	Дата		

2. Додаткова масштабованість - кластер налаштований таким чином, що можна додавати нові системи в кластер з невеликим кроком. Кластери мають можливість додавати системи по горизонталі. Це означає, що до кластерів може бути додано більше комп'ютерів, щоб підвищити його продуктивність, надмірність і відмовостійкість (здатність системи продовжувати роботу з несправним вузлом).

3. Висока доступність - оскільки ми знаємо, що кожен вузол в кластері є автономним комп'ютером, відмова одного вузла не означає втрату обслуговування. Один вузол можна відключити для обслуговування, тоді як інші кластери беруть на себе навантаження цього окремого вузла.

4. Бажана ціна / продуктивність. Кластери, як правило, налаштовуються для підвищення продуктивності і доступності на окремих комп'ютерах, при цьому, як правило, вони набагато більш економічні, ніж окремі комп'ютери порівняно з швидкістю або доступністю.

2.3 Кластерне обладнання

Існує багато апаратних компонентів, які допомагають створити кластер. Можна взагалі розбити на шість загальних категорій: самі комп'ютери, мережеві витратні матеріали, фізичне зберігання комп'ютерів, обладнання для розподілу електроенергії, консоль для доступу до комп'ютера та запчастин.

Комп'ютери - Кластери можна створювати із загальних комп'ютерів. Але для великих кластерів, можуть бути комп'ютери, спеціально розроблені для високопродуктивних обчислень придбані у таких постачальників, як Supermicro. Ці системи є кращими за базові ПК, оскільки вони поставляються в упаковках високої щільності, таких як 1 або 2 U, що встановлюються в стійку корпусів (U становить 1,75 "вертикального простору в стійці). У будь-якому випадку комп'ютери повинні працювати добре з Linux. Для масштабованості обчислювальні вузли повинні підтримувати завантаження PXE та IPMI контроль.

Головний вузол, як правило, більш продвинутий, ніж обчислювальні вузли, оскільки він є точкою входу для всього кластера. Він повинен підтримувати RAID-масиви жорстких дисків і повинен мати принаймні два порти Ethernet, один для підключення до зовнішньої мережі, а інший - для ізольованої внутрішньої обчислювальної вузлової мережі.

Обчислювальні вузли в ідеалі повинні бути малими, дешевими та продуктивними. Залежно від застосування кластера, вони можуть мати будь-яку комбінацію потужних процесорів, великий обсяг оперативної пам'яті, великих жорстких дисків або графічних процесорів.

Мережа - Ваш комутатор мережі повинен мати принаймні стільки портів, скільки у вас обчислювальних вузлів. Додаткові порти завжди зручніші, якщо ви вирішите додати до свого кластера у майбутньому ще один вузол. Також необхідні мережеві кабелі.

Фізична конфігурація - Кластер можна створити з ПК на полиці. Однак професійно побудований кластер зазвичай використовує спеціальні монтовані комп'ютери. Комп'ютери часто стоять на рейках, що дозволяє висунути їх настільки далеко, щоб зняти кришку не відриваючи їх фізично від стійки. Бажано залишити достатньо вільного місця для кабелів на задній панелі комп'ютерів, щоб вони могли працювати під час витягування для діагностичної цілі.

Важливим фактором є місце розташування кластера. Скупчення може бути досить галасливим завдяки вентиляторам, тож потрібно ставити його там, де білий шум не буде заважати. А скупчення також може виробляти багато тепла. Якщо виділення велике, вам знадобиться вентиляція або спеціальний пристрій, блок кондиціонування.

Розподіл енергії - комп'ютери забирають багато енергії, а багато комп'ютерів багате споживання. Схеми, на яких вони працюють, повинні мати можливість обробляти дані, і вам знадобиться живлення смужки для розподілу потужності. Якщо ваш кластер невеликий, кілька роз'ємів живлення споживчого класу має бути достатньо. В іншому випадку існують великі стійкові монтажні силові смуги, які

					КВРКІ 170173.17.01.13 ПЗ	Арк. 23
Зм.	Арк.	№ докум.	Підпис	Дата		

подають струм. Крім того, головний вузол і блок зберігання слід підключити до безперебійного джерела живлення, щоб вони не зупинилися внаслідок відключення електроенергії, та запобігти псування даних.

Доступ - непрактично мати клавіатуру, відеомонітор і мишу для кожного вузла кластера. Однак, хороша ідея – підключити монітор, мишку, клавіатуру до головного вузла. Це гарантує, що ви завжди зможете отримати доступ до свого кластера, та виконувати адміністративні завдання. Існують спеціальні вироби, такі як кріплення в стійку РК-монітор та клавіатура, які можуть слугувати цій меті.

Після налаштування кластера ви зможете отримати доступ до обчислювальних вузлів за допомогою SSH з головного вузла. При нормальній роботі вузли можуть бути безголовими (працювати без клавіатури, монітора та миші). При ненормальній роботі, наприклад, при початковій установці кластера або коли діагностуючи проблему, ви можете отримати доступ до обчислювальних вузлів за допомогою аварійного кошика, який є мобільним пристроєм який містить монітор мишку та клавіатуру з довгими кабелями, які можна підключити до будь-якого вузла, що має проблеми. Інший варіант це комутатори можуть використовувати стандартні кабелі вводу-виводу (такі як USB і VGA), або вони можуть працювати повністю через IP, якщо це підтримує BIOS комп'ютерів. Чим більше вузлів задіяно, тим дорожче будуть апарати які містять монітор, клавіатуру, мишку та кабелі.

Запасні - Матеріали розбиваються. Коли у вас багато речей (як у кластері), вони часто ламаються. Наприклад, скажімо, у вас є 100 жорстких дисків серед усіх комп'ютерів у вашому кластері, і кожен жорсткий диск призначений для роботи протягом 20 років. Це річний коефіцієнт відмов 5%, тож ви можете розраховувати, що 5 з них не вдасться задіяти через рік, це приблизно один раз на 10 тижнів. Цей аналіз застосовується до всіх комп'ютерних компонентів, а це означає, що крім запасних жорстких дисків, бажано також придбати запасну оперативну пам'ять, блоки живлення і, можливо, навіть материнські плати та процесори. У великому кластері доцільно мати запасне мережеве обладнання, а у виробничих умовах цілий

					КВРКІ 170173.17.01.13 ПЗ	Арк. 24
Зм.	Арк.	№ докум.	Підпис	Дата		

запасний головний вузол. Запчастини для ремонту обчислювальних вузлів не є настільки необхідними, оскільки дисфункціональні вузли можуть бути просто відключені в офлайновому режимі або канібалізовані на частини. Таким чином, частини, необхідні для побудови власного кластера, такі:

1. RAID-пам'ять (інтегрована безпосередньо в головний вузол або вузол зберігання, або як окремий пристрій).
2. Обчислювані вузли.
3. Мережеві комутатори.
4. Мережеві кабелі.
5. Стійка та кріпильне обладнання.
6. Розетки.
7. Джерело безперебійного живлення.
8. Екран, мишка, клавіатура та кабелі.
9. Комплект запасних частин (жорсткі диски, оперативна пам'ять, блоки живлення).

2.4 Налаштування мережі

Як тільки зупинитись на апаратному забезпеченні, потрібно спланувати, його підключити. Почнемо з спілкування. Спочатку дамо кластеру ім'я. Імена використовуються як псевдоніми для IP-адрес, що значно полегшує людині ідентифікацію окремих комп'ютерів у мережі. Скупчення комп'ютер використовує дві різні мережі: зовнішню мережу (вона ж "Інтернет"), та внутрішню.

Головний вузол підключається до внутрішньої мережі, яку кластер використовує для внутрішнього спілкування. Отже, потрібно налаштувати два імена, одне для зовнішньої мережі та одне для внутрішньої мережі.

Зовнішня мережа - використовується лише головним вузлом. Назва на зовнішньому Мережа зазвичай форматується як `hostname.domain.suffix`, де є ім'я

					КВРКІ 170173.17.01.13 ПЗ	Арк. 25
Зм.	Арк.	№ докум.	Підпис	Дата		

хосту все, що завгодно, і domain.suffix відноситься до організації, що використовує кластер.

Внутрішня мережа - використовується усіма вузлами кластера. Внутрішнє ім'я зазвичай має компонент імені хосту із зовнішнього імені, що використовується разом з нумерацією схеми. Наприклад, ми додаємо дві цифри до кінця імені хосту для кожного вузла: name00 (головний вузол), name01 (перший обчислювальний вузол) тощо. Ця схема обмежує нас на 100 вузлів, але їх можна легко розширити для подальшого оновлення. Іменування комп'ютерів життєво важливо для людей, щоб мати можливість підтримувати кластер, але комп'ютери самі мають справу з числовими IP-адресами. Спосіб отримання IP-адреси на зовнішню мережу залежить від адміністратора мережі, але ви отримуєте повне панування над внутрішньою мережею. Існує два методи призначення IP-адрес у внутрішній мережі: статичне та динамічне.

Статичне призначення - Кожен обчислювальний вузол налаштовується індивідуально із власною IP-адресою. Це займає багато часу, оскільки налаштування проводиться вручну. IP-адреси для великої кількості вузлів не є практичними.

Динамічне призначення - Кожен обчислювальний вузол має однакову конфігурацію та приймає його IP-адресу від головного вузла через мережу на основі його унікального MAC-адресу.

2.5 Вузол зберігання

До цього часу в описі кластера я згадував один головний вузол, який діє як точка доступу до кластера, разом з безліччю обчислювальних вузлів для виконання завдань кластера. Багато великих кластерів будуть додатково відокремлювати завдання, особливо якщо головний вузол стає вузьким місцем для роботи кластера. В такі моменти можна прийняти рішення мати окремий вузол зберігання для управління файлами, до яких обчислювальні вузли потребують доступу,

					КВРКІ 170173.17.01.13 ПЗ	Арк. 26
Зм.	Арк.	№ докум.	Підпис	Дата		

наприклад, програма програмне забезпечення та домашній каталог кожного користувача.

2.6 RAID- масиви

При зберіганні великих обсягів даних настійно рекомендується використовувати RAID (резервний масив незалежних дисків). Це може бути інтегровано безпосередньо в головний вузол, а окремий компонент, підключений до головного вузла, або частина окремого вузла зберігання. RAID-пристрій працює, поєднуючи кілька невеликих фізичних дисків, щоб сформувати один більший і швидший віртуальний диск. RAID-масив може також забезпечити надмірність даних, що дозволяє диску вийти з ладу зберігаючи дані. Це абсолютно життєво важливо у виробничих умовах. Як було згаданий раніше, з багатьма компонентами виникають часті несправності компонентів. Відмова одного обчислювального вузла - це не кінець світу, оскільки на ньому немає нічого того, чого не може бути втрачено без відновлення, тому кластери часто використовують по одному жорсткому диску для кожного обчислювального вузла. Однак втрата всіх додаткових або користувацьких даних кластера стане катастрофою, що призведе до проблем. Є кілька часто використовуваних рівнів RAID, які досягають збільшеного розміру та швидкості, або для досягнення цих цілів.

1. RAID 0 забезпечує розбиття пам'яті та збільшення продуктивності за рахунок «розмежування» даних через два або більше дисків. Це означає, що послідовні сегменти даних зберігаються на різних дисках. Це може значно покращити час читання в деяких додатках, проте одна неполадка одного з дисків призводить до втрати всіх даних. Привід RAID 0 такий же великий, як і його розмір найменший привід.

2. RAID 1 забезпечує надійність, не збільшуючи при цьому обсяг пам'яті та продуктивність. Способом досягнення надійності являється "дзеркальне відображення даних". Данні записується на два або більше дисків, дозволяючи

					КВРКІ 170173.17.01.13 ПЗ	Арк. 27
Зм.	Арк.	№ докум.	Підпис	Дата		

одному залишитись в нерухомому стані. Розмір накопичувача RAID 1 такий же, як і найменший.

3. RAID 5 схожий на RAID 0, за винятком того, що він включає надлишкове поширення інформації про паритету на три або більше дисках. Це дозволяє будь-якому одному диску вийти з ладу без втрати даних. RAID в цілому буде зберігати стільки, скільки найменший накопичувач.

4. RAID 6 схожий на RAID 5, за винятком того, що він має два диски, що мають резервну парність, інформація розповсюджується на чотири або більше накопичувача. Це дозволяє двом дискам вийти з ладу без втрати даних. Зберігання буде обмежене розміром найменшого накопичувача, удвічі меншим ніж кількість накопичувачів.

5. Hot Spares - це порожні накопичувачі, що входять до складу пристрою RAID 5 або 6. Коли один диск виходить з ладу, RAID пристрій відновлює інформацію, що раніше була на цьому накопичувачі, на запасний за допомогою надлишку інформації, яка розповсюджується на інших дисках. Якщо гарячі запасні частини не включені, цей процес буде починаються лише тоді, коли вручну замінити несправний диск. Якщо не можете замінити диск, або немає запасного диску, це грозить виходом з ладу додаткового накопичувачами та призвести до втрати даних перед. Незалежно від того, скільки гарячих запасних частин ви забезпечите собі, ваші дані не є на 100% захищеними від втрати, та втрати жорсткого диска. Використання RAID-пристрою може зменшити ймовірність втрати даних, але це не усуне втрату. Отже, якщо дані взагалі важливі (потрібно докладати зусиль до побудови кластера), переконайтесь, що у вас є доступ до іншого комп'ютера в іншому фізичному місці, куди можна створити резервну копію файлів.

6. Накопичувач RAID 7 влаштований досить складно. По суті, він являє собою спеціалізований комп'ютер, що працює під управлінням власної операційної системи. Основу архітектури RAID 7 становить внутрішня шина X-bus з пропускною спроможністю 80 Мб / с. Для порівняння зазначимо, що пропускна

					КВРКІ 170173.17.01.13 ПЗ	Арк. 28
Зм.	Арк.	№ докум.	Підпис	Дата		

здатність найбільш популярного в даний час каналу SCSI-2 становить 20 Мб / с, а швидкість зчитування даних з фізичного жорсткого диска зазвичай не перевищує 35 Мб / с.

Тому навіть одночасне підключення до RAID 7 чотирьох комп'ютерів ніяк не позначиться на часі доступу до даних. На практиці запити на доступ до даних ніколи не надходять одночасно, так що пристрої, що випускаються в даний час під торговою маркою RAID 7, допускають підключення до 12 комп'ютерів.

RAID 7 випускається в трьох виконаннях: настільному (дозволяє використовувати від 3 до 12 фізичних дисків), монтується в стійку (від 3 до 27 дисків) і "консоль" (від 12 до 48 дисків). Відмінною особливістю архітектури RAID 7 є те, що користувач може встановлювати в пристрій фізичні диски будь-якої ємності, будь-якого швидкодії, будь-якого виробника (зрозуміло, у виробників системи є "чорні списки" дисків, так як часто навіть "однойменні" диски однієї фірми мають ледь відрізняються один від одного модифікації, вельми істотно відрізняються експлуатаційними характеристиками). Ємність дискового простору RAID 7 в даний час може становити від 5 до 420 Гб.

Можливе створення кластера з трьох одночасно працюючих масивів. В цьому випадку сумарний обсяг дискового простору буде перевищувати 1 Тб. За своїм призначенням встановлюються в RAID жорсткі диски діляться на робочі, контрольні та резервні. К особливостям архітектури відноситься те, що незалежно від того, скільки робочих дисків контрольний диск (т. Е. Диск, що містить контрольні суми) може бути всього один. Це забезпечує максимально плавну наращиваемость системи, так як при додаванні навіть одного диска її ємність збільшується (в RAID інших типів для збільшення ємності зазвичай необхідно підключати відразу цілу групу дисків).

Для комп'ютера RAID 7 виглядає як звичайне SCSI-пристрій, тому до обчислювального устаткування і використовуваному програмному забезпеченню не пред'являється ніяких спеціальних вимог. Не потрібно і наявності додаткових драйверів в застосовуваній операційній системі. Тому дітище Storage Computer

					КВРКІ 170173.17.01.13 ПЗ	Арк. 29
Зм.	Арк.	№ докум.	Підпис	Дата		

працює з усіма комп'ютерними платформами IBM PC, IBM RS / 6000, IBM AS-400, HP, Digital, Sun, Silicon Graphics, Tricord, CRAY і т. Д. І сумісний з усіма операційними системами Windows NT, Novell, SCO Unix, Solaris, SunOS, HP-UX, DigitalUNIX, AIX, DG-UX, OS / 400 і ін.

RAID 7 працює під управлінням власної операційної системи, і його конфігурація здійснюється за допомогою підключається до масиву стандартної клавіатури. Стан пристрою, реальну швидкість передачі даних і багато інших змінюються в реальному масштабі часу характеристики можна бачити на екрані стандартного монітора, який підключається до накопичувача так само просто, як до будь-якого комп'ютера. Цікаво відзначити, що розроблене інженерами Storage Computer пристрій можна використовувати для відбраковування накопичувачів! Справа в тому, що масив, перш ніж почати роботу з новим фізичним диском, піддає його дуже жорсткого тестування.

Крім того, аналогічні по потужності перевірки RAID 7 регулярно влаштовує своїм дискам "за власною ініціативою" і "у вільний від основної роботи час". Веде RAID 7 і ретельну статистику збоїв, які супроводжують роботу будь-якого жорсткого диска і зазвичай ретельно приховуваних від користувача операційними системами "загального призначення". Разом з тим, за словами пана Автоманова, RAID 7 "бореться за життя" встановленого в нього фізичного диска значно довше, ніж звичайний настільний комп'ютер. І лише в крайньому випадку проводить реконструкцію даних на диск з "гарячого резерву". Подальше вдосконалення масивів RAID 7 буде йти по шляху збільшення використовуваного дискового простору і розширення списку підтримуваних SCSI-інтерфейсів (SCSI-3, parallel і serial, Fiber Channel), а також вдосконалення алгоритмів операційної системи SOS.

2.7 Збірка

Після того, як ви збрали все обладнання та запланували конфігурацію, можна розпочати збірку кластера. Це частина роботи кластера: власне складання

					КВРКІ 170173.17.01.13 ПЗ	Арк. 30
Зм.	Арк.	№ докум.	Підпис	Дата		

кластера. Розташуємо свої вузли для хорошого потоку повітря. При прокладанні кабелів обов'язково використовуємо різнокольорові кабелі для внутрішнього та зовнішнього кабелів мереж і позначимо їх на обох кінцях. Це може зайняти багато часу та це зменшить відстеження проблеми, щоб виявити, що її причиною став мінливий мережевий кабель. В ідеалі при запуску всі ваші обчислювальні вузли повинні бути однаковими, як з точки зору внутрішньої конфігурації обладнання та зовнішнього кабелю.

2.8 Технічне обслуговування

Масштабованість кластера вимагає простоти налаштування всіх обчислювальних вузлів з однаковими конфігураціями. Ця можливість корисна в кількох сценаріях: спочатку встановити кластер, перевстановити вузол для обслуговування або для додавання нових вузлів до кластера.

Існує два методи досягнення цієї мети. Перший - виконати повну установку на один вузол, зберегти образ диска та записати цей образ на всі інші вузли. На жаль, це стратегія призводить до втрати підтримки неоднорідності. Якщо ми хочем додати вузли іншої архітектури, ніж та, що вже є в кластері, нам доведеться починати з нуля при встановленні їх.

Інший метод полягає у написанні всіх змін в операційній системі, щоб вони могли бути застосовується під час автоматичної установки. Такі сценарії встановлення зазвичай записують основні налаштування подібних до тих, які було б налаштовано в системі встановлення, список програмних пакетів до встановити поза початковою системою та скрипт для обробки всіх інших модифікацій. Ця установка методу вирішує проблему неоднорідності в тому, що установник обробляє вибір програмного забезпечення, дозволяючи використовувати один і той же сценарій на декількох архітектурах, припускаючи, що всі запити програмного забезпечення пакуються для різних архітектур. Крім того, сценарій, чистої інсталяції є чудовим ресурсом при діагностиці майбутні проблеми або для

					КвРКІ 170173.17.01.13 ПЗ	Арк. 31
Зм.	Арк.	№ докум.	Підпис	Дата		

оновлення операційної системи. Більшість основних дистрибутивів підтримують цей метод. На CentOS це називається kickstart.

На практиці використовується комбінація цих двох методів. Наприклад, для побудови використовуються сценарії кластера, а зображення можна використовувати для заміни несправного жорсткого диска на обчислювальному вузлі.

2.9 Програмне забезпечення

Створюючи кластер, важливо знати, що ми з ним будемо з робити. Ми повинні мати вже деякі дослідження, щоб переконатись, що програмне забезпечення існує для досягнення вашої мети. Наприклад, ми створили високопродуктивний кластер обчислювальної хімії, в якому працює квантова хімія програми та програмне забезпечення для моделювання молекулярної динаміки. Для цього потрібні обчислювальні вузли з багатоядерними центральними процесорами, графічними процесорами, великим об'ємом оперативної пам'яті та значним простором. Важливим фактором є ліцензування прикладного програмного забезпечення, яке працюватиме на кластері. Наприклад, існує безліч програм з обчислювальної хімії з відкритим кодом, для яких ліцензування не є проблемою, наприклад, MOPAC, GAMESS та ORCA. Можна придбати сайт ліцензію на комерційне програмне забезпечення, таке як Gaussian, і використовувати його в кластері. Інші комерційні програми, такі як QChem, вимагають прив'язки до конкретних вузлів, на яких вони будуть запускатись.

2.10 NLB-кластери

При використанні балансування навантаження на кожен з хостів створюється віртуальний сетевий адаптер із незалежним від реального IP та MAC-адресом. Цей віртуальний інтерфейс представляє кластер як єдиний вузол, клієнти звертаються до нього саме за віртуальною адресою. Усі запити отримують кожним вузлом

					КВРКІ 170173.17.01.13 ПЗ	Арк. 32
Зм.	Арк.	№ докум.	Підпис	Дата		

кластера, але обробляються лише одним. На всіх узлах запускається служба балансування мережевого навантаження (Служба збалансування мережевого навантаження), яка, використовуючи спеціальний алгоритм, не вимагає обміну даними між вузлами, приймає рішення, потрібно чи одному або іншому вузлу обробляти запит або не обробляти. Вузли обмінюються серцебиттям-повідомленнями, шукаючи їх доступність. Якщо хост прекрасно видає серцебиття або з'являється новий вузол, інші вузли починають процес збігання (конвергенція), які знову перерозподіляють навантаження. Балансування може бути реалізоване в одному з двох режимів:

1. Одноадресний - одноадресна розсилка, коли замість фізичного MAC використовується MAC віртуального адаптера кластера. У цьому випадку вузли кластера не можуть обмінюватися між собою даними, використовуючи MAC-адресу, лише через IP(або другий адаптер, не пов'язаний з кластером); У межах одного кластера слід використовувати лише один із цих режимів. Можливо налаштувати кілька NLB-кластерів на одному мережевому адаптері, вказав конкретні правила для портів. Такі кластери називають віртуальними. Їх застосування дає змогу задати для кожного додатка, вузла або IP-адресу конкретні комп'ютери у складі первинного кластера або блокування трафіку для кількох додатків, не затрачуючий трафік для інших програм, що виконуються на даному вузлі.

2. Двоадресний NLB-компонент може бути прив'язаним до кількох вузлів мережевим адаптером, що дозволить налаштувати ряд незалежних кластерів на кожному вузлі. Також слід знати, що налаштування кластерів серверів та NLB на одному вузлі неможлива, оскільки вони по-різному працюють із мережевими пристроями. Адміністратор може зробити нову гібридну конфігурацію, що володіє достоїнствами обох методів, наприклад, створений NLB-кластер і настроєна реплікація даних між вузлами. Ніяке повторення не виконується постійно, час від часу, отже, інформація на деяких вузлах буде відрізнятися.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 33
Зм.	Арк.	№ докум.	Підпис	Дата		

Для організації NLB-кластерів додаткове ПО не потрібно, все виробляє вбудованими засобами Win2k3. Для створення, підтримки та моніторингу NLB-кластери використовують компонент «Диспетчер балансування мережевого навантаження» (Менеджер балансування навантаження мережі), який знаходиться у вкладі «Адміністрування» «Панелі управління» (команда NLBMgr). Так як компонент «Балансування навантажень мережі» відображається як стандартний мережевий драйвер Windows, встановлення NLB можна виконати та за допомогою компонента «Мережеві підключення», в якому доступний відповідний пункт. Но краще використовувати тільки перший варіант, одночасне задіявання диспетчера NLB та «Мережевих підключень» може принести до непередбачуваних результатів. Диспетчер NLB дозволяє проводити пошук та управління з одного місця робочої угоди кількох кластерів і вузлів. Можлива також установка NLB-кластера на комп'ютері з одним мережевим адаптером, пов'язаним з компонентом «Балансування навантажень мереж», но в цьому у випадку режиму одноадресного диспетчера NLB на цьому комп'ютері не може бути виконуватись для управління іншими узлами, а самі вузли не можуть змінитися іншою інформацією. Тепер визиваємо диспетчер NLB. Кластерів у нас немає, тому з'явилося вікно не містить жодної інформації. Вибираємо в меню «Кластер» пункт «Новий» і починаємо заповнювати поле в вікні «Параметри кластера». В полі «Налаштувати IP-параметри кластера» ввести значення віртуального IP-адреси кластера, маску підмережі і повне ім'я. Значення віртуального MAC-адреса встановлюється автоматично. Нижче вибираємо режим роботи кластера: одноадресний або багатоадресний. Звернемо увагу на флажок «Дозволити віддалене керування» - всі документи Microsoft рекомендують рекомендувати його не використовувати уникнення проблеми, пов'язаної з безпекою. Замість цього слід застосовувати диспетчер або інші засоби віддаленого управління, наприклад інструментарій управління Windows (WMI). Якщо ви вирішуєте питання про його використання, слід виконати всі надлежащі заходи в захищених мережах, заклавши додатково брандмауером UDP-порти 1717 і 2504. Після заповнення всіх полів нажимаємо

					КВРКІ 170173.17.01.13 ПЗ	Арк. 34
Зм.	Арк.	№ докум.	Підпис	Дата		

«Далі». В вікні «IP-адреса кластера» при необхідності можна додавати додаткові віртуальні IP-адреси, які будуть використовуватись цим кластером. У наступному вікні «Правила для портів» можна задати баланс навантаження для одного або для груп портів усіх або вибраного IP за протоколам UDP або TCP, а також блокувати доступ до кластеру визначеним портом (що міжмережний екран не замінний). Кластер обробляє запити для всіх портів (0–65365). Краще цей список обмежити, щоб в нього входило в його лише дійсно необхідні. Хоча, якщо немає бажання возитися, можна залишити все, як є. Доречі, в Win2k за умовою всього трафіку, направлений до кластера, обробляв лише вузол, який має найвищий пріоритет, інші вузли підключаються лише при виході з строю основного. Наприклад, для IIS потрібно включити лише порти 80 (http) та 443 (https). Приміром можна зробити так, щоб, наприклад, захищені з'єднання обробляли лише певні сервери, на яких встановлений сертифікат. Для додавання нового правила нажимаємо «Додати», що з'явилося в діалоговому вікні IP-адреса використовується, або якщо правило розповсюджується на всіх, щоб залишити флажок «Все». У полях «С» і «По» діапазон портів встановлюється одне і теж значення - 80. Ключовим полем є «Режим фільтрації» (режим фільтрування) – тут задається, кем буде оброблений цей запит. Доступно три поля, що визначають режим фільтрації: «Деякі вузли», «Один вузол» та «Відключити цей діапазон портів». Вибір «Один вузол» означає, що трафік, зроблений на вибраний IP (комп'ютер або кластера) із вказаним номером порту, буде обробляти активним вузлом, який має найменшим показником пріоритету. Вибір «Відключити...» означає, що такий трафік буде відбиратися всім учасникам кластера. У режимі фільтрації «Деякі вузли» можна додатково вказати варіант визначення ходінства клієнтів, щоб зробити трафік із заданого клієнта до одного з того вузла кластера. Можливі три варіанта: «Ні», «Одне» або «Клас C ». Вибір першого означає, що на будь-яку заявку буде відповідати вибраний вузол. Не слід його використовувати, якщо в правилі вибран протокол UDP або «Кілька». При виборі інших пунктів управління клієнтами буде визначено по конкретному IP або

					КВРКІ 170173.17.01.13 ПЗ	Арк. 35
Зм.	Арк.	№ докум.	Підпис	Дата		

діапазону мереж класу С. І так, для нашого правила з 80-метровим портом зупиніть свій вибір на варіантах «Декількох вузлів - клас С». Правило для заповнення аналогічно, що використовується «Один вузол щоб клієнту завжди відповідав основний вузол з найменшим пріоритетом. Якщо диспетчер виявить несумісне правило, буде виведено попереджувальне повідомлення, додатково в журнал подій Windows буде внесена відповідний запис. Далі підключаємося до майбутнього кластера, вводячи його ім'я або реальний IP, і визначаєм інтерфейс, який буде підключено до мережі кластерів. В вікні «Параметри вузла» вибираємо із списку пріоритет, уточнюємо мережеві налаштування, задаємо початковий стан вузла (працює, зупинено, припинено). Пріоритет одночасно є унікальним ідентифікатором узла; чим менше номер, тим вище пріоритет. Вузел з пріоритетом 1 є головним-сервером, в першу чергу отримуючим пакети та діючий як менеджер маршрутизації. Флажок «Зберегти налаштування після перезапуску комп'ютера» дозволяє у випадку збою або перезагрузки цього вузла автоматично ввести його в стрій. Після натискання на «Готово» у вікні Диспетчера з'являється запис про новий кластер, у якому покаже присутній один вузол. Наступний вузол додати також просто. Вибираємо в меню «Додати вузол» або «Підключити до існуючого», залежно від того, з яким комп'ютером виробляє підключення (він уже входить у кластер або ні). Після цього в вікні використовуючи ім'я або адресу комп'ютера, якщо прав для підключення достатньо, новий вузол буде підключений до кластеру. Перший час його значок його ім'я буде відрізнятися, але коли завершиться процес схожень, він буде таким же, як і у вас першого комп'ютера. Так як диспетчер відображає властивості вузлів на момент свого підключення, для уточнення поточного стану вибирає кластер і в контекстному меню пункту «Обновити». Диспетчер підключається до кластеру і показує оновлені дані. Після встановлення NLB-кластер не можна змінити DNS-запис, щоб розпізнавання імені тепер шукало на IP-кластері.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 36
Зм.	Арк.	№ докум.	Підпис	Дата		

2.11 Висновок

В цьому пункті я висловив сіркування, що стосуються проектування кластерного комп'ютера. Починаючи ми описали наші цілі дизайну, життєво важливими є те, що наша мережа масштабована, щоб вмістити будь-яку кількість вузлів без їх встановлення та адміністрування стає непрактичним. Я тоді обговорювали обладнання, яке входить у кластер, включаючи комп'ютери, мережу обладнання, фізичне зберігання, розподіл електроенергії, доступ та запасні частини. Я знайшов, а схема розділення диска для головного вузла та обчислювальних вузлів, що дозволяє легко створювати резервні копії, оновлення та перевстановлення. Описав мережу кластера, включаючи зовнішні підключення до мережі та ізолювану внутрішню мережу. Описав фізичну 8 складання кластера, представив важливість обслуговування та торкнувся питань кластера застосування. Розібрав NBL-кластери.

					КвРКІ 170173.17.01.13 ПЗ	Арк.
						37
Зм.	Арк.	№ докум.	Підпис	Дата		

3 ПРОГРАМНО АПАРАТНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ

3.1 Вибір комплектуючих для побудови кластера

Переред тим як почати побудову та налаштування кластерної мережі нам потрібно обрати компоненти комп'ютерів які ми будемо використовувати та налаштовувати. Для початку оберемо процесор на базі якого ми будемо створювати комп'ютери які вподальшому об'єднаємо в одну кластерну мережу. на вибір нам підпадають процесори двох виробників таких як Intel та AMD. Для того щоб полегшити подальше налаштування системи ми виберемо процесори одного сімейства, однієї лінійки від єдиного виробника. Процесори Intel краще виконують обчислення на одне ядро, а представлення від AMD краще використовують розподілення задачі на декілька ядер та потоків. Обрахувавши за та проти проти було вирішено використовувати процесори від Intel, адже в нашому кластері важливишим аспектом буде обчислювальна потужність кожного з ядер, які об'єднуються в єдну мережу під управління одного комп'ютера який знизить недоліки роботи процесора Intel. Терпер ми можемо вибрати саму модель процесора, материнську плату, пам'ять, диск на якому буде зберігатися інформація, та корпус в якому розташуємо дані комп'ютери. Все що ми тут оберемо розповсюджується на всі комп'ютери мережі адже це збільшить сумісність системи і головне що цезбільшить надійність та можливість ремонту одного з вузлів кластера використовуючи запчастини одного з робочого вузлів. Для початку в системі ми використаємо процесор Процесор Intel® Xeon® E-2288G який за досить малу ціну надасть нам високу продуктивність (його технічні характеристики вказані в таблиці 3.1, зовнішній виглфд на рисунку 3.1).

					КвРКІ 170173.17.01.13 ПЗ	Арк.
						38
Зм.	Арк.	№ докум.	Підпис	Дата		

Таблиця 3.1 - Технічні характеристики процесора Intel® Xeon® E-2288G

Колекція продукції	Процесор Intel® Xeon® E-2288G
Кодова назва	Coffee Lake
Вертикальний сегмент	Server
Номер процесора	E-2288G
Стан	Launched
Дата випуску	Q2'19
Літографія	14 nm
Рекомендована ціна	\$ 539.00
Кількість ядер	8
Кількість потоків	16
Базова тактова частота	3,70 GHz
Максимальна тактова частота	5,00 GHz
Максимальна частота з Turbo Boost	5,00 GHz
Кеш-пам'ять	16 MB Intel® Smart Cache
Розрахункова потужність	95 W
Сокет	1151
Тип пам'яті	DDR4-2666
Ціна	13 744 грн



Рисунок 3.1- Зовнішній вигляд процесора Intel® Xeon® E-2288G

Далі нам потрібно обрати під даний процесор материнську плату. Найкраще розкриє даний процесор материнська плата сімейства Z чіпсета. Вибір впав на рішення від Gigabyte у вигляді материнської плати Gigabyte Z390 D. Через не велику ціну та добре продуману систему охолодження ланцюгів електрохарчування материнської плати та північного моста це один з найкращих варіантів(технічні характеристики материнської плати Gigabyte Z390 D вказані в таблиці 3.2, зовнішній вигляд на рисунку 3.2) .

Таблиця 3.2 - Технічні характеристики материнської плати Gigabyte Z390 D.

Тип	Материнська плата Intel
Роз'єм процесора (Socket)	LGA1151-V2
Процесори	Процесори LGA1151 (Coffee Lake)
Чіпсет (північний міст)	Intel Z390
Тип пам'яті	DDR4 DIMM
Сумісність ОЗУ	DDR4 для ПК
Кількість слотів	4 штуки
Кількість каналів	2
Максимальний об'єм пам'яті	128 Гігабайт

Продовження таблиці. 3.2 - Технічні характеристики материнської плати Gigabyte Z390 D.

Мінімальна частота	2133
Максимальна частота	4266
Мережевий адаптер (LAN)	1000 Мб/с
Кількість SATA 3	6 штуки
Кількість PCI-E 1x	3 штуки
Кількість PCI-E 16x	3 штуки
Кількість вбудованих портів USB 2.0	2 штуки
Кількість вбудованих портів USB 3.1	1 штука
Кількість слотів M.2.	1 штука
Роз'єм PS/2	1 штука
Кількість зовнішніх портів USB 3.1	6 штук
Роз'єм HDMI	1 штука
Ціна	3000 грн



Рисунок 3.2 - Зовнішній вигляд материнської плати Gigabyte Z390 D

					КВРКІ 170173.17.01.13 ПЗ	Арк. 41
Зм.	Арк.	№ докум.	Підпис	Дата		

Наступним кроком буде вибір оперативної пам'яті. Головним критерієм у виборі пам'яті виступає тип підтримуваної пам'яті процесором та материнською платою та частота на якій буде безперебійно працювати пам'ять на даній конфігурації системи. Найкраще підходить сюди кіт оперативної пам'яті 2*8 HyperX FURY DDR 4. Данна пам'ять може працювати у всіх частотах у діапазоні від 2400 до 3733, та радіатор охолодження мінімізує перегрів пам'яті, що зменшить ризик виходу з ладу(технічні характеристики вказані в таблиці 3.1.3, зовнішній вигляд на рисунку 3.1.3).

Таблиця 3.3 – Технічні характеристики пам'яті HyperX FURY DDR 4

Форм фактор	UDIMM
Підтримка XMP	Так
Авто налаштування конфігуратора Plug N Play	Так
Швидкість	2400-3733
CAS - латентність	15,16,17,18,19
Напруга	1,2 В, 1,35 В
Ємкість модулів	2*8гб
Висота	34,1 mm.
Ціна	2650 грн



Рисунок 3.4 –Зовнішній вигляд WD Green 2tb

Для роботи без проблем знадобиться якісне живлення кожного елемента мережі кластера. Для цього блок живлення має видавати потужність не менше 600 ват на один комп'ютер мережі, та мати сертифікат мінімально 80 Plus Bronze, що являється символом гарантії того що блок витримає навантаження в мережі електроенергії та забезпечить електрикою у достатній кількості всі частини комп'ютера. Вибір впав на блоки живлення від компанії CHIEFTEC моделі Proton 600W. Дана компанія давно себе зарекомендувала як надійний виробник блоків живлення використовуюючи мінімум перемичок, та якісні тайванські і японські конденсатори (технічні характеристики вказані в таблиці 4. 5, зовнішній вигляд на рисунку 4.5).

					КВРКІ 170173.17.01.13 ПЗ	Арк.
						44
Зм.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.5 – Технічні характеристики блоку живлення Chieftec Proton 600

Форм-фактор	ATX
Потужність	600 Вт.
Вентилятор	120 мм.
Роз'єм для живлення відеокарти	6+2 pin, 2 штуки
Роз'єм для живлення процесора	4 pin, 8 pin
Роз'єм для живлення материнської плати	24 pin
КПД (сертифікат 80 Plus)	Bronze
Корекція коефіцієнта потужності (PFC)	активний
+5V	18 А.
+3.3V	18 А.
+12V1	50 А.
-12V	0.3 А.
+5Vsb	2.5 А.
Ціна	1560 грн.



Рисунок 4.5 - Зовнішній вигляд блоку живлення Chieftec Proton 600W

Тепер можемо приступити до останнього пункту, вибору місця в якому буде знаходитись даний комп'ютер, тобто до корпусу. Головним критерієм при виборі корпусу виступає практичність, скільки портів ми маємо у вільному доступі на передній панелі. Ми використаєм корпус Gamemax моделі ET-209-NP (технічні характеристики вказані в таблиці 3.6, зовнішній вигляд зображено на рисунку 3.6).

Таблиця 3.6 – Технічні характеристики корпусу Gamemax ET-209-NP

Форм-фактор	ATX
Форм-фактор материнської плати	ATX, Mini-ITX, microATX
Кількість відсіків 5.25"	1
Кількість відсіків 3.5"	3
Кількість відсіків 2.5"	2
Охолодження	Задня панель 1 x 80 мм.
Додатково	2 x USB2.0, 1 аудіо вихід, 1 вихід на мікрофон
Розміри	175 x 405 x 330 мм.
Ціна	520 грн.

Середня ціна комп'ютера в кластері складає 22674 грн без отоволокна. Для мережі нам знадобиться мінімум 9 комп'ютера, отже фінальна ціна мережі складає 204 066 гривень.

3.2 Побудова мережі

Наступним кроком створення нашої кластерної мережі являється її фізична побудова та налаштування. По завданню нам потрібно зв'язати пт між собою у вигляді решітки тобто один ПК з'єднується з двома сусідніми і в центрі даної решітки буде знаходитись головний керуючий комп'ютер. З'єднання не впливає на швидкість роботи всього кластера адже навантаження на вузли кластера розподіляє головний комп'ютер. Для початку ми повинні встановити програмне забезпечення на всі комп'ютери кластер, в даному випадку ми використовуємо комп'ютери на базі операційної системи Windows тому серйовищем для налаштування кластера ми оберемо Windows server 2016. Microsoft Windows Server 2016, раніше називався Windows Server vNext, є серверна операційна система(ОС). Серверна операційна система спеціально розроблена, щоб служити платформою для запуску мережевих додатків.

Дуже очікуваною особливістю Windows Server 2016 є включення двох рідних контейнерів- контейнерів Windows Server і Hyper-V контейнерів. Контейнери Windows Server працюватимуть безпосередньо в ОС, але ізольовані один від одного. Hyper-V Контейнери забезпечують посилену ізоляцію і запуск від віртуальної машини Microsoft Hyper-V (VM). Windows Server 2016 також надаватиме вбудовану підтримку Docker, яка може використовуватися для керування контейнерами обох типів.

Windows Server 2016 також замінить знімки Hyper-V виробничими контрольними точками, які дозволяють адміністраторам відкотитись до більш ранньої точки часу без відновлення з резервної копії. На відміну від знімків, контрольні точки використовують служби точного копіювання томів. Нова версія

					КВРКІ 170173.17.01.13 ПЗ	Арк. 48
Зм.	Арк.	№ докум.	Підпис	Дата		

Hyper-V у Windows Server 2016 буде підтримувати віртуальний модуль надійної платформи, який дозволяє шифрувати віртуальну машину за допомогою BitLocker. Інші нові та оновлені функції Windows Server 2016 включають:

1. Nano Server- легкий варіант установки, який призначений для роботи виключно в хмарних і контейнерних сценаріях і на 93% менший, ніж традиційні налаштування Windows Server. Nano Server не включає графічний інтерфейс або традиційний фреймворк .NET; адміністратори будуть керувати Nano екземпляри сервера віддаленого за допомогою PowerShell і Windows інструментарій управління.

2. Мережевий контролер— нова серверна роль, яка забезпечує централізовану програмувану точку автоматизації для настроювання, керування, моніторингу та виправлення неполадок пристроїв і служб віртуальної та фізичної мережі. Мережевий контролер може бути використаний з продуктами системного центру, таких як менеджер віртуальних машин та диспетчер операцій, а також PowerShell.

3. Якість зберігання послуг (Storage QoS),що забезпечує можливість централізованого моніторингу та управління продуктивністю сховища для віртуальних машин і автоматично покращує справедливість ресурсів зберігання між декількома віртуальними машинами, використовуючи один і той же кластер файлового сервера.

4. Репліка сховища— нова функція, яка забезпечує синхронну реплікацію на рівні сховища між серверами або кластерами для аварійної готовності та відновлення.

5. Зберігання Spaces Direct, що дозволяє високо доступні та масштабовані програмно визначені сховища для приватної хмари. Storage Spaces Direct має два режими розгортання: приватне хмарне сховище, де воно відокремлене від гіпервізора, і приватна хмарна гіпер-сходи, де Storage Spaces Direct і гіпервізор знаходяться на тих же серверах.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 49
Зм.	Арк.	№ докум.	Підпис	Дата		

6. Прокат оновлень для Hyper-V і масштабу кластерів файлових серверів. Ця функція дозволяє користувачам додавати вузол Windows Server 2016 до кластера Hyper-V з вузлами під керуванням Windows Server 2012 R2.

7. Гарячі додати і видалити віртуальну пам'ять і віртуальних мережевих адаптерів під час роботи віртуальної машини.

8. Just Enough Administration, набір інструментів PowerShell, який обмежує користувачів виконувати лише завдання, для яких вони дозволені як частина своєї ролі.

Перший технічний попередній перегляд Windows Server 2016 був випущений 1 жовтня 2014 року, а другий попередній перегляд був випущений 4 травня 2015 року. Третій технічний попередній перегляд був випущений 19 серпня 2015 року і включав перший загальнодоступний попередній перегляд для контейнерів Windows Server Containers, а також консоль аварійного керування, яка дозволяє користувачам переглядати та виправляти конфігурації мереж безпосередньо з консолі Nano Server. Наступний попередній перегляд включатиме Hyper-V Контейнери. На відміну від попередніх версій, Windows Server 2016 не був запланований до випуску одночасно з Windows 10 - останньою основною версією клієнтської ОС Microsoft. Реліз запланований на 2016 рік. Технічні вимоги до Windows Server 2016 - це процесор із тактовою частотою 1,4 ГГц, 512 МБ пам'яті, 32 ГБ дискового простору та адаптер Ethernet обсягом 1 гігабіт. Отже дана програма повністю підходить під наші задачі

Після встановлення програмного забезпечення ми пристосовуємо до підключення комп'ютерів один до одного як зображено логічній схемі 3.6 Головний керуючий комп'ютер ми розташовуємо в центрі нашої мережі до нього ми підключимо девайси вводу та виводу інформації (монітор, мишка, клавіатура, принтер сканер), а до інших вузлів підключиння пристроїв вводу виводу інформаціїї адже вони будуть слугувати тільки як вузлами обробки інформації, тобто машинами на яких тільки обробляється інформація яка надходить з

					КВРКІ 170173.17.01.13 ПЗ	Арк. 50
Зм.	Арк.	№ докум.	Підпис	Дата		

головного комп'ютера, і результат цієї обробки виводиться на головний комп'ютер даної кластерної мережі.

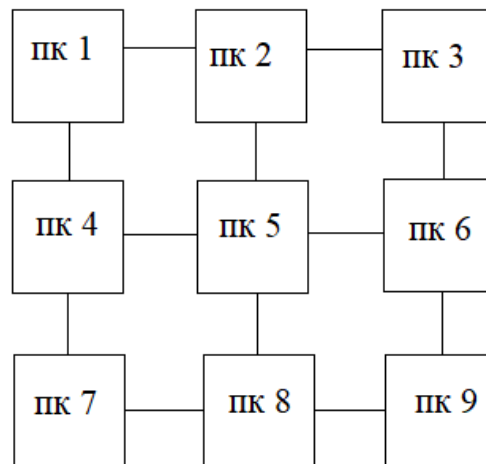


Рисунок 3.6 – Логічна схема комп'ютерної кластерної мережі

Після того як ми з'єднали комп'ютери можна приступати до налаштування кластерної мережі в середовищі Windows server 2016. Розпочнімо з встановлення функції відмови кластеризації. Потрібно встановити функцію відмовостійкої кластеризації на кожному сервері, який слід додати як вузол відмовостійкого кластера:

1. Крок 1: Запускаємо диспетчер серверів.
2. Крок 2: Вибераємо Додати ролі та функції з меню Керування.
3. Крок 3: Натискаємо Далі на сторінці Перед початком роботи.
4. Крок 4: На сторінці Вибір типу інсталяції оберемо Рольове або Функціональне встановлення та натискаємо Далі.
5. Крок 5: На сторінці Вибір цільового сервера вибираємо сервер, на який ми хочемо встановити функцію, а потім натискаємо Далі.
6. Крок 6: Натискаємо Далі на сторінці Вибір ролей сервера.
7. Крок 7: На сторінці Вибір функцій встановлюємо прапорець Кластеризація відмови.

8. Крок 8: Щоб встановити інструменти управління відмовостійким кластером, вибираємо Додати функції та натискаємо Далі.

9. Крок 9: На сторінці Підтвердження вибору встановлення натискаємо кнопку Встановити. (Для функції відмови кластеризації перезавантаження сервера не потрібно).

10. Крок 10: Після завершення встановлення натискаємо кнопку Закрити.

11. Крок 11: Повторяємо цю процедуру на кожному комп'ютері, який ви хочете додати як вузол відмовостійкого кластера.

Тепер ми переходимо до створення відмовостійкого кластера за допомогою оснастки диспетчера відмов кластера. Перш ніж розпочати, підключимо сховище iSCSI до всіх вузлів і переконаємось, що всі віртуальні комутатори у вузлах ідентичні:

1. Крок 1: Вибираємо менеджер відмов кластера в меню Інструменти диспетчера серверів.

2. Крок 2: У диспетчері відмов кластера вибираємо опцію Створити кластер на панелі керування.

3. Крок 3: Відкриється майстер створення кластера; На сторінці Перед початком роботи вибираємо Далі (Рисунок 3.7).

					КВРКІ 170173.17.01.13 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52

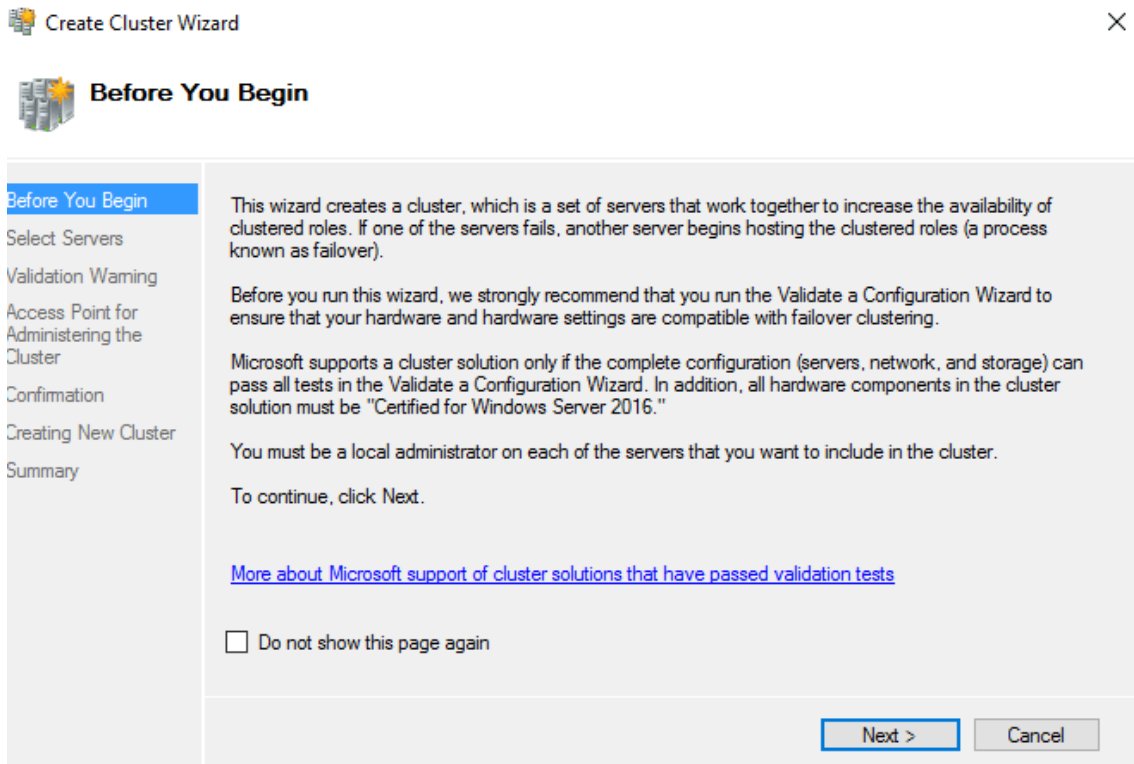


Рисунок 3.7 - Майстер створення кластера

4. Крок 4: Тепер переглянамо та вибираємо вузли та натискаємо Далі (рисунок 3.8).

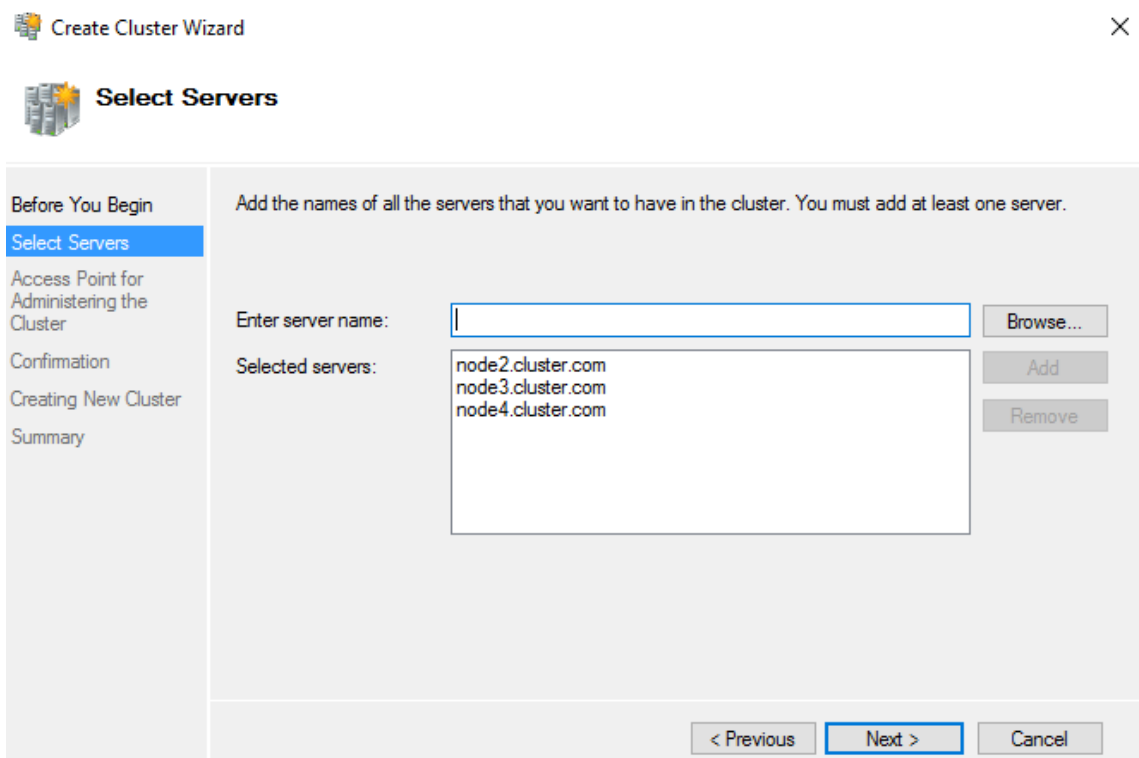


Рисунок 3.8 – Майстер створення кластера

7. Крок 7: На сторінці Підсумок потрібно переконайтись, що відмовостійкий кластер успішно створений. Якщо були застереження або помилки, потрібно переглянути підсумковий результат або вибрати Переглянути звіт, щоб переглянути повний звіт. Вибераємо Готово(рисунок 3.11).

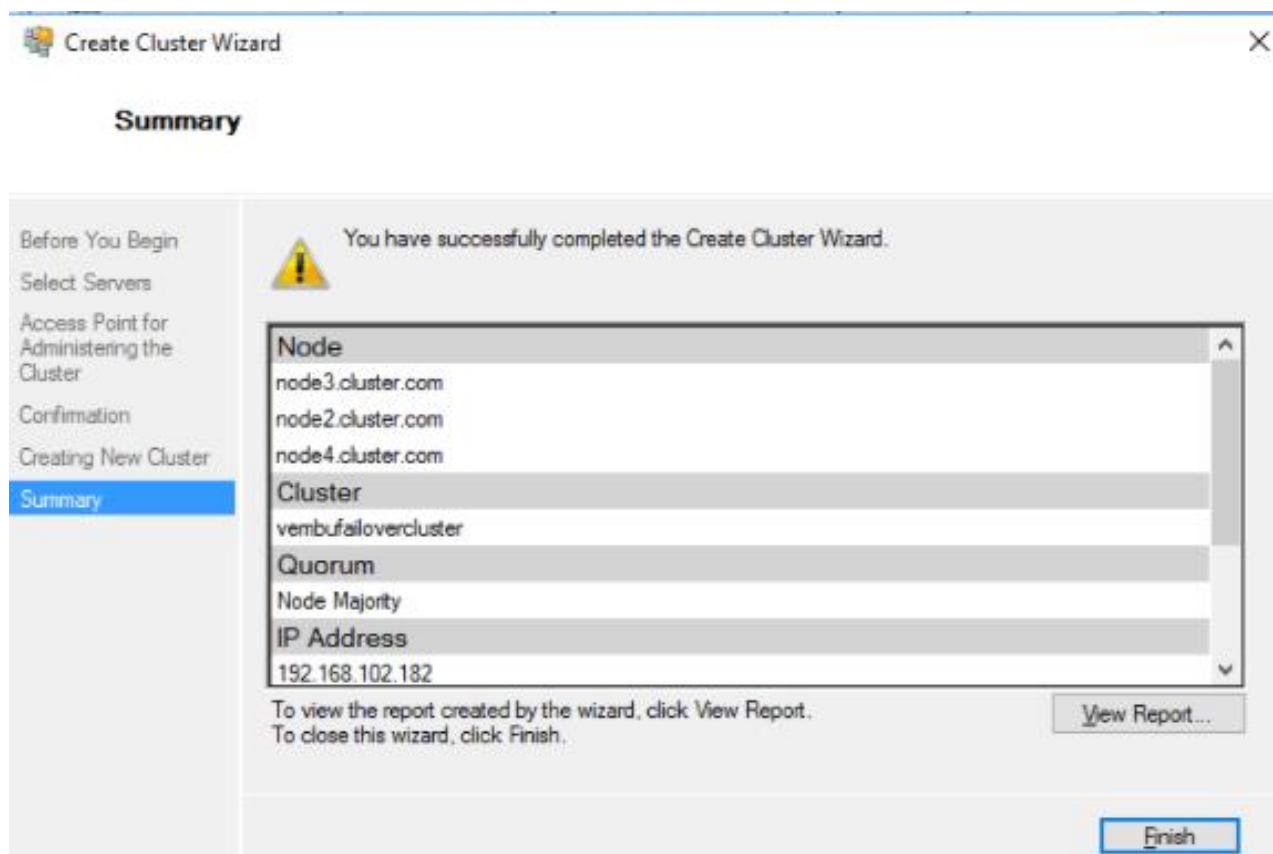


Рисунок 3.11 - Майстер створення кластера

8. Крок 8: Тепер відкрийте Failover Cluster Manager, розширюємо сховище та натискаємо Диски.

9. Крок 9: Клацнімо на Додати диск і вибираємо доступне сховище iSCSI (рисунок 3.12).

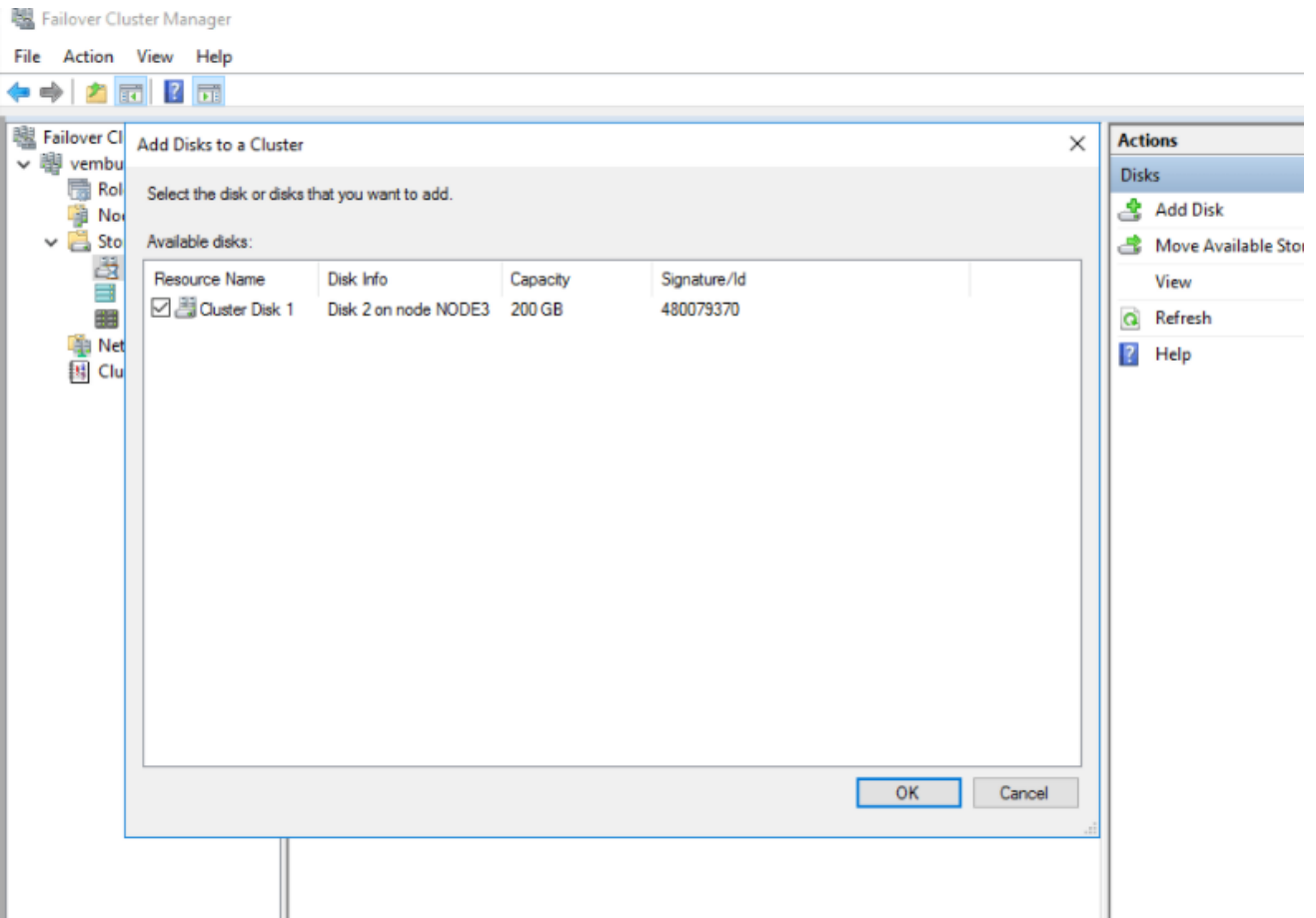


Рисунок 3.12 - Failover Cluster Manager

10. Крок 10: Потім клацнімо правою кнопкою миші на кластерному диску та виберемо Додати до сховища кластера (рисунок 3.13).

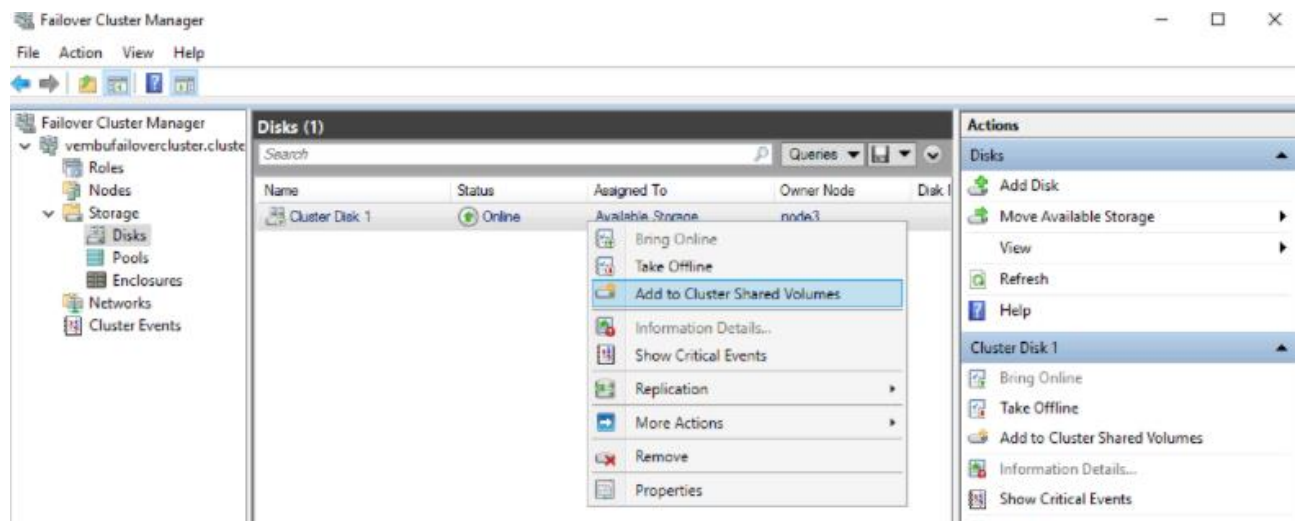


Рисунок 3.13 - Failover Cluster Manager

11. Крок 11: ми також можемо перевірити пам'ять, відкривши C: \ ClusterStorage у всіх вузлах.

Тепер все готово і наша мережа прийняли свій вже фінальний вигляд який зображений на фізичній схемі (рисунок 3.14).

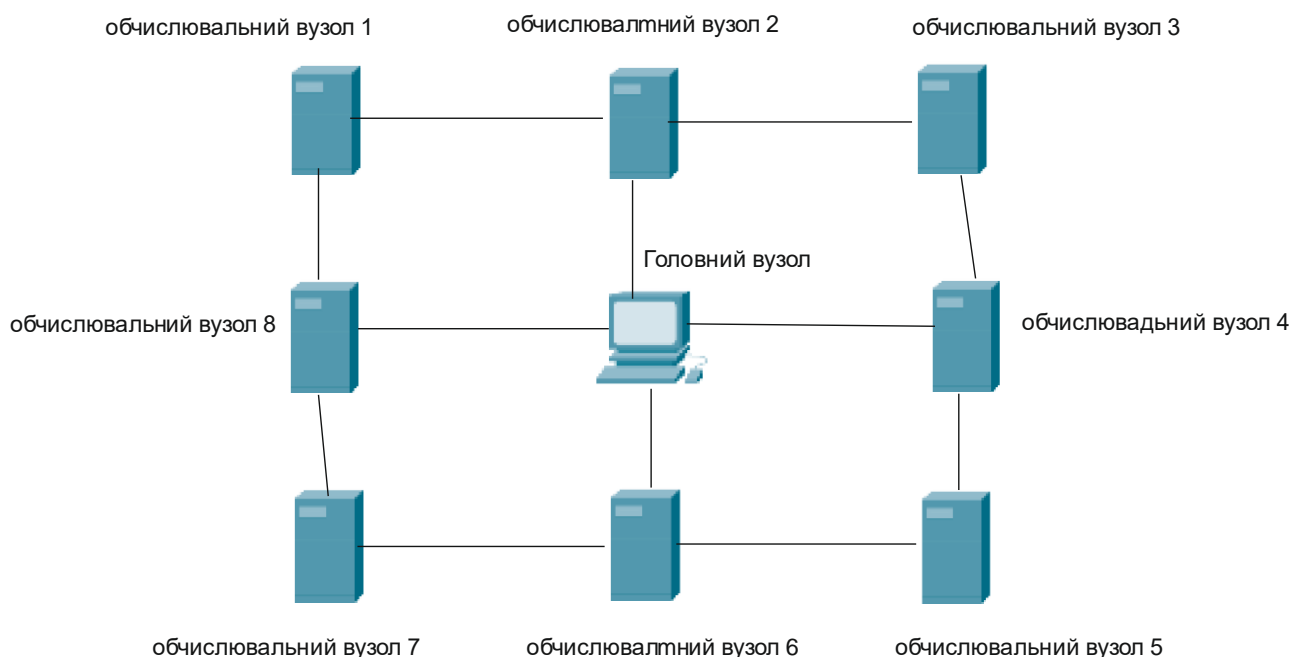


Рисунок 3.14 – Фізична схема кластерної системи

Також ми можна налаштувати кластерну мережу за допомогою Windows Power Shell. Наведені нижче командлети Windows PowerShell виконують ті самі функції, що й при створенні кластера відмовостійкого використання за допомогою оснастки Failover Cluster Manager:

1. Крок 1: Відкрийте редактор Windows Powershell.
2. Крок 2: Встановіть функцію відмовостійкого кластера Windows разом із інструментами управління.
3. cmd: `Install-WindowsFeature -Ім'я відмовостійкого кластеризації - IncludeManagementTools.`

4. Крок 3: Наведений нижче cmd запускає всі тести перевірки кластера на комп'ютерах з іменами Server1 та Server2.

5. Cmd: Test-Cluster –Node Server1, Server2.

6. Примітка: Командлет Test-Cluster видає результати у файл журналу в поточному робочому каталозі. Наприклад: C: \ Users \ AppData \ Local \ Temp.

7. Крок 4: Створіть відмовостійкий кластер з назвою MyCluster із вузлами Server1 та Server2 та призначте статичну IP-адресу 192.168.10.182 та перенесіть усі відповідні дані в відмовостійкий кластер.

8. Cmd: New-Cluster –Name MyCluster –Node Server1, Server2 –StaticAddress 192.168.10.182.

9. Крок 5: Створіть відмовостійкий кластер з іменем MyCluster в OU кластера домену testvembu.com. Cmd: New-Cluster -Name CN = MyCluster, OU = Cluster, DC = testvembu, DC = com -Node Server1, Server2. Ось і все, кластер відмови створюється зі статичним IP, сховищем та розміщується в домені testvembu.com.

Нові функції відмовостійких кластерів Windows Server 2016:

1. Постійне оновлення кластерної операційної системи.

2. Завдяки такій новій функції, як оновлення Cluster OS, адміністратор може оновити операційну систему кластерних вузлів з Windows Server 2012 R2 до Windows Server 2016, не зупиняючи служби Hyper-V. Ця нова функція допомагає уникнути штрафних санкцій за простої згідно з Угодами про рівень обслуговування (SLA).

3. Хмарний свідок відмови кластера.

4. Новий тип свідчення кворуму, який використовує Microsoft Azure для визначення того, який вузол кластера повинен бути повноважним, якщо вузол переходить у автономний режим.

5. Служба моніторингу здоров'я системи.

6. Покращує повсякденний моніторинг, експлуатацію та досвід обслуговування кластерів Storage Spaces Direct.

Домени помилок:

					КВРКІ 170173.17.01.13 ПЗ	Арк. 58
Зм.	Арк.	№ докум.	Підпис	Дата		

1. Допомагає визначити, який домен помилки використовувати з кластером Storage Spaces Direct. Домен несправності - це набір апаратних засобів, що розділяє одну точку відмов, наприклад, вузол сервера, шасі сервера або стійку.

2. VM балансування навантаження.

3. Навантаження розподіляється між вузлами рівномірно з його допомогою, у відмовостійкому кластері, виявляючи зайняті вузли та переносючи віртуальні машини на цих вузлах до менш зайнятих.

4. Спрощені багатоканальні та багатосетеві кластерні мережі SMB:

5. Дозволяє користувачеві легко налаштувати кілька мережевих адаптерів у кластері.

6. Кластери робочих груп та мультидоменів у Windows Server 2016.

7. Windows Server 2016 руйнує попередній бар'єр створення кластера між вузлами-членами, приєднаними до того самого домену, і вводить можливість створювати відмовостійкий кластер без залежностей Active Directory. Наступна конфігурація, реалізована у відмовостійких кластерах, тепер може бути створена в:

8. Однодоменний: кластери з усіма вузлами, приєднаними до одного домену.

9. Багатодомен: кластери з вузлами, які є членами різних доменів.

10. Робоча група: Кластери з вузлами, які є серверами-членами / робочою групою (не приєднані до домену).

Отже нові та вдосконалені функції відмовостійкого кластера Windows Server 2016 зробили концепцію кластеризації та високої доступності простішою, ніж будь-коли. Таким чином, ми рекомендуємо оновити попередні версії відмовостійкого кластера до нового відмовного кластера Windows Server 2016, де ви зможете скористатися перевагами всіх нових функцій.

ВИСНОВКИ

В ході дослідження можливостей прискорити обробку інформації та збільшити надійність системи зберігання інформації, було досліджено можливість з'єднання декількох комп'ютерів в одну систему, які модулюють роботу одного супер комп'ютера на декількох комп'ютерах. Дана можливість називається кластеризацією. Саме завдяки кластеризації комп'ютерів з'явилась можливість без додаткових витрат маючи декілька машин створити одну потужну машину яка прискорює час, швидкість обробки інформації, та також збільшує захист важливої інформації на всіх носіях, адже вона розподіляється на декілька сховищ тим самим зменшуючи ризик втрати даних через вихід зладу одного з вузлів кластерної мережі. Після дослідження ми приступили до фізичного представлення мережі де ми за допомогою Windows server 2016 налаштували комп'ютери на роботу в кластері, а саме з дев'яти представлених комп'ютерів ми вибрали один який являється головним вузлом і сім інших, які в свою чергу виступають обчислювальними вузлами. Всі маніпуляції які проводяться з нашою кластерною системою задіяні від головного комп'ютера, це задання завдання, считування, введення та виведення інформації, що в свою чергу присорило та збільшило надійність як системи так і системи збереження та захисту інформації.

Це прискорило роботу у нашій установі у вісім раз що дозволило виконувати більш складніші та затратні проекти за менший час, що в свою чергу збільшило продуктивність самої компанії.

					КвРКІ 170173.17.01.13 ПЗ	Арк.
						60
Зм.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Ілля Євсєєв. Використання PVM. Введення в програмування.http://www.csa.ru/~il/pvm_tutor/.
2. Al Geist, Adam Beguelin, Jack Dongarra, Weicheng Jiang, Robert Manchek, Vaidy Sunderam. PVM: Parallel Virtual Machine. A Users 'Guide and Tutorial for Networked. Parallel Computing. <http://www.netlib.org/pvm3/book/pvm-book.html> .
3. Євген Борисов. Обчислювальні системи надвисокої продуктивності.http://mechanoid.narod.ru/high_perf/index.html.
4. Дмитро Арапов. Чи можна перетворити мережу в суперкомп'ютер?<http://www.osp.ru/os/1997/04/6.html>.
5. Павлов В.М. Програмне забезпечення для побудови систем розподілених обчислень. Реферат по міждисциплінарного державного іспиту.
6. Костянтин Михайленко. Паралельний стиль. <http://www.altlinux.ru>.
7. Михайло Кузьмінський. Кластери на базі ОС Linux. *ComputerWorld* 5/1998.
8. А.М. Горелик. Засоби підтримки паралельності в мовах програмування. *Відкриті системи* 02/1995.
9. Андрій Кузнєцов. Паралельні світи.http://ccc.ru/magazine/depot/00_08/web3.htm10.
10. Вл. В. Воєводін. Суперкомп'ютерна грань комп'ютерного світу. <http://parallel.ru/vvv/intro2hpc.html>.
11. Е.С.Борісов. Декомпозиція послідовних програм при паралельних обчисленнях. <http://mechanoid.narod.ru/parallel/semiautomata/>.
12. В.Н. Дацюк, А.А. Букатов, А.І. Жегулев. Методичний посібник з курсу "Багатопроцесорні системи та паралельне програмування".
13. Кільпіо, П. Кайгородов. Паралельні системи, їх створення і застосування. <http://www.xform.ru/library/article/default.asp?I=20>.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 61
Зм.	Арк.	№ докум.	Підпис	Дата		

27. Ершов, К. С. Анализ и классификация алгоритмов кластеризации. // *Новые информационные технологии в автоматизированных системах*. 2016. No19. С. 274-279.

28. Бериков, В. С. Современные тенденции в кластерном анализе / В. С. Бериков, Г. С. Лбов. Всероссийский конкурсный отбор обзорноаналитических статей по приоритетному направлению "Информационно-телекоммуникационные системы", 2008. 26 с.

29. Duran, B. S. Cluster Analysis A Survey / B. S. Duran, P. L. Odell. Springer 1974.

30. Мандель, И. Д. Кластерный анализ // *Финансы и статистика* 1988. 176 с.

31. Lutz, M. Programming Python // *O'Reilly Media* 1996.

32. Blanco Silva, F. J. Learning SciPy for Numerical and Scientific Computing . Packt publishing, 2015.

33. Müller, A. C. Introduction to Machine Learning with Python: A Guide for Data Scientists O'Reilly Media 2016.

34. Paul S. Bradley, Usama M. Frayyad — Refining Initial Points for KMeans Clustering: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.50.8528&rep=rep1&type=pdf>.

35. Jasper Snoek, Hugo Larochelle, Ryan P. Adams Practical Bayesian Optimization of Machine Learning Algorithms [Электронный ресурс]. Режим доступа: <http://papers.nips.cc/paper/4522-practical-bayesian-optimization-of-machine-learning-algorithms.pdf>.

36. Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrodl Constrained K-means Clustering with Background Knowledge <https://web.cse.msu.edu/~cse802/notes/ConstrainedKmeans.pdf>.

37. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel. *Scikit-learn: Machine Learning in Python* <http://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>38.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 63
Зм.	Арк.	№ докум.	Підпис	Дата		

38. Sayak Paul K-Means Clusteringin Python with scikit-learn 2018.
<https://www.datacamp.com/community/tutorials/k-means-clusteringpython>.
39. Swaroop C H. A Byte of Python2013.
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>.
40. Alex Smola and S.V.N. Vishwanathan — INTRODUCTION TO MACHINE LEARNING [Электронный ресурс]. 2008. Режим доступа: <http://alex.smola.org/drafts/thebook.pdf>.
41. Andrea Trevino — Introduction to K-means Clustering [Электронный ресурс]. 2016. – Режим доступа: <https://www.datascience.com/blog/k-means-clustering>.
42. AurélienGéron — Hands-On Machine Learning with Scikit-LearnandTensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems 2015.
https://www.academia.edu/37010160/HandsOn_Machine_Learning_with_ScikitLearn_and_TensorFlow.
43. A.K.Jain,M.N.Murty,P.J.Flynn—Data Clustering:A Review
<http://www.csee.umbc.edu/nicholas/clustering/p264-jain.pdf>.
44. Суслов, С. А. Кластерный анализ: сущность, преимущества и недостатки. — 2011.с. 51-56.
45. Ершов, К. С. Анализ и классификация алгоритмов кластеризации. *Новые информационные технологии в автоматизированных системах*.2016. №19. — С. 274-27943.
46. Бериков, В. С. Современные тенденции в кластерном анализе / В.С.Бериков, Г. С. Лбов. — Всероссийский конкурсный отбор обзорноаналитических статей по приоритетному направлению "Информационно-телекоммуникационные системы", —2008. —26 с.
47. Duran, B. S. Cluster Analysis—A Survey / B. S. Duran, P. L. Odell.Springer— 1974. Computing . —Packt publishing, 2015.
48. Müller, A. C. Introduction to Machine Learning with Python: A *Guide for Data Scientists O'Reilly Media*.2016.

					КВРКІ 170173.17.01.13 ПЗ	Арк. 64
Зм.	Арк.	№ докум.	Підпис	Дата		

49. Paul S. Bradley, Usama M. Frayyad Refining Initial Points for KMeans Clustering:<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.50.8528&rep=rep1&type=pdf>.

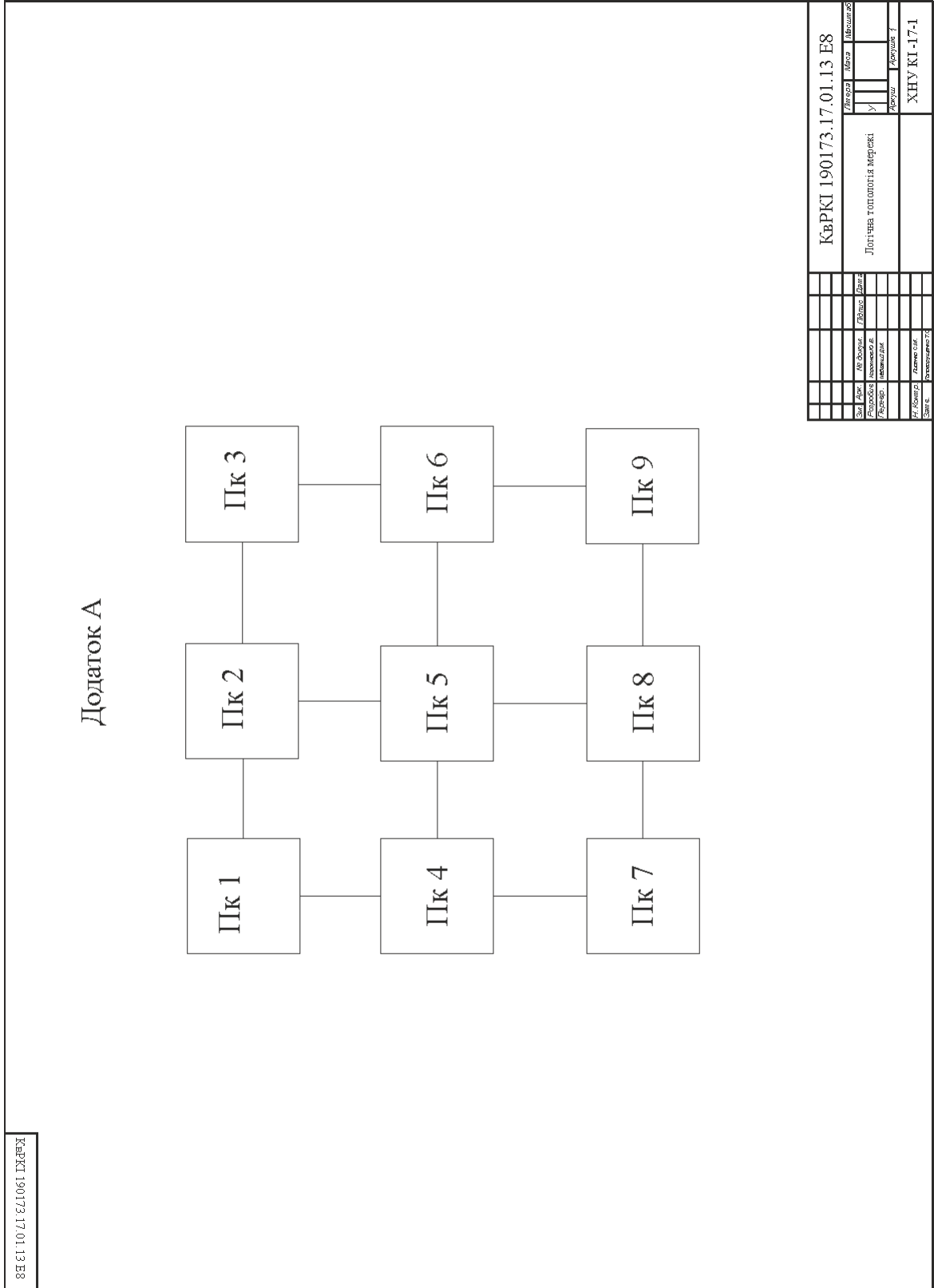
50. Dogra D., Ahmed A. & Bhaskar H. Smart video summarization using Mealy machine-based trajectory modelling for surveillance applications. *Multimed Tools Appl.* 2016. Vol. 75. Pp. 6373–6401.

51. Klimovich A.S., Solov'ev V.V. Minimization of Mealy finite-state machines by internal states gluing. *J. Comput. Syst. Sci. Int.* 2012. Vol. 51. Pp. 244–255.

					КВРКІ 170173.17.01.13 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		65

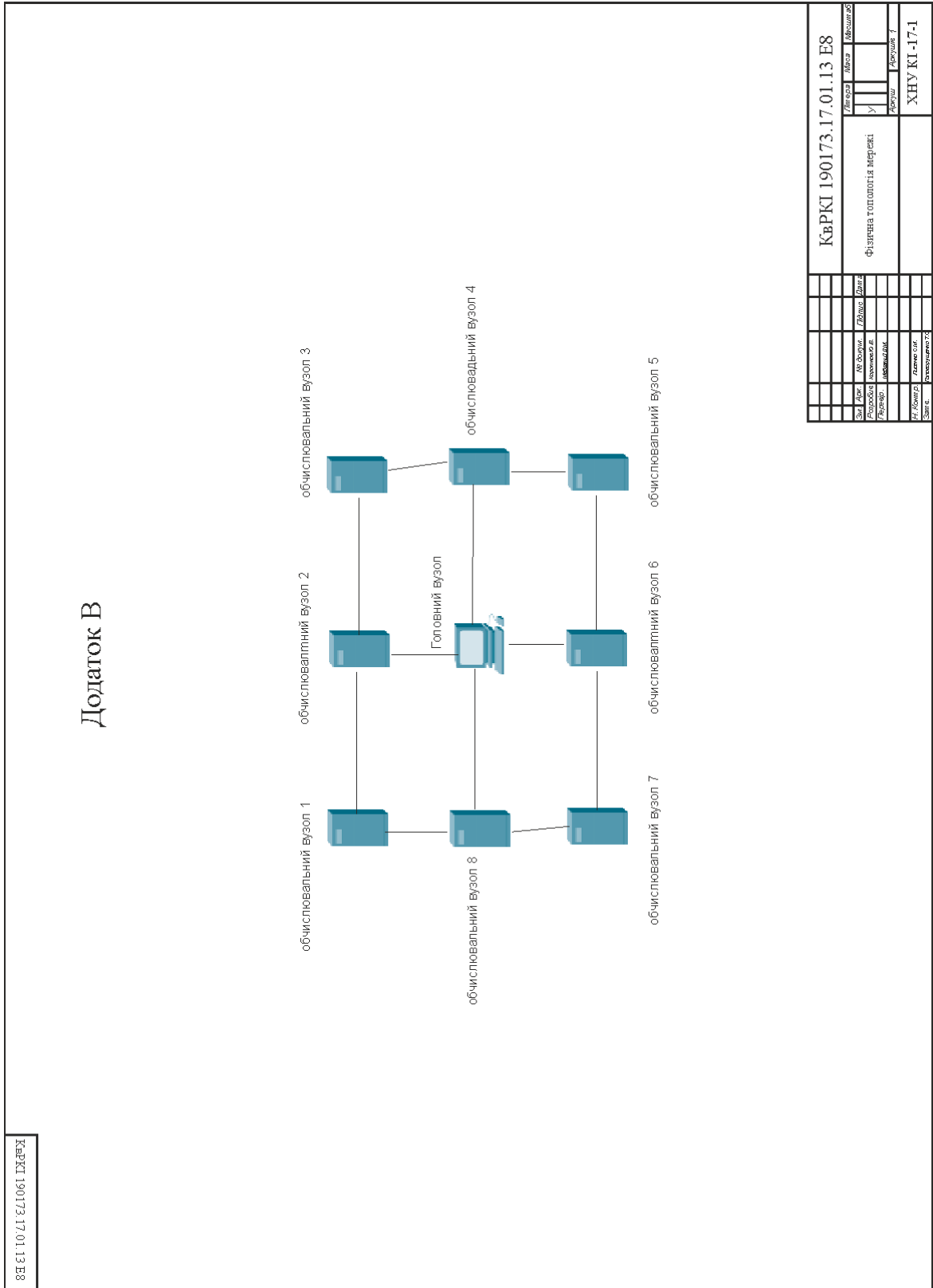
Додаток А
(обов'язковий)

Копія креслення «Копія креслення логічної топології кластерної комп'ютерної мережі»



Додаток В (обов'язковий)

Копія креслення «Копія креслення фізичної топології кластерної комп'ютерної мережі»



Ім'я користувача:
Кафедра КІ

Дата перевірки:
02.06.2021 17:58:40 EEST

Дата звіту:
02.06.2021 17:58:59 EEST

ID перевірки:
1008144154

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100005591

Назва документа: **Коротков_Кластерна система загального призначення на основі топології «Решітка»**

Кількість сторінок: 65 Кількість слів: 12319 Кількість символів: 91416 Розмір файлу: 2.89 MB ID файлу: 1008224548

7.35% Схожість

Найбільша схожість: 3.14% з Інтернет-джерелом (<https://shpora.me/mike/exam>)

6.81% Джерела з Інтернету

109

Сторінка 67

0.54% Джерела з бібліотеки

03

Сторінка 68

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

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

Замінені символи

3

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 0.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 12%**

ID: 92095 Название: Кластерна система загального призначення на основі топології «Решітка» Добавлено в БД: 2021-06-02 Авторы: Коротков Ю.В. Руководители: Медзатий Д.М. Консультанты: Опоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	81550	678	636 (1%)	7 (1%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник Коротков Юрій Володимирович

Тема Кластерна система загального призначення на основі топології «Решітка»

Спеціальність 123 Комп'ютерна інженерія

Обсяг кваліфікаційного проекту:

кількість листів креслень 3; кількість сторінок записки 60

1. Короткий зміст КП та прийнятих рішень В рамках кваліфікаційної роботи розроблено кластерну систему загального призначення на основі топології «решітка»

2. Висновок про відповідність КП кваліфікаційному завданню Кваліфікаційний проект у повній мірі відповідає поставленому завданню як в теоретичній, так і в практичній частині даного проекту

3. Характеристика виконання кожного розділу проекту, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому, теоретичному, розділі кваліфікаційного проекту якісно та в повній мірі розглянуті методи вирішення поставленої задачі, був проаналізований кожен аспект, який стосується теми кваліфікаційної роботи. У наступному розділі було здійснено обґрунтування обраної структури кластерної системи на основі порівняння різних можливих варіантів побудови цієї мережі. У основній проектній частині кваліфікаційної роботи була реалізована сучасними методами та рішеннями логічна структуризація системи. Резервування каналів Internet здійснено на основі новітніх можливостей операційних систем серверного типу.

4. Позитивні сторони проекту: Кваліфікаційна робота відповідає сучасним вимогам до проектування кластерних комп'ютерних систем та містить ряд інноваційних рішень. Також важливим позитивним аспектом кваліфікаційної роботи є реалізована можливість резервування каналів Internet, що надає можливість гарантувати постійний доступ користувачів мережі до певних серверів, які мають критичне значення.

5. Негативні сторони проекту Надмірна кількість теоретичного матеріалу, відсутність початкових налаштувань. В рамках кваліфікаційної роботи варто було приділити більшу увагу високошвидкісним способам передачі даних, оскільки потреба в даному рішенні шороку зростає.

6. Оцінка графічного оформлення та пояснювальної записки проекту Графічне оформлення виконане відповідно до суті кваліфікаційної роботи. У перших двох (основних) листах креслення відображені основні фізичні та логічні зв'язки системи, використання технології кластеризації системи. У додаткових листах креслення розглянуті питання резервування каналів та порівняння різних варіантів побудови кластерних систем. В загальному графічне оформлення виконане на належному рівні, єдиним недоліком є нерівномірність розмірів шрифтів, що робить розгляд графічних креслень дискомфортним. Пояснювальна записка відповідає задекларованим нормам для її оформлення.

7. Відгук про роботу в цілому. В загальному кваліфікаційна робота відповідає вимогам. Весь матеріал кваліфікаційної роботи структурований, чіткий та послідовний. Усі розділи проекту йдуть у вірній послідовності, що дозволяє чітко розуміти викладений матеріал в рамках даної кваліфікаційної роботи. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу при проектуванні кластерної системи.

8. Інші зауваження _____

9. Оцінка кваліфікаційної роботи. Розглянувши позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що він заслуговує оцінку «добре»/4.75/В.

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи) доцент кафедри автоматизації, комп'ютерно-інтегрованих технологій та телекомунікацій, к.т.н. Федула Микола Васильович

« 07 » червня 2021 р.

(підпис)

Завідувачу кафедри КІСП
д-ру техн.наук, проф. Говорущенко Т. О.

Короткова Юрія Володимировича.

ПІБ здобувача вищої освіти

ФПКТС, 4 курсу, групи КІ-17-1

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність плагіату ознайомлений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

10 червня 2021

дата



підпис

**РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Кластерна система загального призначення на основі топології «Решітка»

Автор: Коротков Юрій володимирович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Медзятий Дмитро Миколайович, д.т.н, професор

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

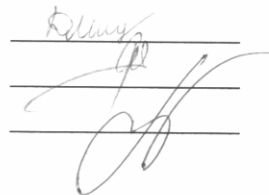
- 1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з 10-40 джерелами на один фрагмент речення;
- 4) в якості запозичень в окремих місцях системою зафіксовано загально доступну інформацію та команди налаштування, які є вхідними даними до великої кількості задач і не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення;
- 5) всі зафіксовані системою ознаки модифікації тексту відносяться до посилань на загальнодоступні ресурси в інтернеті, такі як наукові статті.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 7.35% і адресується до 232 першоджерела, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІСП



Медзятий Д.М.

С. М. Лисенко

Т. О. Говорущенко