

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

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“___” _____ 2026 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи керування оптичною
комунікаційною технікою ЦОД”

Виконав здобувач вищої освіти
IV курсу, групи КІ-22-2
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Борисова В.О.
«___» _____ 2026 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Буравченко К.О.
«___» _____ 2026 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітній ступінь бакалавр
Галузь знань_12 “Інформаційні технології”
Спеціальність 123 “Комп’ютерна інженерія”
Освітньо-професійна (освітньо-наукова) програма “Комп’ютерна інженерія”

ЗАТВЕРДЖУЮ
Завідувач кафедри
д.т.н., проф.
_____ Олексій СМІРНОВ
« 27 » лютого 2026 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ПЕРШИМ (БАКАЛАВРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Борисовій Валерії Олексіївні

(прізвище, ім’я, по батькові)

1. Тема роботи *Програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД*

2. Керівник роботи *Буравченко Костянтин Олегович, канд. техн. наук, доцент*
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 136-02 від 27.02.2026 року

3. Строк подання студентом роботи до захисту 23.05.2026 р.

4. Мета та завдання випускної кваліфікаційної роботи: *Метою роботи є розробка програмного забезпечення системи керування оптичною комунікаційною технікою ЦОД*

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Етапи програмування системи.

5. Впровадження системи в промислову експлуатацію.

6. Висновки

6. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

7. Дата видачі завдання « 27 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.03.2026 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2026 р.	
3.	Розробка моделі компонента	20.03.2026 р.	
4.	Розробка структур даних	25.03.2026 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.03.2026 р.	
6.	Програмування алгоритмів	10.04.2026 р.	
7.	Оформлення ПЗ	17.04.2026 р.	
8.	Попередній захист роботи	23.05.2026 р.	

Дата видачі завдання
« 27 » лютого 2026 р.

Підпис керівника

Буравченко К.О.
(прізвище та ініціали)

Завдання прийнято до виконання
« 27 » лютого 2026 р.

Підпис здобувача

Борисова В.О.
(прізвище та ініціали)

АНОТАЦІЯ

Борисова В.О. Програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи керування оптичною комунікаційною технікою ЦОД.

Метою розробки є програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД.

Результат роботи – програмна реалізація системи керування оптичною комунікаційною технікою ЦОД.

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

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

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

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, оптична комунікаційна техніка, ЦОД

ABSTRACT

Borysova V.O. Software for the control system of optical communication equipment of a data center. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed, which is intended for the control system of optical communication equipment of a data center.

The purpose of the development is the software for the control system of optical communication equipment of a data center.

The result of the work is the software implementation of the control system of optical communication equipment of a data center.

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

A convenient user interface has been developed. Instructions for working with software are provided.

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

The program was developed in the Python environment.

Keywords: computer engineering, optical communication technology, data center

3MIST

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	2
ВСТУП.....	3
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	10
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	11
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.....	11
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	21
2.3 Розгорнута постановка завдання	23
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	25
3.1 Опис функціонування системи	25
3.2 Розробка структурної схеми.....	32
3.3 Розробка функціональної схеми	39
3.4 Розробка діаграми процесів.....	49
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	51
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	51
4.2 Захист розробленого програмного забезпечення.....	79
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	81
6 ОСНОВНІ ВИСНОВКИ.....	87
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	89

					ВКРБ-123.26.0032.00.00.ПЗ				
Вим	Арк.	№ докум.	Підп.	Дата	Програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД	Літ.	Аркуш	Аркушів	
Розроб.		Борисова В.О.				Б		1	95
Перев.		Буравченко К.О.							
Н.контр.		Коваленко А.С.							
Затв.		Смірнов О.А.							
						ЦНТУ КІ-22-2			

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ГРП	—	головний розподільний пункт
ЕТМ	—	електротехнічна модель
ІТ	—	інформаційні технології
ЛОМ	—	локальна обчислювальна мережа
ПЗ	—	програмне забезпечення
ПРП	—	проміжний розподільний пункт
РПП	—	розподільний пункт поверху
СКС	—	структуровані кабельні системи
ЦОД	—	центр обробки даних
QoS	—	механізми контролю й керування якістю

ВСТУП

Актуальність теми. Рознімання LC потенційно дозволяє формувати тракти передачі зі швидкістю до 200 Гбіт/с, завдяки чому в доступній для огляду перспективі він буде як і раніше широко використовуватися при побудові інфраструктури ЦОД. Однак під впливом специфічних умов експлуатації конструкція з'єднувача перетерпіла зміни.

Структура кабельного тракту СКС досить жорстко зафіксована на нормативному рівні. Проектна й експлуатаційна гнучкість, необхідна для нормальної реалізації проектів, досягається за рахунок наявності декількох різновидів цього комплексного об'єкта. Будь-яка стаціонарна лінія й тракт завжди містять рознімні з'єднувачі, за допомогою яких лінійний і шнуровий кабелі з'єднуються між собою й потім підключаються (при необхідності через відповідні адаптери) до активного мережного устаткування.

Волоконно-оптичні кабельні тракти СКС для ЦОД діляться на два різновиди: дуплексні й багатоволоконні. Останні застосовуються, коли для збільшення пропускної здатності каналів зв'язку використовується принцип просторового ущільнення, тобто здійснюється паралельна передача по декількох волокнах. Для кожного різновиду тракту діючими редакціями стандартів передбачається свій тип з'єднувача: LC для дуплексних і МРО/МТР для багатоволоконних.

Споконвічно передбачалося, що дуплексні тракти будуть підтримувати швидкості до 10 Гбіт/с, хоча в перспективі можливе нарощування цього значення до 25 Гбіт/с. У випадку ж збільшення темпу передачі до 40 Гбіт/с і вище обов'язково повинен здійснюватися перехід на класичну паралельну передачу з декількома волокнами в одному напрямку.

У середині другого десятиліття ХХІ століття транспорт даних у багатьох ЦОД організується переважно на основі 40-гігабітних каналів зв'язку, при цьому

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

явно виражена тенденція переходу на швидкості 100 Гбіт/с, що веде до зменшення відносної частки LC.

Повному витисненню LC із проектів великих ЦОД перешкоджає підтримка їм наступних можливостей:

- збільшення тактової частоти сигналу до значення трохи вище 25 ГГц;
- застосування багаторівневого амплітудно-імпульсного кодування PAM4;
- впровадження схем короткохвильового спектрального мультиплексування BiDi і SWDM.

Крім того, стимулом до збільшення обсягів використання двоволоконних рішень служать:

- масове застосування технології агрегування каналів (port trunking) шляхом об'єднання широко розповсюджених 10-гігабітних потоків в один 40-гігабітний;
- побудова оптичної частини одного з 200-гігабітних варіантів новітніх інтерфейсних модулів формату QSFP-DD за схемою Dual LC.

Через дані обставини, перехід на схему фізичної паралельної передачі, а виходить, і відмова від LC не є безумовно обов'язковими, принаймні поки не досягнуті швидкості $10 \times 2,5 \times 2 \times 4 = 200$ Гбіт/с.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД.

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

- Огляд існуючих систем керування оптичною комунікаційною технікою ЦОД.
- Дослідження системи керування оптичною комунікаційною технікою ЦОД.
- Програмна реалізація системи керування оптичною комунікаційною технікою ЦОД.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі керування оптичною комунікаційною технікою ЦОД.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

К6ПЗ_2026

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Опишемо проблеми, що виникають при переході сигналу з одного світловода в інший. Втрата потужності або загасання оптичної хвилі виникає при неточному centruванні світловодів. У цьому випадку частина променів просто не переходить у наступний світловод, або входить під кутом більше критичного. При неповному фізичному контакті волокна утвориться повітряний зазор. У зв'язку із чим виникає ефект зворотних втрат. Частина променів при проходженні прозорих середовищ із різною щільністю відбивається у зворотному напрямку. Досягаючи резонатора, вони підсилюються й викликають перекручування сигналів.

Неідеальна геометрична форма волокон також вносить вклад у втрати потужності. Це може бути й еліптичність світловода й нецентричність його серцевини. Торець самого світловода може містити деформації: відколи й шорсткості, що у свою чергу зменшує робочу поверхню зіткнення волокон.

Наконечники оптичних коннекторів

У такий спосіб необхідно точно й щільно сполучити обоє світловода. Щоб забезпечити схоронність тендітного волокна при багаторазовому сполученні, їх прикінцеві відрізки поміщають у керамічні, пластмасові або сталеві наконечники. Більшість наконечників мають циліндричну форму з діаметром 2,5 мм. Зустрічаються конічні конструкції, а коннектори LC мають наконечник діаметром 1,25 мм.

Усередині наконечників існує канал, у який уводиться й фіксується хімічним або механічним способом очищений від оболонки світловод. При видаленні захисного покриття можуть використовуватися як спеціальні механічні інструменти, так і хімічно активні розчини. Усередині наконечника світловод

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

може фіксуватися як по всій довжині каналу (частіше це методи на основі клеячи), так і в кінці введення волокна в наконечник (механічні методи). Процес механічної фіксації займає набагато менше часу (до декількох хвилин) і заснований на "придавлюванні" волокна за допомогою полімерних матеріалів. Але він є менш надійним і недовговічним. Хімічний спосіб говорить сам за себе. Найчастіше фіксуючим составом у даній технології виступають епоксидні розчини, як найбільш надійні. Однак період повного загустівання такого состава досить тривалий – до доби. Тому при необхідності більше швидкого монтажу коннекторів можуть застосовуватися інші компоненти або спеціальні печі для сушіння. Після установки світловода в коннектор необхідно відшліфувати торець наконечника. Виступаючий надлишок волокна віддаляється спеціальними інструментами. Основний принцип полягає в надрізі й обламуванні світловода, після чого можна приступати до безпосереднього полірування поверхні.

Особливий інтерес викликає форма торців наконечників. Їхня обробка являє собою ціле мистецтво. Найпростіший варіант торця – плоска форма. Їй властиві більші зворотні втрати, оскільки ймовірність виникнення повітряного зазору в околиці світловодів велика. Досить нерівностей навіть у неробочій частині поверхні торця. Тому частіше застосовуються опуклі торці (радіус закруглення становить порядку 10-15 мм). При гарному центруванні щільне зіткнення світловодів гарантується, а значить більш імовірно відсутність повітряного зазору. Ще більш просунутим рішенням є застосування закруглення торця під кутом у кілька градусів. Округлені торці менше залежать від деформацій, утворених при з'єднанні коннекторів, тому подібні наконечники витримують більша кількість підключень (від 100 до 1000).

Також важливий матеріал наконечника. Більша частина числа коннекторів будуються на основі керамічних наконечників, як більше стійких.

Після прикінцевування світловодів коннекторами необхідно зробити аналіз якості поверхні наконечника. Найчастіше для цього застосовуються мікроскопи. Професіональні прилади мають кратність збільшення в сотні разів і

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

постачені спеціальним підсвічуванням з різних ракурсів. Вони можуть також мати інтерфейс підключення до додаткового вимірювального устаткування. Відповідно до стандарту TIA/EIA 568A величина зворотних втрат для багатомодового волокна в оптичних конекторах не повинна перевищувати -20 Дб, а для одномодового -26 Дб.

LC-коннектор

Коннектори типу LC – це малогабаритний варіант SC-коннекторів . Він також має прямокутний перетин корпуса. Конструкція виконується на пластмасовій основі й постачена засувкою, подібній до засувки, що застосовується в модульних конекторах мідних кабельних систем. Внаслідок цього й підключення конектору виробляється схожим образом. Наконечник виготовляється з кераміки й має діаметр 1.25 мм. Зустрічаються як багатомодові, так і одномодові варіанти коннекторів. Ніша цих виробів – багатопоРТові оптичні системи.

FC-коннектор

В одномодових системах зустрічається ще один різновид коннекторів – FC. Вони характеризуються відмінними геометричними характеристиками й високим захистом наконечника.

FDDI-коннектор

Для підключення дуплексного кабелю можуть використовуватися не тільки спарені SC-коннектори. Часто в цих цілях застосовують FDDI-коннектори. Конструкція виконується із пластмаси й містить два керамічних наконечники. Для виключення неправильного підключення лінка коннектор має несиметричний профіль. Технологія FDDI передбачає чотири типи використовуваних портів: А, В, S і М. Проблема ідентифікації відповідних лінків вирішується за рахунок постачання коннекторів спеціальними вставками, які можуть розрізнятися по колірній гамі або містити буквені індекси. В основному даний тип використовується для підключення до оптичних мереж прикінцевого устаткування.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

MT-RJ-коннектори

Гарантовані параметри кабельних складань:

- Прямі втрати <0.5 дБ (типове значення – 0.25 дБ для ММ).

Області застосування:

- Проводка в будинках (горизонтальна й backbone).
- LAN.
- Телекомунікаційні мережі.

Особливості:

- Розмір і конструкція засувки аналогічні RJ-45.
- Дуплексний ферул.
- Низька вартість.
- Висока щільність портів.
- Відповідність стандартам ISO/IEC 11801 і TIA/EIA 568A.
- Низькі прямі втрати:.

< 0.22 дБ для ММ

< 0.19 дБ для ОМ

Розробка коннектору MT-RJ переслідувала рішення наступних завдань: малий розмір, низька вартість і простота установки. Використання коннектору MT-RJ збільшує щільність портів у два рази в порівнянні зі стандартними коннекторами й робить його ідеальним для використання в додатках типу to-the-desk. Дизайн коннектору відповідає вимогам TIA. У коннекторі MT-RJ використовується поліпшена версія індустріального стандарту для коннекторів типу RJ-45. Саме малий розмір і зручність засувки аналогічної RJ-45 визначають переваги даного коннектору при використанні в горизонтальній проводці до робочого місця. Особливістю системи MT-RJ від Molex є використання різних PN для коннекторів модифікації «тато» (з напрямними штирьками, що виступають із ферула) і «мама» (з дірочками під штирьки). Є дві модифікації адаптера, одна з яких установлюється в гніздо для симплексного SC адаптера.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

1.2 Область застосування

Областю застосування є структуровані кабельні системи (СКС) побудовані з використання оптико-волоконних ліній передачі даних у ЦОД, а також побудова топології цих систем. Різноманітні технології, архітектури й додатка, голосові, інформаційні дані, передача відео зображень і контрольних сигналів, системи контролю доступу, сигналізації, відеоспостереження – все повинне сполучатися й діяти як єдине ціле. СКС – єдина інженерна інфраструктура будинку, завдяки якій можливе створення інтегрованої системи, повністю прозорої для користувачів і не залежної від використовуваних додатків. Цінність і актуальність структурованої системи полягає в тім, що вона забезпечує універсальний, незалежний сервіс і підключення будь-якого стандартного устаткування, роботу будь-якого стандартного додатка. На базі СКС можлива реалізація різноманітних нестандартних додатків за допомогою застосування спеціальних пристроїв – адаптерів, конверторів, розгалужувачів і подібних їм пристроїв.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи керування оптичною комунікаційною технікою ЦОД, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

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

napoCAD СКС

Проектування структурованих кабельних систем (СКС) будинків і споруджень різного призначення, кабеленесучих систем і телефонії.

Автоматичне розміщення робочих місць

В napoCAD СКС реалізовані алгоритми автоматичного розміщення робочих місць виходячи із площі приміщення. Число робочих місць у приміщенні можна настроїти як для всього проекту, так і для кожного приміщення індивідуально. Автоматична установка коробів у приміщеннях, автоматичне заповнення шафи комутаційним устаткуванням і автоматичним трасуванням кабелів дозволяють швидко виконати проект на стадії «П».

Автоматичне маркування устаткування

В napoCAD СКС реалізована автоматичне маркування робочих місць установлених на кресленні. У налаштуваннях програми задаються маски маркування для кожного порту робочого місця. Всі маркування інтелектуально відслідковуються й при зміні в масці маркування й, наприклад, при зміні порядку проходження робочих місць у приміщенні або на поверсі. Затверджені маркування робочих місць у силу можна автоматично додати на креслення.

Проектна документація

За результатами роботи в napoCAD СКС формуються наступні проектні документи:

- плани розташування устаткування й прокладки кабельних трас;
- специфікація устаткування, виробів і матеріалів;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

- кабельні журнали;
- табличні документи.

Технологія проектування OpenBIM

Продукт папоCAD СКС повністю реалізує основний принцип проектування Open BIM: створення єдиної інформаційної моделі проекрованої системи за допомогою найбільш підходящим і перевірених часом інструментів. Завдяки підтримці експорту в обмінні файли стандарту IFC, інформаційні моделі систем безпеки, виконані в папоCAD СКС, легко вбудовуються в загальну інформаційну модель проектового об'єкта на будь-якій BIM-платформі, будь те ARCHICAD, Revit, Allplan або яка-небудь інша.

Бази дані устаткування

До програми папоCAD СКС додаються бази даних виробників структурованих кабельних систем, шаф і кабеленесучих систем. Всі бази даних легко редагуються й поповнюються. Крім того, для групи користувачів можна організувати загальну мережну бібліотеку баз дані устаткування на сервері. При запуску програми локальні бази синхронізуються з мережною базою, що дозволяє працювати навіть без мережного підключення.

Кабеленесучі системи

В папоCAD СКС можна створювати системи кабельних каналів практично будь-якої складності з використанням трьох основних типів кабельних каналів: лотків, труб і коробів.

У програмі реалізована функція по визначенню орієнтації траси, що дозволяє автоматично визначати типи використовуваних сполучних елементів при поворотах траси, наприклад, внутрішні й зовнішні кути для коробів.

При налаштуваннях конфігурацій лотків можна також набудовувати й використовувати конфігурації вузлів кріплення, щоб одержувати детальну специфікацію й автоматично створювати динамічні перетини лотків.

Весь комплекс підсистем СКС

Система папоCAD СКС містить наступні підсистеми:

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

- Підсистема робочого місця.
- Горизонтальна підсистема.
- Магістральна підсистема будинку.
- Телекомунікаційна підсистема.

Область застосування

Система папоCAD СКС значно підвищує ефективність проектування структурованих кабельних систем («слаботочки»). Завдяки власному графічному ядру, папоCAD СКС не залежить від інших графічних систем, а підтримка формату DWG дозволяє швидко обмінюватися інформацією із суміжниками й замовниками.

Програмний продукт папоCAD СКС призначений для рішення наступних завдань:

- консолідація інформації із проекту;
- створення системи кабельних каналів;
- проектування горизонтальної підсистеми;
- проектування магістральної підсистеми будинку;
- проектування розподільних пунктів поверху й будинку;
- проектування магістральних кабелів і кросів телефонії будинку;
- автоматичне трасування кабелів;
- автоматичне складання звітних документів (специфікація, кабельні журнали, схеми komponування монтажних конструктивів, структурна схема).

Ядро системи

Платформа папоCAD Plus – російська універсальна САПР, що містить всі необхідні інструменти базового проектування й випуску креслень.

Освоїти папоCAD Plus зможе практично будь-який проектувальник, що має досвід роботи в популярних САПР: меню, кнопки, панелі й командний рядок легко пізнавані. Все це дозволяє швидко приступитися до роботи, затративши мінімум часу на перенавчання персоналу.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

Організація роботи

Робота в папоCAD СКС побудована навколо інструмента Менеджер проекту – фактично центральної бази даних проекту, що містить креслення, автоматично формовані звіти й результати розрахунків, а також дозволяє збирати всі необхідні документи для виконання проекту (технічне завдання, пояснювальні записки й т.п.) і гнучко набудовувати будь-які елементи.

Програма папоCAD СКС дозволяє завантажувати векторну архітектурно-будівельну підоснову плану спорудження, у тому числі у форматі DWG.

У рішення вже убудовано 16 баз даних виробників кабельних систем, шаф і кабеленесучих систем. Всі бази даних легко редагуються й поповнюються в міру роботи над проектом.

Крім того, для групи користувачів можна організувати загальну мережну бібліотеку баз дані устаткування на сервері. При запуску програми локальні бази синхронізуються з мережною базою, що дозволяє працювати навіть без мережного підключення.

Моделювання

Продукт папоCAD СКС – це швидкий і безболісний спосіб перейти від роботи з окремими кресленнями до інформаційного моделювання структурованих кабельних систем. Єдина модель системи дозволяє вносити зміни «на лету», тому що вся зв'язана інформація обновляється автоматично, що скорочує число помилок і розбіжностей.

У цілому побудова інформаційної моделі в процесі проектування дозволяє:

- максимально наблизити проект до умов монтажу й експлуатації системи;
- робити розрахунки з урахуванням технічних характеристик використовуваного в проекті устаткування;
- використовувати методи оцінного розрахунку устаткування;
- завжди мати актуальну й погоджену інформацію із проекту;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

- моментально вносити графічні й технічні зміни.

Створення структурованих кабельних систем

Система папоCAD СКС містить наступні підсистеми:

- **Підсистема робочого місця** При роботі з підсистемою робочого місця враховуються різні варіанти установки робочих місць (на стіну, у короб, лючок, сервісну колону й т.п.) і будь-які варіанти комплектації кожного робочого місця. Робочі місця формуються як вручну, так і в автоматичному режимі, а також мають унікальне маркування й графічне позначення.

- **Горизонтальна підсистема** Трасування кабелю горизонтальної підсистеми відбувається в автоматичному режимі по кабельних каналах або по координатах установки устаткування, що з'єднується, при оцінному розрахунку кабелю. При цьому враховуються запаси кабелю на укладання в кабельних каналах, укладання в шафі й укладання в коробі з боку робочого місця.

- **Магістральна підсистема будинку** При моделюванні в папоCAD СКС всі поверхи будинку зв'язуються в єдину модель, що дозволяє без праці трасувати кабель магістральної підсистеми будинку між кресленнями поверхів.

- **Телекомунікаційна підсистема** При роботі із шафами в папоCAD СКС можна автоматично заповнити їх кроссовим устаткуванням виходячи з кількості портів робочих місць. При автоматичному заповненні враховуються як панелі кросу СКС, так і панелі телефонного кросу й комутатори або панелі другого подання портів комутаторів.

Створення системи кабельних каналів

В папоCAD СКС можна створювати системи кабельних каналів практично будь-якої складності з використанням трьох основних типу кабельних каналів: лотків, труб і коробів.

У програмі реалізована функція по визначенню орієнтації траси, що дозволяє автоматично визначати типи використовуваних сполучних елементів при поворотах траси, наприклад, внутрішні й зовнішні кути для коробів.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

При налаштуваннях конфігурацій лотків можна також набудувати й використовувати конфігурації вузлів кріплення, щоб одержувати детальну специфікацію й автоматично створювати динамічні перетини лотків

Після трасування кабелю по кабельних каналах розраховується ємність кабельних каналів з урахуванням всіх типів прокладених кабелів, а також з урахуванням максимального відсотка заповнення кожного типу кабельного каналу. Якщо поперечний переріз кабелів вище припустимого для кабельного каналу, папоCAD CRC видасть помилку у вікні перевірок.

Також у програмі реалізоване автоматичне розміщення коробів у приміщеннях для робочих місць із урахуванням заданої висоти монтажу й геометрії приміщення.

Робота з електротехнічною моделлю

Всі з'єднання в проєкті включаються в єдину електротехнічну модель (ETM), що дозволяє швидко й безпомилково створити з'єднання як горизонтальної, так і магістральної підсистеми будинку.

ETM проєкту містить у собі ті ж команди, що й на панелях інструментів, що дозволяє перемаркірувати об'єкти, навіть не відкриваючи саме креслення. При роботі із шафами можна автоматично заповнити їх кроссовим устаткуванням виходячи з кількості портів робочих місць. При автоматичному заповненні враховуються як панелі кросу СКС, так і панелі телефонного кросу й комутатори або панелі другого подання портів комутаторів.

Загальна електротехнічна модель кабельної системи формується:

- при виконанні автоматичного трасування кабелю по кабельних каналах – як по горизонтальним, так і по вертикальних ділянках;
- при маркуванні устаткування, використовуваного в з'єднаннях кабельної системи. Значення маркування автоматично обновляються при внесенні змін у проєкт;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

3D-модель проекрованої системи

3D-модель будується прямо на кресленні плану поверху, що дозволяє мати доступ до об'єктів і редагувати їхні характеристики.

При створенні 3D-моделі кожний елемент розміщується у власний шар, тому видимість об'єктів в DWG-Файлах гнучко регулюється.

Вивантаження в IFC

Продукт папоCAD СКС повністю реалізує основний принцип проектування Open BIM: створення єдиної інформаційної моделі проекрованої системи за допомогою найбільш підходящим і перевірених часом інструментів. Завдяки підтримці експорту в обмінні файли стандарту IFC, інформаційні моделі систем безпеки, виконані в папоCAD СКС, легко вбудовуються в загальну інформаційну модель проектованого об'єкта на будь-якій BIM-платформі, будь те ARCHICAD, Revit, Allplan або яка-небудь інша.

Документування проекту

Система папоCAD СКС не тільки мінімізує помилки при проектуванні, але й автоматично формує звітні документи за вітчизняними стандартами для наступного вивантаження або на поле креслення, або в Microsoft Office або OpenOffice.org. У будь-який момент користувач може одержати наступні документи:

- робочі креслення поетажних планів, оформлені відповідно до вітчизняних стандартів, з автоматично промартільним устаткуванням і розставленими винесеннями, а також з можливістю додавання рамки за ДСТУ 21.1101-2013;
- три варіанти кабельного журналу, що дозволяють відслідковувати зв'язку горизонтальної й магістральної підсистем будинку;
- відомість креслень основного комплекту, відомість посилавальних і прикладених документів за ДСТУ 21.1101-2013;
- експлікація приміщень за ДСТУ 21.501-93;
- таблиці прокладки кабелю;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

- таблиця використовуваних УГО, з можливістю її створення як для всього проекту, так і для кожного плану поверху;
- звіт по робочих місцях, що відображає кількість установлених робочих місць і кількість і типи портів робочих місць по приміщеннях і поверхам, а також і по всьому будинку;
- специфікація устаткування й матеріалів за ДСТУ 21.110-95. Дані вносяться в специфікацію за принципом «що є в плані поверху, тобто й у звіті».
- автоматично створювана схема заповнення монтажної шафи комутаційним устаткуванням;
- автоматично генеруєма структурна схема з'єднань.

Проектування СКС

СКС – це структурована кабельна система, універсальна система будинку або групи будинків, призначена для використання на тривалий період часу без реструктуризації. Автоматичне проектування СКС можливо за допомогою нашої програми – «Експерт-СКС».

Проектування СКС – завдання підрядних організацій. При проектуванні СКС ураховуються технічні й архітектурні особливості будинків і приміщень, точне розташування всіх робочих місць, гнучкість конфігурації й можливості її розширення. Перший етап СКС – це технічне завдання, що містить у собі вимоги замовника по числу й розташуванню робочих місць, класи й категорії системи.

Проектування СКС на другому етапі – це комерційна пропозиція, що містить пояснювальну записку, структурні схеми СКС, специфікація робіт і устаткування. Технічні рішення визначаються добре підготовленою комерційною пропозицією. Проектування СКС – це найважливіший щабель створення СКС компанії.

Пропонується програма «Експерт-СКС», за допомогою якої ви зможете:

- Використовувати повного функціонала для проектування геодезичних, архітектурних і інших схем.
- Проектувати СКС будь-якого виду на вихідному плані.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

– Одержите доступ до бази даних стандартних виробів відомих виробників.

– Повна навігація й коректування готової СКС.

– Випуск розрахункових документів (плани, креслення, відомості, специфікації, журнали й ін.).

Хочете швидко, комплексно й зручно автоматизувати всі процеси проектування СКС, тоді скористайтеся програмою «Експерт-СКС». Використовуючи програму, зможете швидко й професійно становити проекти, підвищите якість виконуваних робіт і просто станете майстром у справі.

Переваги програми автоматизації проектування мережного устаткування в Структурованих Кабельних Мережах – «Експерт-СКС»:

– Велика база елементів по всіх великих виробниках.

– Проектування й подання СКС, як окремої інфраструктури.

– Можливість використання готових наробітків (параметричне проектування).

– Складання будь-яких конфігурацій вузлів.

– Повний контроль всіх з'єднань, виключення помилок.

– Точний розрахунок коробів, кабелів, а також вартості СКС на будь-якому етапі проектування.

– Експорт у системи складання кошторисних документів і подальше обґрунтування витрат.

– Всі економічні й технічні документи (ДСТУ, ISO, CD, TIA, EN, prEN).

– Автоматичне маркірування ділянок СКС і вузлів відповідно до обраним вас шаблоном.

«Експерт-СКС» має можливість видавати результати на українській, російській і англійській мовах.

Проектування СКС кабельних мереж у програмі CAD5D

Веб-застосунок по проектуванню. Хмарний CAD для проектування інженерних мереж і будівельних робіт.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

Функції:

- Авторастановка розеток.
- Автопрокладка кабелів.
- Авторозрахунок з урахуванням запасу кабелю.
- 3D вид.
- Специфікація (Відомість ресурсів).
- Кабельний журнал.
- Журнал устаткування.
- Кошторис.
- Діаграма Шафи.
- Схема проекту (багатоповерхова).
- Діаграма міжшкафних з'єднань.
- Діаграма по оптоволокну.

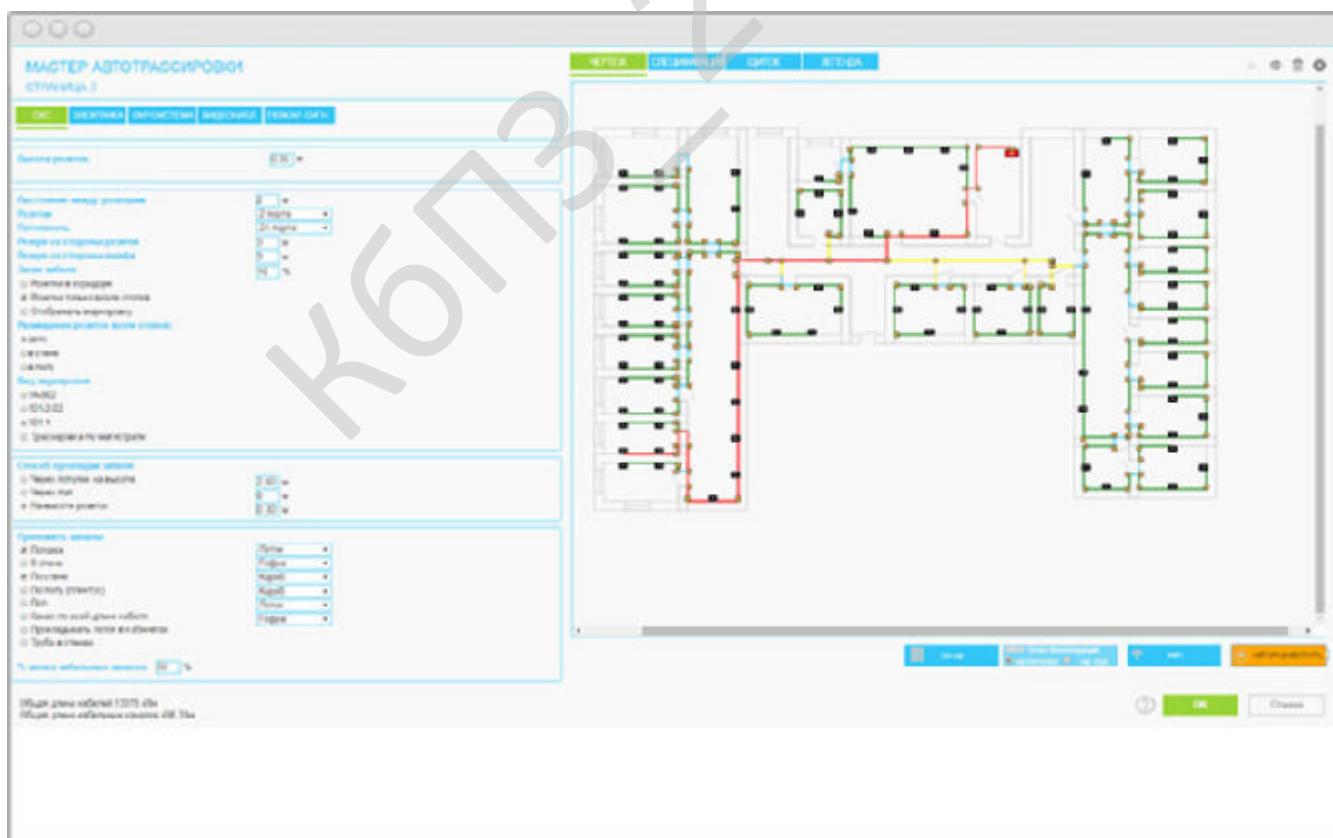


Рисунок 2.1 – Интерфейс користувача CAD5D

Автотрасування

CAD5D автоматично розставить розетки й зробить автоматичне підключення до шафи. Усього в парі кліків виробляється автомаркування.

3D вид

3D вид від першої особи й зверху, з усім устаткуванням для демонстрації ваших об'єктів.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Як мова програмування обрана Python. Python – високорівнева мова програмування загального призначення з акцентом на продуктивність розроблювача й читаність коду. Синтаксис ядра Python мінімалістичний. У той же час стандартна бібліотека включає великий обсяг корисних функцій.

Python підтримує кілька парадигм програмування, у тому числі структурне, об'єктно-орієнтоване, функціональне, імперативне й аспектно-орієнтоване. Основні архітектурні риси – динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багатопоточні обчислень і зручні високорівневі структури даних. Код у Python організовується у функції й класи, які можуть поєднуватися в модулі (які у свою чергу можуть бути об'єднані в пакети).

Еталонною реалізацією Python є інтерпретатор CPython, що підтримує більшість активно використовуваних платформ. Він поширюється вільно під дуже ліберальною ліцензією, що дозволяє використовувати його без обмежень у будь-яких застосунках, включаючи пропрієтарні. Є реалізації інтерпретаторів для JVM (з можливістю компіляції), MSIL (з можливістю компіляції), LLVM і інших. Проект PyPy пропонує реалізацію Python на самому Python, що зменшує витрати на зміни мови й постановку експериментів над новими можливостями. Python – мова програмування, що активно розвивається, нові версії (з додаванням/змінюю

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

язикових властивостей) виходять приблизно раз у два з половиною року. Внаслідок цього й деяких інших причин на Python відсутні ANSI, ISO або інші офіційні стандарти, їхня роль виконує CPython.

Python портований і працює майже на всіх відомих платформах – від КПК до мейнфреймів. Існують порти під Microsoft Windows, практично всі варіанти UNIX (включаючи FreeBSD і Linux), Plan 9, Mac OS і Mac OS X, iPhone OS 2.0 і вище, Palm OS, OS/2, Amiga, AS/400 і навіть OS/390, Symbian і Android.

При цьому, на відміну від багатьох портуємих систем, для всіх основних платформ Python має підтримку характерних для даної платформи технологій (наприклад, Microsoft COM/DCOM). Більше того, існує спеціальна версія Python для віртуальної машини Java – Jython, що дозволяє інтерпретаторові виконуватися на будь-якій системі, що підтримує Java, при цьому класи Java можуть безпосередньо використовуватися з Python й навіть бути написаними на Python. Також кілька проектів забезпечують інтеграцію із платформою Microsoft .NET, основні з яких – IronPython і Python.Net.

Python підтримує динамічну типізацію, тобто тип змінної визначається тільки під час виконання. Тому замість «присвоювання значення змінної» краще говорити про «зв'язування значення з деяким ім'ям». У Python є убудовані типи: бульові, рядки, Unicode-рядки, цілі числа довільної точності, числа із плаваючою комою, комплексні числа й деякі інші. З колекцій Python підтримує кортежі (*tuples*), списки, словники (асоціативні масиви) і, починаючи з версії 2.4, безлічі. Всі значення в Python є об'єктами, у тому числі функції, методи, модулі, класи.

Додати новий тип можна або написавши клас (*class*), або визначивши новий тип у модулі розширення (наприклад, написаному мовою C). Система класів підтримує спадкування (одиначне й множинне) і метапрограмування. Можливе спадкування від більшості убудованих типів і типів розширень.

Всі об'єкти діляться на посилальні й атомарні. До атомарного ставляться *int*, *long*, *complex* і деякі інші. При присвоюванні атомарних об'єктів копіюється їхнє значення, у той час як для посилальних копіюється тільки покажчик на

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

об'єкт, таким чином, обидві змінні після присвоювання використовують те саме значення. Посилальні об'єкти бувають змінювані й незмінні. Наприклад, рядки й кортежі є незмінними, а списки, словники й багато інших об'єктів – змінюваними. Кортеж у Python є, по суті, незмінним списком. У багатьох випадках кортежі працюють швидше списків, тому якщо ви не плануєте змінювати послідовність, то краще використовувати саме їх.

Мова має чіткий і послідовний синтаксис, продуману модульність й масштабованість, завдяки чому вихідний код написаних на Python програм легко читаємий.

Python – стабільна й розповсюджена мова. Він використовується в багатьох проектах і в різних якостях: як основна мова програмування або для створення розширень і інтеграції застосунків. На Python реалізоване велика кількість проектів, також він активно використовується для створення прототипів майбутніх програм. Python використовується в багатьох великих компаніях.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи керування оптичною комунікаційною технікою ЦОД.

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

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

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

в) розробити програмне забезпечення системи, що дозволить реалізувати

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

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

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

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

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

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

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Особливі вимоги відносно застосування LC у ЦОД

Відвід тепла, виділюваного в простір апаратної зали ЦОД працюючими серверами й іншим комп'ютерним і телекомунікаційним устаткуванням, здійснюється переважно за допомогою системи повітряного охолодження. Для забезпечення її ефективного функціонування необхідно, щоб:

- охолоджуваний об'єкт мав мінімальний обсяг;
- 19-дюймовий монтажний конструктив входив до складу системи в якості одного з компонентів структури гарячих і холодних коридорів.

Іншими словами, комутаційне устаткування оптичної підсистеми інформаційної проводки ЦОД повинне мати збільшену щільність конструкції. Зворотним боком такого підходу є виникнення певних проблем з комутацією.

Оптичний інтерфейс двоволоконних модулів SFP, SFP+ і аналогічних їм виконаний у форм-факторі LC. Щоб зберегти зручне для експлуатації симетричне виконання різних шнурових виробів, компоненти оптичного рознімання LC доцільно доробити з урахуванням високої щільності конструкції комутаційної техніки. Модернізовані варіанти LC:

- як і раніше ставляться до групи малогабаритних, або SFF-коннекторів (від англ. Small Form Factor);
- зберігають повну або забезпечують дуже високий ступінь сумісності із традиційними виробами;
- мають тільки дуплексну форму виконання;
- постачені елементами, що полегшують відключення від розеток навіть за умови їх щільного багаторядного розташування на лицьовій пластині оптичних полиць;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

– дозволяють легко, без використання спеціальних інструментів, міняти полярність у процесі експлуатації устаткування.

При реалізації спеціалізованих конструкцій додатково враховуються наступні фактори:

– користувачі не зацікавлені в зміні процесу підключення існуючих вилок LC;

– гранична простота дизайну розеток не дозволяє радикальним образом поліпшити їхню конструкцію;

– площа миделя вилок типового LC набагато перевищує перетин навіть 32-волоконного стрічкового кабелю (максимальна кількість волокон у схемі паралельної передачі);

– для відключення вилки від розетки не обов'язково мати прямий доступ до фіксатора;

– конфігурація СКС у ЦОД міняється досить рідко.

Таким чином, основні зусилля розроблювачів зосереджуються на зміні конструкції дуплексної вилки. Більше того, конструктивним доробкам піддаються переважно середня й почасті задня частини її корпусу.

Робочу частину елемента відключення фіксатора качани в розетці можна відсунути від передньої панелі оптичного кросу без погіршення споживчих якостей рознімання. Для цього в конструкцію вилки вводиться механізм, що дозволяє міняти напрямок впливу на важіль фіксатора засувки: осьове тягнуче зусилля перетворюється у вертикальне. Сам же механізм фіксації залишається незмінним або переробляється незначно.

При впливі, що відключає, важіль засувки переміщається в напрямку корпусу. Воно може бути створене за рахунок тиску на важіль зверху або ж шляхом нижнього тягнучого зусилля.

Реалізація першого підходу в більшості випадків припускає використання верхньої довгої твердої стрижневої тяги. Другий заснований на тязі з м'яким закінченням, притиснутої впритул до корпусу вилки. Сама тяга може бути

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

твердої стрижневий (цей підхід реалізований у виробках LC-HD японської компанії Senko) або виконаної за м'якою схемою – у вигляді стрічки.

Для зручності роботи вільний кінець важеля забезпечується наверхшм кулястої, циліндричної, конусоподібної або іншої форми, зручної для захвата двома пальцями.

Ще один спосіб підвищення зручності роботи – застосування знімної тяги, що підключається до вилки тільки в момент відключення. Робочий кінець такої тяги має Т-образну форму, а на движку механізму відключення передбачено два виступи, за які чіпляються відгини тяги. Знімна тяга має ту ж довжину, що й типовий ключ, тому їх можна носити в одному зв'язуванні – для цього на другому кінці тяги передбачене кільце. Дане рішення застосовується в серійній продукції американської компанії Sanwa.

Поряд з механізмами стрижневого типу можлива їхня реалізація по обойменній схемі. У цьому випадку відключення качани відбувається за класичною схемою push-pull. Суть рішення складається в застосуванні підпружиненої рухливої обойми, що повністю охоплює корпус і при лінійному переміщенні впливає на важіль засувки. Такий механізм відключення застосовується принаймні у двох серійних розніманнях (BladePatch компанії Siemon і LX-HD компанії Huber + Suhner). Основні відмінності – довжина й форма движка.

Двигун обойменного механізму звичайно впливає безпосередньо на важіль засувки. Однак твердих обмежень на довжину його ходу ні, що дає можливість увести в кінематичну схему проміжний елемент, що відтягає долілиць задню частину важеля засувки. Перевагою такого підходу стає збільшення плеча сили, що трохи полегшує відключення.

Відомо ще одне рішення, в основу якого покладена відмова від тягнучого зусилля на користь натискання. Механізм, що його реалізує, являє собою довгий важіль, пов'язаний зі штовхачем засувки через проміжний шарнір. Для відключення вилки робочий кінець важеля варто підняти, а не давити на нього

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

долілиць. Можливість застосування такого підходу доведена його впровадженням у деякі зразки подібних по дизайні рознімань LX.5 і F-3000. Про його використання в розніманнях LC нічого невідомо.

Рішення по зміні полярності вилок

Через порівняно малі обсяги використання волоконної оптики на рівні горизонтальної підсистеми ймовірність виконання процедури зміни полярності в офісних СКС дуже мала. Інакше відбувається справа в ЦОД, де лінійна частина стаціонарної лінії реалізується на основі стрічкового кабелю.

Необхідність зміни полярності елементів оптичного рознімання часто виникає при формуванні на їхній базі дуплексних трактів передачі. Зміна полярності дуплексних рознімань може виконуватися як на качанні, так і в розетці. З погляду форм-фактора вилок LC має несиметричну структуру. Тому перехід на іншу полярність при незмінному положенні фіксуючої засувки може бути реалізований відповідно до двох основних схем: перестановка вилок або різні види їхнього повороту.

Перший прийом широко практикувався відносно вилок, які збираються в дуплексну конструкцію за допомогою багаторазової пружної кріпильної обойми.

Рішення другої групи більше зручні, але для їхньої реалізації потрібне ускладнення конструкції вилок. Один з варіантів цієї схеми заснований на тім, що передня частина корпусу вилок допускає незалежне обертання навколо наконечника при відкритому тримачі (. У цьому випадку корпус обертається разом з важелем засувки, а в завершення процедури потрібно перевернути захисну кришку на 180°. Для позначення окремих вилок, крім їхнього фарбування в різні кольори, використовується символічне маркування А і В, що наноситься на різні сторони корпусу.

Оригінальний спосіб зміни полярності цієї групи запропонований тайваньською компанією JYH ENG TECHNOLOGY. Він застосовний до моноблочної дуплексної вилки, у якої, на відміну від традиційних конструкцій, верхня частина корпусу окремих вилок має гладку поверхню. Фіксатор

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

реалізований за схемою із прямою орієнтацією важеля засувки і кільцевий утримувачкою у вигляді асисиметричної обойми, що надягається на задню частину корпусу. Для зміни полярності досягну зрушують назад до виходу на кабель, перевертають на 180° і знову насувають на штатне місце.

Захист важеля засувки

Як прототип корпусу вилок LC був обраний аналогічний елемент модульного рознімання для витопарних кабелів. Однак у випадку застосування такого рішення висока ймовірність ушкодження важеля фіксуєної засувки при відключенні шнура від пасивного комутаційного й активного мережного устаткування. Ризик механічного ушкодження качани помітно зростає при великій кількості шнурових виробів. На сильно завантаженому комутаційному полі системний адміністратор, відключивши вилок, найчастіше просто витягає кабель із пучка таких же. При такому обігу зі шнуром вільний кінець важеля засувки, що виступає над корпусом, чіпляється за інші кабелі й легко обламується.

Найбільше часто проблема недостатньої механічної експлуатаційної надійності засувки усувається шляхом установки пружного захисного козирка, що нависає над вільним кінцем важеля засувки. Звичайно цей додатковий конструктивний елемент монтується на надільці, що перебуває на задній частині корпусу. Такий підхід зручний тим, що захисний козирок, на який можна нажати більшим пальцем, помітно полегшує відключення вилок.

Інші рішення по захисту – наприклад, досить популярні в мідножилній підсистемі вилок із двома додатковими виступами («рисячі вуха») – зустрічаються на практиці в одиничних випадках.

Відомо також зміна класичного дизайну качани й перехід на так зване заднє виконання важеля (англ. rear pivot latch), що вперше було запропоновано компанією Panduit. Суть рішення полягає в тім, що важіль прикріплений до задньої частини корпусу качани й спрямований своєю робочою частиною вперед. Від традиційної засувки залишений тільки невеликий передній напрямний носик,

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

що задає орієнтацію вилки й дозволяє вставити неї в розетку тільки в правильному положенні.

Окремо вкажемо на те, що застосування в конструкції вилки механізму виду push-pull частково вирішує проблему захисту важеля засувки від механічних ушкоджень. Це пов'язане із природним зменшенням величини або повним усуненням «сходи», що важільна засувка утворить у задній частині корпуса качани.

Компанія Fluke запропонувала незвичайне й дотепне рішення: експлуатаційна надійність симплексної вилки підвищується за рахунок переходу на ремонтпридатний дизайн фіксатора. Важіль засувки виконаний у вигляді окремого компонента, що вставлений у гніздо переднього напрямного виступу й утримується в робочому положенні тільки зусиллям, що притискає, пружини. Розбірна конструкція фіксатора допускає заміну важеля у випадку його механічного ушкодження.

Головним недоліком підходу Fluke є наявність відразу двох малогабаритних пружин, що обмежує область застосування модернізованої вилки: вона підходить лише для вимірювальної техніки.

Розетка зі змінною полярністю

Конструкція принаймні однієї розетки дозволяє міняти полярність у процесі експлуатації СКС. Суть нововведень полягає в тім, що в лицьовій пластині корпуса виробу виконаний другий, протилежний основному, виріз для важеля засувки (мал. 4). Завдяки цій особливості вилку можна вставляти в розетку у двох положеннях: прямому й зворотному.

Для усунення невизначеності щодо орієнтації вилки непотрібний виріз можна закрити знімною заглушкою. Для її установки передбачені монтажні гнізда на передній поверхні корпуса.

Перехідна розетка

Японська компанія Senko запропонувала оригінальне рішення по поліпшенню масогабаритних характеристик модулів, призначених для типових

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

слотових оптичних полиць СКС для ЦОД. Суть підходу полягає в тому, що для прикінцевування волокон усередині корпусу панелі використовується не стандартний LC, а коннектор Micro-LC. Незважаючи на назву, у нього дуже мало загального із прототипом, від якого фактично запозичений тільки наконечник, що centruє. Як фіксатор використовується байонетна гайка, а завдяки відсутності хвостовика загальна довжина качани скоротилася на 31 мм. Можливість установки тільки на волокна в буферному покритті 900 мкм не критична через спеціалізований характер виробу. Взаємодія з вилкою LC шнурового кабелю відбувається через перехідну розетку.

Модернізація елементів кріплення розеток

Модернізація розеток здійснюється з метою економії площі, займаної ними на лицьовій панелі групового комутаційного устаткування.

Основним засобом такої економії є зменшення кількості крапок кріплення, що дозволяє максимально зблизити окремі розетки й домогтися більшої щільності портів.

У випадку переходу на трирядну схему установки розеток щільність портів збільшується до 72/1U.

Слід зазначити, що не всі різновиди вилок LC сумісні з високоплотним комутаційним полем на основі модернізованих розеток. Найбільш універсальними в цьому випадку є качани з довгою тягою, що взаємодіє з важелем засувки за нижньою схемою. Скорочення висоти рознімання дозволяє зблизити ряди розеток по вертикалі й до 40% збільшити щільність конструкції за умови застосування полиць висотою 2U.

Модернізація розеточних модулів здійснюється із застосуванням двох різних підходів. У першому випадку розеточний модуль являє собою моноблочний виріб, а збільшення щільності конструкції досягається нарощуванням кількості одиночних розеточних гнізд без зміни числа крапок кріплення. Найбільшу популярність одержали зчетверені варіанти (CommScore, Leviton). Звертання до них дає можливість довести щільність дуплексних портів до 72/1U.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

Що стосується кількості окремих точок підключення в багатоволоконній розетці й напрямку орієнтації (пряма й кутова), те трохи більшу гнучкість дають конструктивно більше складні складальні варіанти, пропоновані, наприклад, японською компанією Senko. Такі вироби відносяться до групи стекових (stackable) рішень: вони являють собою конструктор, у якому є внутрішній або центральний, а також два зовнішніх (ліві й правий) варіанти розеток. Останні відрізняються наявністю виступу під кріпильні гвинти.

Формування групи розеток здійснюється за зразком системи «Лего», для чого передбачені спеціальні виступи й отвори. Механічна стабільність досягається шляхом установки розеток «у стек» одна на іншу довгою стороною корпусу. Лінійка виступів-отворів орієнтована перпендикулярно стосовно поздовжньої розетки (пряма установка) або під кутом 45° (кутова установка).

Специфічні умови експлуатації оптичної комутаційної техніки в ЦОД вплинули на характер змін, внесених у конструкцію з'єднувача. Через більше високу складність дизайну оптичних вилок саме вони найчастіше модернізуються з метою поліпшення споживчих якостей комутаційного устаткування для ЦОД.

Рознімання LC дозволяє формувати тракти передачі зі швидкістю до 200 Гбіт/с и в доступній для огляду перспективі буде, як колись, широко використовуватися при побудові інфраструктури ЦОД. Сучасні оптичні полки на його основі легко забезпечують щільність портів 72/1U.

3.2 Розробка структурної схеми

Структурована кабельна система (СКС) – це фізична основа інфраструктури будинку, що поєднує в єдиний комплекс різні мережні інформаційні сервіси.

Такими сервісами є:

- електросилова мережа й освітлення;
- комп'ютерна мережа;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

- IP-телефонія;
- система контролю й керування доступом (СКУД);
- система пожежної сигналізації й пожежогасіння;
- система охоронних датчиків;
- системи відеоспостереження;
- аудіосистема (системи оповіщення й радіо).

Принципи побудови СКС

Структурована кабельна система забезпечує передачу сигналів всіх типів і є основою локальної обчислювальної мережі (ЛОМ). ЛОМ поєднує телефони, комп'ютери й інше устаткування. Кожна точка підключення забезпечує доступ до всіх ресурсів мережі. Тому на кожному робочому місці досить двох ліній – комп'ютерної й телефонної. Ці лінії можуть бути взаємозамінні.

Нижче представлені основні принципи побудови СКС.

Структуризація

Кабельна проводка і її складові розбиваються на окремі підсистеми. Кожна підсистема виконує певні функції й має зв'язок з іншими підсистемами й мережним устаткуванням. Кожна підсистема має засоби перемикання, що дозволяють легко змінювати конфігурацію системи. При побудові системи можуть використовуватися різні види кабелю й комутаційного устаткування, залежно від умов конкретного проекту.

Універсальність

Кабельна система будується по принципах відкритої архітектури з технічними характеристиками, певними в стандартах. Параметри електричних і оптичних кабельних трас підсистем і їхніх інтерфейсів вказуються в нормативній документації. Таким чином, кабельна система може використовуватися для передачі сигналів різних додатків за допомогою кабелів усього двох типів – кручена пара й оптоволокно.

Комутація підсистем СКС один з одним і з активним мережним устаткуванням здійснюється певним набором шнурів з універсальними

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

розніманнями, що полегшує адміністрування кабельної системи й адаптацію її до різних додатків.

Надмірність

СКС передбачає можливість розширення – її топологія й устаткування забезпечують можливість збільшити кількість устаткування, що підключається, і обсяг трафіку. Все устаткування СКС вибирається з резервом по продуктивності, по можливості установки додаткових модулів і по розширенню функціональності.

Надійність

Виробники СКС гарантують працездатність і відповідність кабельної системи стандартам протягом усього терміну служби. У випадку аварії в СКС швидко локалізується несправна ділянка, виконується перехід на резервну лінію й проводяться ремонтні роботи. Відновлення роботи СКС здійснюється без зупинки роботи мережі силами адміністратора СКС, без залучення сторонніх фахівців.

Гнучкість

Функціонуюча СКС без зміни кабельної системи й без яких-небудь додаткових витрат надає наступні можливості:

- модифікація програмно-апаратного комплексу;
- керування переміщенням користувачів у будинку;
- зміна кількості користувачів;
- поділ користувачів на групи по різних ознаках.

Економічність

Великі первинні вкладення в СКС швидко окупаються за рахунок менших витрат на модифікацію й підтримку телекомунікаційної інфраструктури. Строк експлуатації СКС значно більше часу життя інших компонентів інформаційної системи (активне мережне устаткування, сервери й персональні комп'ютери, програмні засоби, телефонні станції й комунікаційне устаткування й т.д.).

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

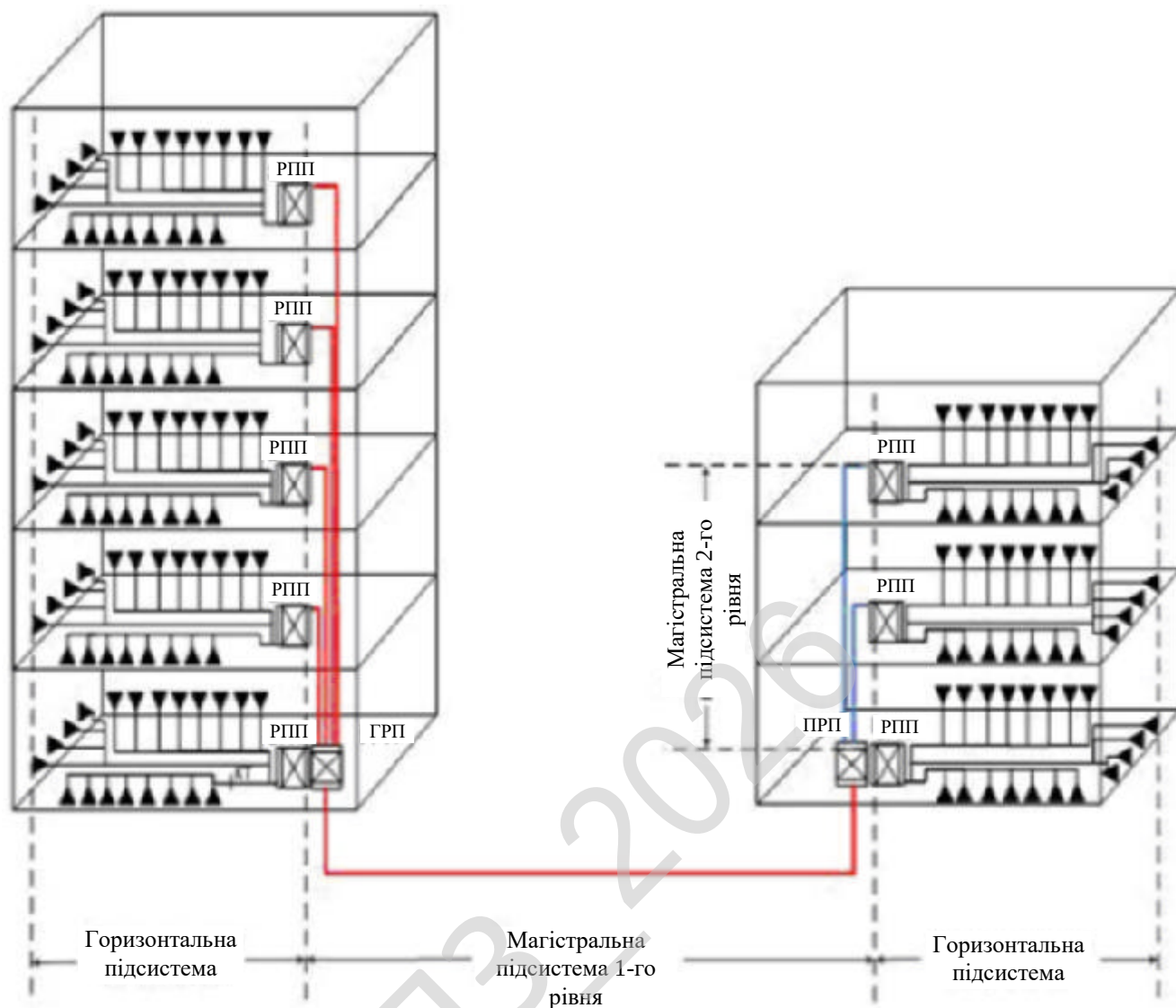


Рисунок 3.1 – Структурна схема системи

РПП – розподільний пункт поверху;

ГРП – головний розподільний пункт;

ПРП – проміжний розподільний пункт.

Довговічність

СКС забезпечить поступовий перехід до високошвидкісних протоколів, які будуть працювати на перспективу, простою заміною активного устаткування, при цьому не буде потрібно реконструкція кабельної системи. Технологічний запас характеристик і стандарти СКС гарантують, що моральне старіння

кабельної проводки трапляється не раніше строку, коли закінчиться її системна гарантія (у більшості виробників він становить 20 років).

Підсистеми СКС

СКС являє собою кабельну систему будинку (групи будинків) ієрархічного типу, що складається зі структурних підсистем. Кожна підсистема виконує свої функції, має певну топологію й состав компонентів. Для кожного типу підсистем у стандартах визначені вимоги, обмеження й правила.

Кабельні підсистеми СКС:

- магістральна підсистема I-го рівня;
- магістральна підсистема II-го рівня;
- горизонтальна підсистема.

Магістральна підсистема I-го рівня

Магістральна підсистема I-го рівня розташована між головним розподільним пунктом і проміжним розподільним пунктом, а також між головним розподільним пунктом і поверховим розподільним пунктом.

Дана підсистема включає:

- магістральні кабелі I-го рівня;
- розподільні пристрої, використовувані для магістрального кабелю I-го рівня;
- комутаційні перемички й шнури, які використовуються для комутації в головному розподільному пункті.

Магістральна підсистема II-го рівня

Проміжний розподільний пункт розділяє магістральну систему СКС на магістральну підсистему I-го рівня й магістральну підсистему II-го рівня.

Дана підсистема включає:

- магістральні кабелі II-го рівня;
- розподільні пристрої, використовувані для магістрального кабелю II-го рівня;

– комутаційні перемички й шнури, які використовуються для комутації в проміжній розподільній крапці.

Горизонтальна підсистема

Горизонтальна підсистема розташована від розподільних пристроїв поверхового розподільного пункту до телекомунікаційних розеток.

Дана підсистема включає:

- горизонтальні кабелі;
- розподільні пристрої, використовувані для горизонтальних кабелів;
- комутаційні перемички й шнури, які використовуються для комутації з поверховим розподільним пунктом;

- телекомунікаційні розетки;
- консолідаційні крапки.

Компоненти СКС

Структурована кабельна система створюється під час будівництва будинку або переустаткування приміщень і при цьому гарантований строк експлуатації становить не менш 10 років.

СКС включає наступне устаткування:

- телекомунікаційні шафи;
- серверні шафи;
- кабелі;
- дротові лотки;
- колона;
- патч-панелі;
- елетрощітки;
- коннектори;
- розетки (комп'ютерні, телефонні);
- телефонні плінти;
- крос-панелі.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

Архітектура СКС

Виділяють два типи архітектури СКС:

- розподілена;
- централізована.

Розподілена СКС

Розподілена архітектура найчастіше використовується для СКС багатоповерхових будинків і комплексів будинків. Розподілена архітектура може мати один або два рівні ієрархії. У першому випадку головний розподільний пункт з'єднаний з поверховим розподільним пунктом за допомогою кросу. У другому випадку СКС складається із трьох підсистем: магістральної I-го рівня, магістральної II-го рівня й горизонтальної підсистеми.

Переваги розподіленої архітектури:

- більша гнучкість СКС;
- простота розширення кабельної системи;
- простота установки кабельної системи.

Недоліки розподіленої архітектури:

- громіздкість кабельної системи (велика кількість компонентів);
- більша площа телекомунікаційних приміщень;
- складність здійснення контролю й забезпечення безпеки.

Централізована СКС

Централізована архітектура може мати один рівень ієрархії без крос-з'єднання горизонтальної або магістральної підсистеми або взагалі не мати рівнів ієрархії й складатися тільки з горизонтальної підсистеми.

Переваги централізованої архітектури СКС:

- невелика кількість компонентів кабельної системи;
- невелика площа телекомунікаційних приміщень;
- невелика кількість активного устаткування;
- відсутність активного й пасивного устаткування для організації магістралі;

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

- простота організації системи резервування активного устаткування.

Недоліки централізованої архітектури СКС:

- велика кількість кабелю;
- низька гнучкість кабельної системи;
- складність розширення СКС;
- складність монтажу;
- складність розмежування зони відповідальності в телекомунікаційному приміщенні при оренді будинку різними організаціями.

3.3 Розробка функціональної схеми

Перейдемо до розгляду функціональної схеми. Сформулюємо й вирішимо завдання аналізу функціональної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД. Для цього розроблена концепція проведення аналізу, вирішені завдання формального опису структури й обчислення її характеристик: навантаження на канали зв'язку, вузли й структуроутворююче устаткування системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД.

На рисунку 3.2 представлена функціональна схема системи, які відповідає запропонованому підходу.

Основною метою аналізу структури є визначення параметрів потоків даних, що проходять по каналах зв'язку системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД й вступні на вузли системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД. Однак, завдання структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД тільки як сукупності вузлів і зв'язків між ними, не дозволяє досліджувати потоки даних, оскільки потоки даних формуються розв'язуваними на системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД завданнями, точніше додатками, які запускаються на вузлах системи проектування топології СКС та

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

керування оптичною комунікаційною технікою ЦОД й обмінюються між собою даними. Отже, для аналізу системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД необхідно відомості про функціональну структуру доповнити відомостями про додатки і їхнє розміщення в системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД.

Результатами аналізу повинні стати математичні залежності, що дозволяють обчислювати чисельні значення характеристик системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД з урахуванням особливостей конкретної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД.

Запропоновано концептуальний підхід до аналізу функціональної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, заснований на дослідженні взаємодії додатків (завдань) як незалежних джерел і приймачів даних. У цьому випадку спочатку визначаються параметри потоків даних між додатками при виконанні всього комплексу завдань (будується інформаційна модель), а потім, залежно від розміщення додатків по вузлах системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД й використовуваного устаткування, визначаються параметри потоків даних між вузлами системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД (будується технічна модель).

При цьому не тільки повністю враховуються всі взаємодії між додатками, але й з'являється можливість проведення аналізу складних ієрархічних мережних структур, шляхом декомпозиції на підмережі системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, що часто застосовується в технологіях VLAN і VPN. Даний підхід розвивається й застосовується в даній роботі.

Відзначимо, що отримані результати аналізу потоків даних, можна надалі використовувати для проведення більше глибоких досліджень із застосуванням відомих методів і моделей, наприклад, СеМО.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

Для аналізу функціональної структури й розрахунку характеристик системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД визначені правила її формального опису, що однозначно задають властивості додатків і структуру системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, що дозволяють будувати моделі для проведення розрахунків. Відповідно до концептуального підходу до проведення аналізу функціональної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД виділені дві складові її структури інформаційна й технічна:

– Інформаційна структура визначає інформаційні потоки між інформаційними вузлами, на яких установлене програмне забезпечення й представляє сукупність вузлів і інформаційних ресурсів, розміщених на цих вузлах. Маючи у своєму розпорядженні дані про інформаційну структуру системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД можна приймати рішення про організацію каналів зв'язку між вузлами системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, визначати необхідні параметри телекомунікаційної системи, формувати структуру системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД.

– Технічна структура – сукупність структуроутворюючого устаткування, технічних вузлів системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД й каналів зв'язку. Технічному вузлу можуть відповідати декілька інформаційних.

Таким чином, для повноцінного аналізу функціональної структури реальної системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД необхідно провести аналіз складових її інформаційної й технічної структур і зв'язати результати аналізу.

Зв'язування результатів аналізу інформаційної й технічної структур має на увазі відображення характеристик інформаційної структури в характеристики технічної структури й визначення параметрів технічної структури на основі параметрів і характеристик інформаційної структури.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

Інформаційна структура системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД задається набором параметрів:

$$SI = \{N, M, D, L, R, S_k (k=1, 2, \dots, L), A_{km} (k=1, 2, \dots, L; m=1, 2, \dots, D), G, H, S\},$$

де:

- N – число працюючих користувачів;
- M – число задіяних вузлів;
- D – число використовуваних додатків;
- L – число розв'язуваних завдань;
- R – число баз даних;
- $S_k = \{p_k, d_k, u_k, W_k\}$, ($k=1, 2, \dots, L$) – набір даних, що описують

розв'язувані завдання, де:

- $p_k = (p_{k1}, p_{k2}, \dots, p_{kD})$, – вектор-рядок, що визначає завданням k додатка, що запускаються;

- $d_k = (d_{k1}, d_{k2}, \dots, d_{kR})$ – вектор-рядок, що визначає використовувані завданням k бази даних (сховища даних);

- $u_k = (u_{k1}, u_{k2}, \dots, u_{kN})$, – вектор-рядок, що визначає завданням k користувачів, що запускають $W_k = \|w_{kij}\|$ ($i=1, 2, \dots, D$; $j=1, 2, \dots, D$) – матриця, що встановлює послідовність запуску додатків завданням k ;

- $A_{km} = \{v_{km}, b_{km}\}$, ($k=1, 2, \dots, L$; $m=1, 2, \dots, D$) – набір даних, що описують додатки, використовувані завданнями, де:

- $v_{km} = (v_{km1}, v_{km2}, \dots, v_{kmR})$, – вектор-рядок, що задає обсяги даних, якими обмінюється додаток m з базами даних, при рішенні завдання k ;

- $b_{km} = (b_{km1}, b_{km2}, \dots, b_{kmD})$ – вектор-рядок, що задає обсяги даних, якими обмінюється додаток m з іншими додатками, при рішенні завдання k ;

- G матриця розміщення додатків по вузлах;

- H – матриця підключення користувачів до вузлів;

– $\mathbf{S}$ – матриця розміщення баз даних по вузлах.

Набір однозначно визначає інформаційну структуру системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД.

Для заданого набору $\mathbf{SI}$ отримані результати, що дозволяють визначити параметри потоків даних між вузлами системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД.

При цьому використовувалася матриця інтенсивностей потоків запитів користувачів на запуск завдань $\mathbf{\Lambda} = \|\lambda_{ij}\|, (i=1,2,\dots,N; j=1,2,\dots,L)$.

Показано, що матриця інтенсивностей потоків даних між вузлами системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД при рішенні завдання k обчислюється за формулою: $\mathbf{A}_k = \lambda_k \mathbf{Z}_k$, $(k=1,2,\dots,L)$, де $\mathbf{Z}_k$ – матриця обсягів даних, переданих між вузлами, при рішенні завдання. Обчислені також матриця інтенсивностей потоку запитів на запуск додатка номер j , на вузлі номер i – $\mathbf{B}^*$ і матриця сумарної інтенсивності потоку запитів до бази даних номер j , на вузлі номер i – Φ .

Таким чином, визначена множина параметрів потоків даних інформаційної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД:
 $\mathbf{PSI}(\mathbf{SI}) = \{\mathbf{\Lambda}, \mathbf{Z}_k, \mathbf{A}_k (k=1,2,\dots,L), \mathbf{A}, \mathbf{B}^*, \Phi\}$.

Розроблено моделі для аналізу ієрархічної трьохрівневої інформаційної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, що дозволяють визначити завантаження каналів зв'язку й мережного устаткування в системі проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, що складається із сукупності підмереж і корпоративних серверів.

Отримано наступні результати:

– матриці інтенсивностей потоків даних між групами й усередині груп

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

кожного рівня $A_1(C_1) = C_1 A(C_1)^T$, $A_2(C_2) = C_2 [A_1(C_1) - dg(A_1(C_1))](C_2)^T$;

– матриці інтенсивностей потоків даних утворених кожним завданням

$A_{1k}(C_1)$ і $A_{2k}(C_1)$, $(A = \sum_{k=1}^L A_k)$.

Досліджено властивості матриць, що відбивають взаємозв'язки потоків даних окремих завдань, сформульовані у вигляді доведених тверджень.

Як міра ефективності ієрархічної структури запропоновані коефіцієнти поглинання інтенсивностей потоків даних на кожному рівні. Коефіцієнт поглинання на кожному рівні відбиває частку сумарної інтенсивності потоків, що локалізується усередині рівня.

Результати аналізу інформаційної структури дозволяють визначати параметри потоків даних між логічними об'єднаннями вузлів системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД. Інформаційна структура реалізується конкретними технічними засобами й втілюється у вигляді технічної структури. При цьому можливо невідповідність інформаційної й технічної структур, пов'язане з технічними характеристиками й можливостями апаратури. Наприклад, на одному технічному вузлі можуть бути встановлені кілька інформаційних вузлів. У зв'язку із цим розроблені методи й засоби аналізу технічної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, створюваної на базі інформаційної структури.

Для аналізу роботи реальної системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД розроблений метод формального опису її технічної структури. Структура реальної системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД формується із застосуванням структуроутворюючого устаткування (комутатори, маршрутизатори), до якого підключаються вузли системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, при цьому створюється багаторівнева (часто трьохрівнева) мережа. Такий

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

підхід застосовується, як правило, при створенні системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД на базі технології VLAN. При цьому групи першого рівня інформаційної структури становлять віртуальні локальні системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД. Другий і третій рівні інформаційної структури призначені для з'єднання цих мереж між собою.

Число комутаторів, використовуваних для з'єднання вузлів технічної структури системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД при створенні груп першого рівня (комутатори першого рівня), визначається особливостями реальної системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД, технічними можливостями комутаторів. Позначимо це число K_1^* , ($K_1^* \geq K_1 \geq 1$). Число комутаторів, для з'єднання комутаторів першого рівня й створення груп другого рівня (комутатори другого рівня) – K_2^* , ($K_2^* \geq K_2 \geq 1$). Число комутаторів третього рівня для з'єднання комутаторів другого рівня – $K_3^* \geq 0$.

Технічна структура системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД задається множиною **ST**, елементами якого є:

– $Y_1^* = \|y_{1ij}^*\|$, ($i = 1, 2, \dots, M$; $j = 1, 2, \dots, K_1^*$) матриця з'єднань технічних вузлів системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД (робітники станції, сервери) з комутаторами першого рівня;

– $Y_2^* = \|y_{2ij}^*\|$, ($i = 1, 2, \dots, K_1^*$; $j = 1, 2, \dots, K_2^*$) матриця з'єднань комутаторів першого рівня з комутаторами другого рівня;

– $Y_3^* = \|y_{3ij}^*\|$, ($i = 1, 2, \dots, K_2^*$; $j = 1, 2, \dots, K_3^*$) матриця з'єднань комутаторів третього рівня з комутаторами другого рівня;

– $X_1^* = \|x_{1ij}^*\|$, ($i, j = 1, 2, \dots, K_1^*$) матриця з'єднань комутаторів першого рівня;

– $\mathbf{X}_2^* = \|\mathbf{x}_{2ij}^*\|$, $(i, j = 1, 2, \dots, K_2^*)$ матриця з'єднань комутаторів другого рівня;

– $\mathbf{X}_3^* = \|\mathbf{x}_{3ij}^*\|$, $(i, j = 1, 2, \dots, K_3^*)$ матриця з'єднань комутаторів третього рівня.

Для кожного рівня задані матриці, що задають пропускні здатності каналів зв'язку, використовуваних при побудові мереж:

$$\mathbf{C}_1^*(\mathbf{Y}_1^*), \mathbf{C}_2^*(\mathbf{Y}_2^*), \mathbf{C}_3^*(\mathbf{Y}_3^*), \mathbf{C}_1^*(\mathbf{X}_1^*), \mathbf{C}_2^*(\mathbf{X}_2^*), \mathbf{C}_3^*(\mathbf{X}_3^*).$$

Для заданої множини **ST** отримані наступні результати:

– матриця інтенсивностей інформаційних потоків, між комутаторами першого рівня й усередині груп технічних вузлів, підключених до комутаторів першого рівня: $\mathbf{A}_1^*(\mathbf{Y}_1^*) = (\mathbf{Y}_1^*)^T \mathbf{A} \mathbf{Y}_1^*$;

– матриця сумарних інтенсивностей інформаційних потоків для комутаторів другого рівня: $\mathbf{A}_2^*(\mathbf{Y}_2^*) = (\mathbf{Y}_2^*)^T [\mathbf{A}_1^*(\mathbf{Y}_1^*) - dg(\mathbf{A}_1^*(\mathbf{Y}_1^*))] \mathbf{Y}_2^*$;

– матриця інтенсивностей інформаційних потоків, між і усередині комутаторів третього рівня: $\mathbf{A}_3^*(\mathbf{Y}_3^*) = (\mathbf{Y}_3^*)^T [\mathbf{A}_2^*(\mathbf{Y}_2^*) - dg(\mathbf{A}_2^*(\mathbf{Y}_2^*))] \mathbf{Y}_3^*$;

– вектори інтенсивностей інформаційних потоків, що надходять на комутатори всіх рівнів: $\lambda_1^* = (\lambda_{11}^*, \lambda_{12}^*, \dots, \lambda_{1K_1^*}^*)$, $\lambda_2^* = (\lambda_{21}^*, \lambda_{22}^*, \dots, \lambda_{2K_2^*}^*)$, $\lambda_3^* = (\lambda_{31}^*, \lambda_{32}^*, \dots, \lambda_{3K_3^*}^*)$;

– вектор сумарних інтенсивностей потоків даних, переданих по каналах зв'язку $\gamma_i^* = (\gamma_{i1}^*, \gamma_{i2}^*, \dots, \gamma_{iM}^*)$, $(i = 1, 2, \dots, M)$.

Отримані також формули для обчислення параметрів потоків даних кожного завдання.



Рисунок 3.2 – Функціональна схема системи

Оскільки в корпоративних мережах великої розмірності використовується маршрутизація, то для обчислення параметрів потоків необхідно застосовувати відповідні алгоритми маршрутизації на графах. Для цього побудована матриця графа зв'язків між комутаторами Ω . Показано, що матриця Ω може бути представлена у вигляді:

$$\Omega = \begin{pmatrix} \mathbf{X}_1^* & \mathbf{Y}_2^* & \mathbf{0} \\ (\mathbf{Y}_2^*)^T & \mathbf{X}_2^* & \mathbf{Y}_3^* \\ \mathbf{0} & (\mathbf{Y}_3^*)^T & \mathbf{X}_3^* \end{pmatrix}.$$

Таким чином, параметри потоків даних у системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД для заданої технічної структури визначаються множиною:

$$\mathbf{PST}(\mathbf{ST}) = \{\mathbf{A}_1^*(\mathbf{Y}_1^*), \mathbf{A}_2^*(\mathbf{Y}_2^*), \mathbf{A}_3^*(\mathbf{Y}_3^*), \lambda_1^*, \lambda_2^*, \lambda_3^*, \gamma_1^*, \Omega\}.$$

Множина $\mathbf{PST}(\mathbf{ST})$ визначає параметри, як сумарних, так і часток потоків даних для окремих завдань (додатків), переданих по каналах зв'язку.

Якість рішення завдань залежить від того, яка частина пропускної здатності каналу (смуга) виділена для кожного завдання, що звичайно досягається шляхом застосування режиму гарантованої якості обслуговування (QoS).

Тому при рішенні завдань розрахунку потоків із застосуванням систем з гарантованою якістю обслуговування проводиться розширення множини $\mathbf{PST}(\mathbf{ST})$ шляхом додавання множини коефіцієнтів поділу каналів – $\mathbf{PKST}$. Це дозволяє визначити залежність характеристик системи проектування топології СКС та керування оптичною комунікаційною технікою ЦОД від смуги пропускання в каналі (комутаторі), що відводиться для завдання.

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

3.4 Розробка діаграми процесів

Розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3. Основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей.

Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників. Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування).

Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи.

Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграми потоків даних містять чотири типи елементів:

- Зовнішні по відношенню до системи сутності.
- Потоки даних між елементами трьох попередніх типів.
- Процеси які являють собою трансформацію даних в рамках описуваної системи.
- Сховища даних (репозиторії).



Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

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

4.1 Блок-схеми та опис алгоритмів функціонування системи

Розглянемо реалізацію бакалаврської дипломної роботи. Були проведені розрахунки і підібрані набори тестових даних для перевірки правильності реалізації проектних рішень.

Було створено блок-схеми роботи системи. Перед їх розглядом необхідно провести роз'яснення який саме тип блок-схем використовується. Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо.

Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними. У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач. Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування).

Всередині фігури записують безпосередньо самі операції. Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елемента.

Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елемента. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній). Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується.

Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи. Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі).

Всередині символу записується назва процесу і передані в нього дані. Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення).

Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи). З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми.

Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

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

Було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів; Діаграма компонент.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності.

Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку. Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

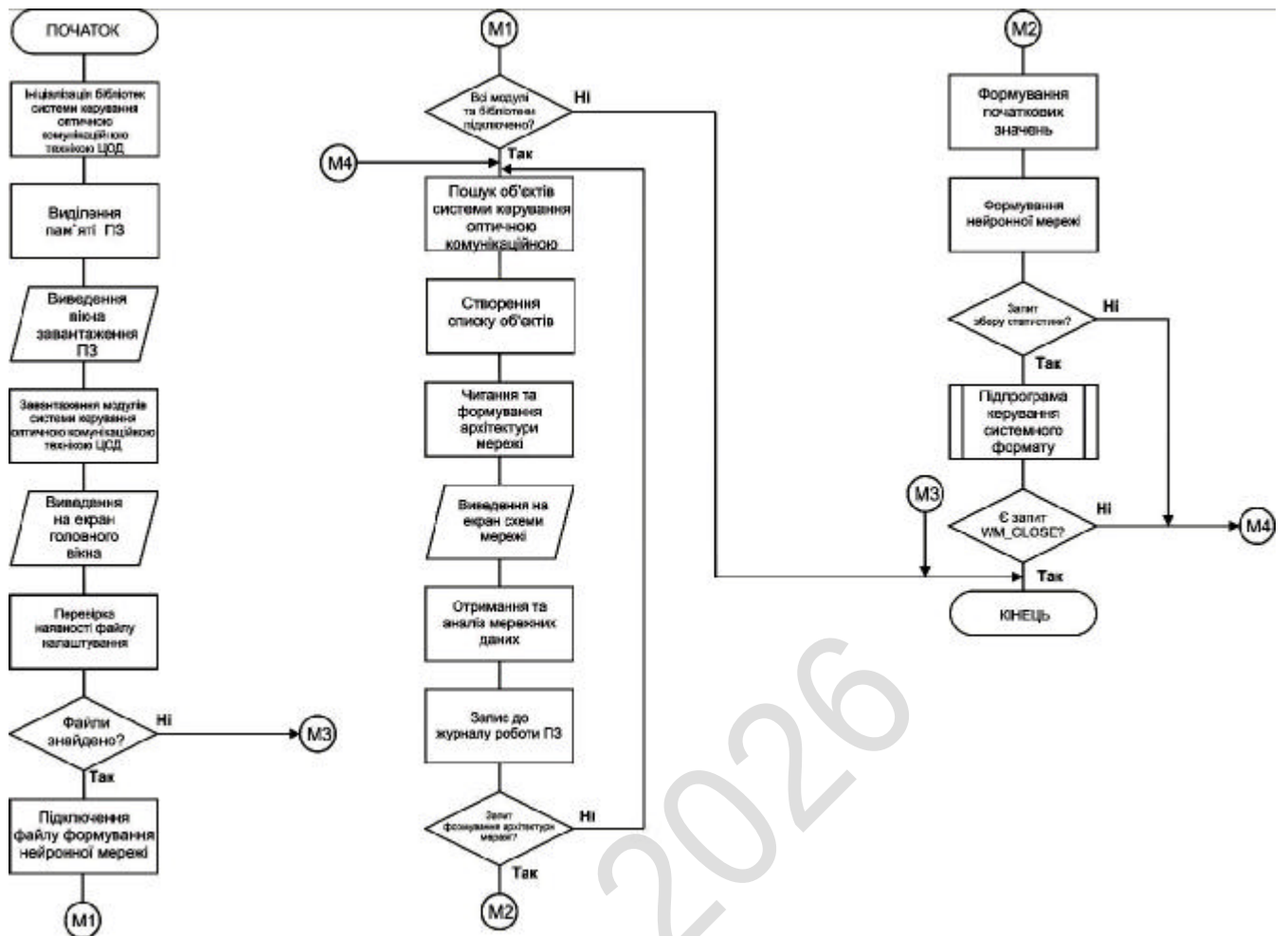


Рисунок 4.1 – Блок-схема основної програми

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

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

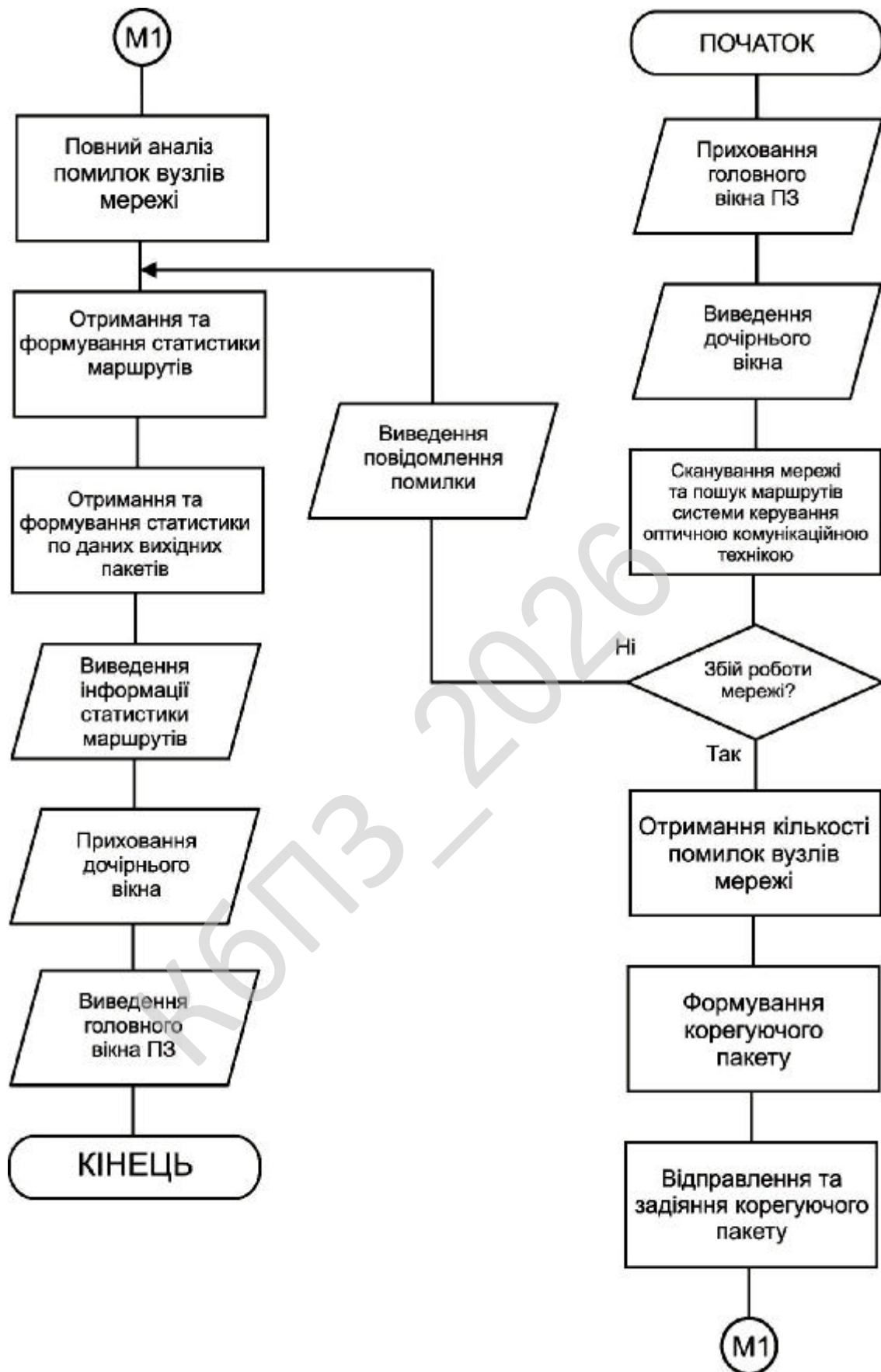


Рисунок 4.2 – Блок-схема роботи підпрограми

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

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

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

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

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

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

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

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

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

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

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

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко буває необхідним, створювати асоціації, що зв'язують відразу кілька класів. Графічно асоціація зображується у вигляді лінії, що з'єднує клас сам з собою або з іншими класами.

Асоціації може бути присвоєно ім'я, яке описує природу відносини. Зазвичай ім'я асоціації не вказується, якщо тільки ви не хочете явно задати для неї рольові імена або у вашій моделі настільки багато асоціацій, що виникає необхідність посилатися на них і відрізняти один від одного. Ім'я буде особливо корисним, якщо між одними і тими ж класами існує кілька різних асоціацій.

Клас, що бере участь в асоціації, грає в ній деяку роль. По суті, це "обличчя", яким клас, що знаходиться на одній стороні асоціації, звернений до класу з іншого її боку. Можна явно позначити роль, яку клас грає в асоціації.

Часто при моделюванні буває важливо вказати, скільки об'єктів може бути пов'язано допомогою одного примірника асоціації. Це число називається кратністю (Multiplicity) ролі асоціації та записується або як вираз, значенням якого є діапазон значень, або в явному вигляді.

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

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

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

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

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

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

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

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

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

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

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

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

Було використано MongoDB – документо-орієнтована система керування базами даних (СКБД) з відкритим вихідним кодом, яка не потребує опису схеми таблиць.

MongoDB займає нішу між швидкими і масштабованими системами, що оперують даними у форматі ключ/значення, і реляційними СКБД, функціональними і зручними у формуванні запитів.

Код MongoDB написаний на мові C++ і поширюється в рамках ліцензії AGPLv3.

MongoDB підтримує зберігання документів в JSON-подібному форматі, має досить гнучку мову для формування запитів, може створювати індекси для різних збережених атрибутів, ефективно забезпечує зберігання великих бінарних об'єктів, підтримує журналювання операцій зі зміни і додавання даних в БД, може працювати відповідно до парадигми Map/Reduce, підтримує реплікацію і побудову відмовостійких конфігурацій.

У MongoDB є вбудовані засоби із забезпечення шардінгу (розподіл набору даних по серверах на основі певного ключа), комбінуючи який з реплікацією даних можна побудувати горизонтально масштабований кластер зберігання, в якому відсутня єдина точка відмови (збій будь-якого вузла не позначається на

роботі БД), підтримується автоматичне відновлення після збою і перенесення навантаження з вузла, який вийшов з ладу.

Розширення кластера або перетворення одного сервера на кластер проводиться без зупинки роботи БД простим додаванням нових машин.

При розробці автори виходили з необхідності спеціалізації баз даних, завдяки чому їм вдалося відійти від принципу «один розмір під усе». За рахунок мінімізації семантики для роботи з транзакціями з'являється можливість вирішення цілого ряду проблем, пов'язаних з нестачею продуктивності, причому горизонтальне масштабування стає простішим. Використовувана модель документів зберігання даних (JSON/BSON) простіше кодується, простіше управляється (у тому числі за рахунок застосування так званого безсхемного стилю (schemaless style), а внутрішнє угруповання релевантних даних забезпечує додатковий виграш в швидкодії.

Нереляційний підхід досить зручний для створення баз даних, у яких горизонтальне масштабування означає розгортання на множині машин. Можливість забезпечувати найкращу продуктивність повинна існувати паралельно з підтримкою більшої функціональності, ніж це дозволяє використання пар «ключ-значення» (у чистому вигляді).

Технологія баз даних має працювати скрізь, починаючи з серверів користувача та віртуальних машин і закінчуючи хмарними технологіями.

MongoDB, на думку розробників, має заповнити розрив між простими сховищами даних типу «ключ-значення» (швидкими і легко масштабованими) і великими СКБД (зі структурними схемами і потужними запитами).

Основні можливості MongoDB:

1. Документо-орієнтоване сховище (проста та потужна JSON-подібна схема даних).
2. Досить гнучка мова для формування запитів.
3. Динамічні запити.
4. Повна підтримка індексів.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

5. Профілювання запитів.
7. Швидкі оновлення «на місці».
8. Ефективне зберігання бінарних даних великих обсягів, наприклад, фото та відео.
9. Журналювання операцій, що модифікують дані в БД.
10. Підтримка відмовостійкості і масштабованості: асинхронна реплікація, набір реплік і шардінг.
11. Може працювати відповідно до парадигми MapReduce.

СКБД управляє наборами JSON-подібних документів, що зберігаються в бінарному форматі BSON. Зберігання і пошук файлів в MongoDB відбувається завдяки викликам протоколу GridFS.

Подібно до інших документо-орієнтованих СКБД (CouchDB тощо), MongoDB не є реляційною СКБД.

Є докладна і якісна документація, велике число прикладів і драйверів для популярних мов Java, C++, C#, PHP, Python, Perl, Ruby.

При випуску одразу було заявлено, що реліз MongoDB 1.0 готовий до використання у виробництві як одиничний хост, так і у зв'язках master/slave.

Код цього релізу досить стабільний і перевірений в промисловій експлуатації протягом 1,5 років. MongoDB рекомендується розгортати мінімум на двох серверах використовуючи реплікацію Master/Slave.

Це забезпечує наявність актуальних даних при виході з ладу однієї з СКБД. MongoDB – продукт досить молодий, і відтак у ньому зустрічаються помилки, з'являються нові можливості тощо. Характерний високий темп розробки (проект пишуть не тільки волонтери, а й компанія людей на повній зайнятості). Компанія-розробник надає платні підтримку, хостинг, консультації.

Приклади запитів

Запити можуть витягати дані з вбудованих об'єктів та масивів. Якщо в колекцію users вставлений такий об'єкт:

```
{
  "username" : "bob",
```

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

```

    "address" : {
      "street" : "123 Main Street",
      "city" : "Springfield",
      "state" : "NY"
    }
  }

```

Ми можемо запитати цей документ (і всі документи з адресою в Нью-Йорку) за допомогою:

```
> db.users.find({"address.state" : "NY"})
```

Можна також запитати елементи масиву:

```
> db.food.insert({"fruit" : ["peach", "plum", "pear"]})
```

```
> db.food.find({"fruit" : "pear"})
```

Система керування базами даних (СКБД, Database Management System, DBMS) – набір взаємопов'язаних даних (база даних) і програм для доступу до цих даних. Надає можливості створення, збереження, оновлення та пошуку інформації в базах даних з контролем доступу до даних.

Першим поколінням СКБД прийнято вважати ієрархічні й мережеві системи. Ці системи отримали широке поширення в 1970-х роках, а першою комерційною системою цього типу була система IMS компанії IBM.

У 1980-х роках ці системи були витіснені системами другого покоління – повсюдно використовуваними і донині реляційними СКБД. У цих системах використовувалися непроцедурні мови управління даними (SQL) і передбачався значний ступінь незалежності даних. Реляційні системи внесли значні удосконалення в управління даними: графічний користувацький інтерфейс (GUI), клієнт-серверні застосунки, розподілені бази даних, паралельний пошук даних та інтелектуальний аналіз даних.

Але вже до кінця 1980-х років існуюча тоді реляційна модель перестала задовольняти розробників в силу низки обмежень. Відповіддю на зростаючу складність програм баз даних стали два нових напрямки розвитку СКБД: об'єктно-орієнтовані СКБД і об'єктно-реляційні СКБД.

У 1991 був утворений консорціум ODMG, основною метою якого стало вироблення промислового стандарту об'єктно-орієнтованих баз даних. Між 1993

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

та 2001 роками ODMG опублікувала п'ять ревізій своїх специфікацій. Остання версія стандарту має індекс 3.0, після чого група розпустилася. До кінця 1990-х існувало близько десяти компаній, що виробляли комерційні продукти, що позиціонуються на ринку як ООСКБД. Найбільш відомими системами даного класу стали Objectivity, Versant виробництва однойменних компаній, а також СКБД Jasmine, випущена компанією CA. Незважаючи на переваги, що дозволяють ефективніше вирішувати певний ряд завдань, об'єктно-орієнтовані системи так і не змогли завоювати значущу частку ринку СКБД, залишившись «нішевим» продуктом.

Постачальниками традиційних реляційних СКБД також була проведена значна робота з об'єднання об'єктно-орієнтованих і реляційних систем. Розробники постаралися розширити мову SQL, щоб включити в неї концепції об'єктно-орієнтованого підходу, зберігаючи переваги реляційної моделі (об'єктні розширення мови SQL були зафіксовані в стандарті SQL:1999).

Основний принцип – це еволюційний розвиток можливостей СКБД без поломки попередніх підходів та зі збереженням наступності з системами попереднього покоління.

Поняття СКБД третього покоління, якими, власне кажучи, і є об'єктно-реляційні СКБД, з'явилося після опублікування групою відомих фахівців в області баз даних «Маніфесту систем баз даних третього покоління». Основні принципи СКБД третього покоління, позначені в маніфесті:

Крім традиційних послуг з управління даними, СКБД третього покоління повинні забезпечити підтримку розвиненіших структур об'єктів і правил. Розвинутіша структура об'єктів характеризує засоби, необхідні для зберігання і маніпулювання нетрадиційними елементами даних (тексти, просторові дані, мультимедіа).

СКБД третього покоління повинні включити в себе СКБД другого покоління. Системи другого покоління внесли вирішальний вклад у двох областях – непроцедурний доступ за допомогою мови запитів SQL і незалежність

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

даних. Ці досягнення обов'язково повинні враховуватися в системах третього покоління.

СКБД третього покоління повинні бути відкриті для інших підсистем. Це включає оснащення різноманітними інструментами підтримки прийняття рішень, доступом з багатьох мов програмування, інтерфейсами до існуючих популярних систем і бізнес-застосунків, можливістю запуску програм з бази даних на іншій машині і розподілені СКБД. Весь набір інструментів і СКБД має ефективно функціонувати на різноманітних апаратних платформах з різними операційними системами.

Крім того, СКБД, що розраховує на широку сферу застосування, повинна бути оснащена мовою четвертого покоління (4GL).

У середині 1990 років було лише кілька дослідних прототипів СКБД, які поєднали найкращі риси реляційних і об'єктно-орієнтованих СКБД. Першим комерційним продуктом, якому були властиві об'єктно-реляційні риси, став Universal Server компанії Informix(згодом була поглинена IBM). В даний час більшість цих ідей вже втілено в реальних комерційних рішеннях, в тому числі і в продуктах основних постачальників СКБД (Oracle Database і IBM DB2).

Розвиток індустрії систем керування базами даних базується на значних фундаментальних наукових дослідженнях. Найчастіше, між самими дослідженнями та їхньою конкретною реалізацією в прикладних рішеннях минають роки, а іноді й десятиліття. Роботу в області управління даними проводять як університетські дослідницькі групи (MIT, Berkeley), так і центри розробок основних постачальників СКБД (Oracle, IBM, Microsoft). Інвестування в управління даними – це довгострокове, і разом з тим, вигідне вкладення коштів. В даний час дослідники мають у своєму розпорядженні засоби, що дозволяють ефективно реалізувати найскладніші запити, що маніпулюють терабайтами й петабайтами різних даних.

Основними тенденціями, які дали привід для проведення різних масштабних досліджень в області баз даних стали:

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

Експонентний ріст даних. Обсяг даних, у тому числі синтетичних, що генеруються автоматизованими системами, значно зріс. Збільшилося і число прикладних областей, в яких вимагається обробка великих обсягів даних. До таких областей тепер відносяться не тільки традиційні корпоративні програми, пошук у веб, але також і наукові дослідження, обробка природних мов, аналіз соціальних мереж тощо.

Значне ускладнення структур використовуваних даних. Прості види даних у вигляді чисел і символьних рядків стали доповнятися численною мультимедійною інформацією, просторовими, процедурними даними та великою кількістю інших складних форматів.

Широке поширення дешевих високопродуктивних апаратних засобів. Щорічно ми спостерігаємо зростання обчислювальних можливостей мікропроцесорів, збільшення ємності і зниження вартості доступних і зручних в експлуатації пристроїв дискової оперативної пам'яті.

Активний розвиток засобів комунікації та «всесвітньої павутини» World Wide Web. WWW стає єдиним інформаційним середовищем, що пронизує весь світ і об'єднує величезне число користувачів та електронних пристроїв.

Поява нових важливих областей застосування СКБД. У першу чергу, це пов'язано з інтелектуальним аналізом даних, сховищами даних, а останнім часом – з паралельними обчисленнями і хмарними технологіями.

Основні характеристики СКБД

1. Контроль за надлишковістю даних.
2. Несуперечливість даних.
3. Підтримка цілісності бази даних (коректність та несуперечливість).
4. Цілісність описується за допомогою обмежень.
5. Незалежність прикладних програм від даних.
6. Спільне використання даних.
7. Підвищений рівень безпеки.

Можливості СКБД

1. Дозволяється створювати БД (здійснюється за допомогою мови визначення даних DDL (Data Definition Language)).

2. Дозволяється додавання, оновлення, видалення та читання інформації з БД (за допомогою мови маніпулювання даними DML, яку часто називають мовою запитів)

3. Можна надавати контрольований доступ до БД за допомогою: Системи забезпечення захисту, яка запобігає несанкціонованому доступу до БД; Системи керування паралельною роботою прикладних програм, яка контролює процеси спільного доступу до БД; Система відновлення – дозволяє відновлювати БД до попереднього несуперечливого стану, що був порушений в результаті збою апаратного або програмного забезпечення.

Основні компоненти середовища СКБД

1. Апаратне забезпечення.

2. Програмне забезпечення.

3. Дані.

4. Процедури – інструкції та правила, які повинні враховуватись при проектуванні та використанні БД.

5. Користувачі: адміністратори даних (керування даними, проектування БД, розробка алгоритмів, процедур) та БД (фізичне проектування, відповідальність за безпеку та цілісність даних); розробники БД; прикладні програмісти; кінцеві користувачі.

Архітектура СКБД

Існує трирівнева система організації СКБД ANSI-SPARC, при якій існує незалежний рівень для ізоляції програми від особливостей представлення даних на нижчому рівні.

Рівні:

1. Зовнішній – представлення БД з точки зору користувача.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

2. Концептуальний – узагальнене представлення БД, описує які дані зберігаються в БД і зв'язки між ними. Підтримує зовнішні представлення, підтримується внутрішнім рівнем.

3. Внутрішній – фізичне представлення БД в комп'ютері.

Логічна незалежність – повна захищеність зовнішніх моделей від змін, що вносяться в концептуальну модель.

Фізична незалежність – захищеність концептуальної моделі від змін, які вносяться у внутрішню модель.

Подійно-орієнтована архітектура (Event-driven architecture, надалі EDA) – шаблон архітектури програмного забезпечення, який призначений для створення подій, їх виявлення, споживання і реагування на них.

Подія може бути визначена як значна зміна стану. Наприклад, коли споживач купує автомобіль, стан автомобіля змінюється з "на продаж" до "продано". Архітектура системи дилера автомобілів може трактувати цю зміну стану як подію, поява якої може стати відомою іншим програмам даної архітектури.

З формальної точки зору, те, що виробляється, публікується, поширюється, виявляється і споживається (як правило, асинхронно) є повідомленням, яке називають сповіщенням про подію (або нотифікацією), а не самою подією, яка є зміною стану, що викликає появу повідомлення.

Події не подорожують, вони просто відбуваються. Проте термін подія часто використовується метонімічно для позначення самого нотифікаційного повідомлення, що може призвести до певної плутанини.

Цей архітектурний шаблон може застосовуватися при проектуванні і реалізації ПЗ і систем, які передають події між слабкозв'язаними компонентами програмного забезпечення і сервісами (службами).

Подійно-орієнтована система як правило складається з емітерів подій (або агентів) і споживачів подій (або стоків). Стоки несуть відповідальність за здійснення реагування на появу події. Реакція не завжди може бути повністю

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

забезпечена самим стоком. Наприклад, стік, може бути відповідальним лише за фільтрацію, трансформацію і відправку події до іншого компонента або він може забезпечити повністю самостійну реакцію на таку подію. Перша категорія стоків може бути заснована на традиційних компонентах, таких як проміжне програмне забезпечення, орієнтоване на обробку повідомлень (message oriented middleware, MOM), в той час, як друга категорія стоків (самостійна реакція в режимі он-лайн) може вимагати більш придатної платформи (фреймворку) для виконання транзакцій.

Розробка ПЗ і систем в подійно-орієнтованій архітектурі дозволяє їм бути сконструйованими способом, який більш відповідає вимогам до їх створення, оскільки такі системи в більшій мірі пристосовуються до непередбачуваних і асинхронних середовищ.

Подійно-орієнтована архітектура (EDA) може доповнювати сервісно-орієнтовану архітектуру (SOA), оскільки сервіси (служби) можуть бути активовані тригерами, які ініціюються при настанні подій.

Ця парадигма особливо корисна, коли стік не забезпечує власного виконання будь-яких дій.

Подійно-орієнтована SOA (SOA-2) розвиває архітектури SOA і EDA для забезпечення більш глибокого і надійного рівня сервісів за рахунок використання раніше невідомих причинно-наслідкових зв'язків, щоб сформувати новий шаблон подій. Цей новий шаблон бізнес-аналітики дає поштовх до небаченого раніше зростання рівня автоматизації підприємства за рахунок привнесення додаткової цінної інформації в описану раніше модель діяльності.

Обчислювальна техніка та сенсорні пристрої (сенсори, датчики, контролери) можуть виявляти зміни стану об'єктів або умов і створювати події, які потім можуть бути оброблені сервісом (службою) або системою.

Подія може складатися з двох частин: заголовка події та тіла події. Заголовок події може включати в себе інформацію таку як, наприклад, назва події, часова мітка події і тип події. Тіло події – це частина, яка описує факт, що стався в

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

дійсності. Тіло події не слід плутати з шаблоном або логікою, яка може бути застосована як реакція на саму подію.

Архітектура, керована подіями, складається з чотирьох логічних рівнів (шарів). Вона починається з виявлення факту, його технічного подання у формі події і закінчується непустою множиною реакцій на цю подію.

Генератор подій.

Першим логічним шаром є генератор подій, який виявляє факт і представляє цей факт подією. Оскільки фактом може бути практично все, що може бути сприйнято, то ним може бути і генератор подій. Наприклад, генератором може бути клієнт електронної пошти, система електронної комерції або певний тип датчика.

Перетворення різних даних, отриманих від датчиків, в єдину стандартизовану форму даних, які можуть бути оцінені, є основною проблемою при розробці та реалізації цього шару. Однак, враховуючи, що подія є строго декларативною, можна легко застосовувати будь-які операції трансформації, тим самим усуваючи необхідність забезпечення високого рівня стандартизації.

Канал подій.

Канал подій – це механізм, через який інформація від генератора подій передається до обробника подій (event engine) або стоку.

Це може бути з'єднання TCP/IP або вхідний файл будь-якого типу (простий текст, формат XML, e-mail тощо). В один і той же час може бути відкрито кілька каналів подій. Як правило, оскільки обробник подій повинен працювати в режимі, наближеному до реального часу, канали подій зчитуються асинхронно. Події зберігаються в черзі, очікуючи наступної обробки механізмом обробки подій.

Механізм обробки подій.

Механізм обробки подій (event processing engine) є місцем, де подія ідентифікується і вибирається відповідна реакція на нього, яка потім виконується. Це також може призвести до породження ряду тверджень. Якщо подія, яка надійшла до механізму обробки подій, є наприклад такою «Запаси продукту ID

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

досягли нижнього допустимого рівня», це може ініціювати, наприклад, такі реакції як «Замовити продукт ID» і «Сповістити персонал».

Наступна подійно-орієнтована дія (післядія).

Щодо того, як можуть проявлятися наслідки події, слід відмітити, що вони можуть проявитись багатьма різними способами і у різноманітних формах (наприклад, повідомлення електронної пошти, надіслане комусь, або ПЗ, що виводить деяке попередження на екран). Залежно від рівня автоматизації, який забезпечується стоком (механізмом обробки подій), ці дії можуть виявитись зайвими.

Є три основні стилі обробки подій: простий, потоковий і складний. Часто ці три стилі використовуються спільно у розвинутій подійно-орієнтованій архітектурі.

Проста обробка подій.

Проста обробка подій стосується подій, які безпосередньо належать до специфічних вимірних змін умов. У випадку простої обробки подій, мають справу з появою відомих подій, що ініціюють післядію (післядії). Проста обробка подій зазвичай використовується для управління потоком робіт в реальному часі, скорочуючи тим самим час затримки і вартість робіт.

Наприклад, прості події можуть створюватись (породжуватись) датчиком, що виявляє зміну тиску в шині або температуру навколишнього середовища.

Обробка потоку подій.

При обробці потоку подій (event stream processing, далі ESP) відбуваються як звичайні, так і відомі події. Звичайні події (заявки, передачі RFID) перевіряються на те, чи є вони відомими, і передаються інформаційним передплатникам. Обробка потоку подій зазвичай використовується для управління потоком інформації в реальному часі і на рівні підприємства, що дозволяє своєчасно приймати рішення.

Обробка складних подій (Complex event processing (CEP)) дозволяє за шаблонами простих і звичайних подій проводити аналіз того, чи наступила

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

складна подія. Обробка складних подій полягає в оцінюванні взаємного впливу подій і в наступному виконанні дій. При цьому, типи подій (відомих або звичайних) можуть перетинатись, а події можуть виникати протягом тривалого періоду часу.

Кореляція подій може бути причинною, тимчасовою або просторовою. СЕР вимагає використання складних інтерпретаторів подій, визначення і підбору шаблонів подій, а також відповідних кореляційних методів. Обробка складних подій зазвичай використовується для виявлення і реагування на аномальну поведінку, загрози і можливості у бізнесі.

Управління вимогами це процес запису, аналізу, трасування, пріоритезації і узгодження вимог та контролю змін і доведення до їх зацікавлених сторін. Це безперервний процес протягом всього життя проекту. Вимога – якість, якій мають відповідати результати проекту (продукту або послуги).

Мета управління вимогами полягає в тому, щоб переконатися, що організація відповідає потребам і очікуванням своїх клієнтів, внутрішніх або зовнішніх зацікавлених сторін. Управління вимогами починається з аналізу і виявлення цілей і обмежень організації. Управління вимогами додатково включає в себе підтримку планування вимог, інтеграції вимог і організації роботи з ними (атрибути для вимог).

Управління вимогами передбачає спілкування між членами проектної групи і зацікавленими сторонами, і адаптацію до змін у вимогах протягом всього проекту. Щоб запобігти перетину поля одного класу вимог з іншим, постійні зв'язки між членами команди розробників є критичними. Наприклад, при розробці програмного забезпечення для внутрішнього використання у бізнесу можуть бути настільки сильні потреби, що він може проігнорувати вимоги користувачів, або вважати, що створені сценарії використання покривають також і користувацькі вимоги.

Відслідковування вимоги фактично означає документування всього життєвого циклу вимоги. Часто необхідно дізнатися першоджерело кожної

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

вимоги. Для цього всі зміни вимог повинні бути задокументовані, щоб досягти стану повного відстеження. Відстежувати треба бути навіть використання реалізованих вимог.

Вимоги мають різні джерела, такі як ділова людина, що замовляє продукт, менеджер зі збуту і фактичний користувач. У всіх цих людей є різні вимоги до продукту. Використовуючи відслідковування вимог, реалізована в системі функція може бути простежена назад до людини або групі, яка замовляла її під час збору вимог. Ця особливість може, наприклад, використовуватися в процесі розробки для пріоритезації вимог, визначаючи, наскільки цінною є дана вимога для певного користувача.

Відслідковування може також використовуватися після розгортання продукту. Наприклад, коли вивчення використання системи показує, що якась функція не використовується, можна визначити навіщо вона була потрібна спочатку.

Завдання управління вимогами

На кожному етапі процесу розробки існують ключові методи і задачі пов'язані з управлінням вимогами. Для ілюстрації, розглянемо наприклад стандартний процес розробки з п'ятьма фазами: дослідженням, аналізом здійсненності, дизайном, розробкою та тестуванням і випуском.

Дослідження. Під час фази дослідження збираються перші три класи вимог від користувачів, бізнесу і команди розробників. У кожній області задають однакові питання: які цілі, які обмеження, які використовуються процеси та інструменти і так далі. Тільки коли ці вимоги добре зрозумілі, можна приступати до розробки функціональних вимог.

Тут необхідне застереження: незалежно від того, як сильно група намагається це зробити, вимоги не можуть бути повністю визначені на початку проекту. Деякі вимоги змінюються, або тому що вони просто не були знайдені спочатку, або тому що внутрішні чи зовнішні сили торкаються проекту в середині циклу. Таким чином, учасники групи повинні спочатку погодитися, що головна

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

умова успіху – гнучкість у мисленні та діях. Результатом стадії дослідження є документ – специфікація вимог, схвалений усіма членами проекту. Пізніше, в процесі розробки, цей документ буде важливий для запобігання розповзання меж проекту або непотрібних змін. Оскільки система розвивається, кожна нова функція відкриває світ нових можливостей, таким чином специфікація вимог прив'язує команду до оригінального бачення системи і дозволяє контрольоване обговорення змін.

У той час як багато організацій все ще використовують звичайні документи для керування вимогами, інші управляють своїми базовими вимогами, використовуючи програмні інструменти.

Ці інструменти керують вимогами використовуючи базу даних, і зазвичай мають функції автоматизації відстеження (наприклад, дозволяючи створювати зв'язки між батьківськими і дочірніми вимогами, або між тестами і вимогами), управління версіями, і управління змінами. Зазвичай такі інструментальні засоби містять функцію експорту, яка дозволяє створювати звичайний документ, екпортуючи дані вимог.

Аналіз здійсненості

На стадії аналізу здійсненості визначається вартість вимог. Для користувальницьких вимог поточна вартість роботи порівнюється з майбутньою вартістю встановленої системи. Задаються питання такі як: «Скільки нам зараз варті помилки введення даних?» Або, «Яка вартість втрати даних через помилки оператора пов'язаної з використанням інтерфейсом?». Фактично, потреба в новому інструменті часто розпізнається, коли подібні питання потрапляють до уваги людей, що займаються в організації фінансами.

Ділова вартість включає відповіді на такі питання як: «У якого відділу є бюджет на це?» «Який рівень повернення коштів від нового продукту на ринку?» «Який рівень скорочення внутрішніх витрат на навчання і підтримку, якщо ми зробимо нову, більш просту в використанні систему?». Технічна вартість пов'язана з вартістю розробки програмного забезпечення та апаратною вартістю.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

«Чи є у нас потрібні люди, щоб створити інструмент?» «Чи потребуємо ми нове устаткування для підтримки нової системи?»

Подібні питання дуже важливі. Група повинна з'ясувати, чи буде новий автоматизований інструмент мати достатню ефективність аби перенести частину тягара користувачів на систему і зекономити час людей.

Ці питання також вказують на основну суть управління вимогами. Людина і інструмент формують систему, і це розуміння особливо важливе, якщо інструмент – комп'ютер або новий додаток на комп'ютері. Людський розум вкрай ефективний у паралельній обробці та інтерпретації тенденцій з недостатніми даними. Комп'ютерний процесор ефективний у послідовній обробці і точному математичному обчисленні. Основна мета управління вимогами для програмного проекту полягала б у тому, щоб гарантувати, що автоматизована робота призначена «правильному» процесору.

Наприклад, «не змушуйте людину пам'ятати, де вона знаходиться в системі. Примусьте інтерфейс завжди повідомляти про місцезнаходження людини в системі». Або «не змушуйте людини вводити ті ж самі дані в два екрани. Примусьте систему зберігати дані і заповнювати їх де необхідно автоматично». Результатом стадії аналізу здійсненності є бюджет і графік проекту.

Дизайн. Припускаючи, що вартість точно визначена і переваги, які будуть отримані, є досить великими, проект може перейти до стадії проектування.

На стадії дизайну основна діяльність управління вимогами полягає в тому, щоб перевіряти чи відповідають результати дизайну документу вимог, щоб упевнитися, що робота залишається в межах проекту.

І знову, гнучкість є ключем до успіху. Ось класичний приклад змін проекту, які відмінно працювали. Проектувальники Форда на початку 1980-х очікували, що ціни на бензин піднімуться до 3,18 дол за галон до кінця десятиліття. На середині процесу дизайну автомобіля Ford Taurus, ціни встановилися приблизно на рівні 1,50 дол за галон. Колектив дизайнерів вирішив, що вони могли б створити більший, більш зручний, і більш потужний автомобіль,

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		74

якщо б ціни на бензин залишилися низькими. Таким чином, вони перепроєктувати автомобіль. Коли новий автомобіль вийшов, він встановив загальнонаціональні рекорди продажів.

У більшості випадків, однак, відступ від оригінальних вимог до такої міри не працює. Таким чином документ вимог стає ключовим інструментом, який допомагає команді приймати рішення про зміни дизайну.

Розробка та тестування. На стадії розробки і тестування, основна діяльність управління вимогами – це гарантувати, що робота і ціна залишаються в межах графіка і бюджету, і що створюваний інструмент дійсно відповідає вимогам. Основним інструментом, використовуваним на цій стадії, є створення прототипу і ітераційне тестування. Для програмного додатка користувацький інтерфейс може бути створений на папері і перевірений з потенційними користувачами, в той час як створюється основа програми. Результати цих тестів записуються в керівництві по дизайну користувацького інтерфейсу і передаються колективу дизайнерів. Це економить їх час і робить їх завдання набагато простіше.

При розробці ПЗ було використано V-Model (або VEE модель) є моделлю розробки інформаційних систем (ІС), спрямованої на спрощення розуміння складнощів, пов'язаних з розробкою систем. Вона використовується для визначення єдиної процедури розробки програмного забезпечення, апаратного забезпечення та людино-машинного інтерфейсу.

Концепція V-подібної моделі була розроблена Німеччиною та США в кінці 1980-х років незалежно один від одного:

– Німецька V-модель була розроблена аерокосмічної компанією IABG в Оттобрунні поряд з Мюнхеном у сприянні з Федеральним департаментом з закупівлі озброєнь в Кобленці, для Міністерства оборони Німеччини. Модель була прийнята німецькою федеральною адміністрацією для цивільних потреб влітку 1992.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

– Американська V-Model (VEE) була розроблена національною радою з системної інженерії (міжнародна – з 1995 року) для супутникових систем, включаючи обладнання, програмне забезпечення та взаємодію з користувачами.

Сучасною версією V-Model є V-Model XT, яка була затверджена в лютому 2005 року. V-модель використовується для управління процесом розробки програмного забезпечення для німецької федеральної адміністрації.

Зараз вона є стандартом для німецьких урядових і оборонних проектів, а також для виробників ПЗ в Німеччині. V-Model являє собою скоріше набір стандартів у галузі проектів, що стосуються розробки нових продуктів. Ця модель багато в чому схожа з Prince2 і описує методи як для проектного управління, так і для системного розвитку.

Основні принципи

Основний принцип V-подібної моделі полягає в тому, що деталізація проекту зростає при русі зліва направо, одночасно з плином часу, і ні те, ні інше не може повернути назад. Ітерації в проекті виробляються по горизонталі, між лівою і правою сторонами літери.

Стосовно до розробки інформаційних систем V-Model – варіація каскадної моделі, в якій завдання розробки йдуть зверху вниз по лівій стороні букви V, а завдання тестування – вгору по правій стороні букви V. Усередині V проводяться горизонтальні лінії, що показують, як результати кожної з фаз розробки впливають на розвиток системи тестування на кожній із фаз тестування.

Модель базується на тому, що прийнятно-здавальні випробування ґрунтуються, насамперед, на вимогах, системне тестування – на вимогах та архітектури, комплексне тестування – на вимогах, архітектурі та інтерфейсах, а компонентне тестування – на вимогах, архітектурі, інтерфейсах та алгоритмах

Цілі

V-модель забезпечує підтримку у плануванні та реалізації проекту. В ході проекту ставляться такі завдання:

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

1. Мінімізація ризиків: V-подібна модель робить проект більш прозорим і підвищує якість контролю проекту шляхом стандартизації проміжних цілей і опису відповідних їм результатів та відповідальних осіб. Це дозволяє виявляти відхилення в проекті і ризики на ранніх стадіях і покращує якість управління проектом.

2. Підвищення та гарантії якості: V-Model – стандартизована модель розробки, що дозволяє домогтися від проекту результатів бажаної якості. Проміжні результати можуть бути перевірені на ранніх стадіях. Універсальне документування полегшує читаність, зрозумілість та контрольованість.

3. Зменшення загальної вартості проекту: Ресурси на розробку, виробництво, управління і підтримку можуть бути заздалегідь прораховані та проконтрольовані. Отримувані результати також універсальні і легко прогножуються. Це зменшує витрати на подальші стадії та проекти.

4. Підвищення якості комунікації між учасниками проекту: Універсальний опис усіх елементів та умов полегшує взаєморозуміння всіх учасників проекту. Таким чином, зменшуються неточності у розумінні між користувачем, покупцем, постачальником і розробником.

Переваги:

– Користувачі V-Model беруть участь у розробці та підтримці V-моделі. Комітет з контролю за змінами підтримує проект і збирається раз на рік для обробки всіх отриманих запитів на внесення змін до V-Model.

– На старті будь-якого проекту V-подібна модель може бути адаптована під цей проект, так як ця модель не залежить від типів організацій та проектів.

– V-model дозволяє розбити діяльність на окремі кроки, кожен з яких буде включати в себе необхідні для нього дії, інструкції до них, рекомендації та докладне пояснення діяльності.

Обмеження. Наступні моменти не враховуються в V-моделі, але можуть бути розглянуті окремо, або можливо адаптувати модель під них:

– Не регулюється розміщення контрактів на обслуговування.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

– Організація і виконання управління, обслуговування, ремонту та утилізації системи не враховуються в V-моделі. Однак, планування і підготовка до цих операцій моделлю розглядаються.

– V-подібна модель більше стосується розробки програмного забезпечення в проєкті, ніж всієї організації процесу.

Переваги:

– У моделі особливе значення надається плануванню, спрямованому на верифікацію та атестацію розроблювального продукту на ранніх стадіях його розробки. Фаза модульного тестування підтверджує правильність деталізованого проектування. Фази інтеграції та тестування реалізують архітектурне проектування або проектування на вищому рівні. Фаза тестування системи підтверджує правильність виконання етапу вимог до продукту і його специфікації.

– У моделі передбачені атестація та верифікація всіх зовнішніх і внутрішніх отриманих даних, а не тільки самого програмного продукту.

– У V-подібної моделі визначення вимог виконується перед розробкою проєкту системи, а проектування ПО – перед розробкою компонентів.

– Модель визначає продукти, які повинні бути отримані в результаті процесу розробки, причому кожен отриманий дани повинні піддаватися тестуванню.

– Завдяки моделі менеджери проєкту можуть відслідковувати хід процесу розробки, так як в даному випадку цілком можливо скористатися тимчасовою шкалою, а завершення кожної фази є контрольною точкою.

Недоліки:

– Модель не передбачає роботу з паралельними подіями.

– У моделі не передбачено внесення вимоги динамічних змін на різних етапах життєвого циклу.

– Тестування вимог в життєвому циклі відбувається занадто пізно, внаслідок чого неможливо внести змін, не вплинувши при цьому на графік виконання проєкту.

– У модель не входять дії, спрямовані на аналіз ризиків.

– Деякий результат можна отримати тільки при досягненні низу букви V.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використати фіналіста конкурсу AES – шифр Rijndael. Він є нетрадиційним блоковим шифром, оскільки не використовує мережу Фейштеля для криптоперетворень. Алгоритм представляє кожний блок кодуємих даних у вигляді двовимірного масиву байт розміром 4x4, 4x6 або 4x8 залежно від установленної довжини блоку. Далі на відповідних етапах перетворення відбуваються або над незалежними стовпцями, або над незалежними рядками, або взагалі над окремими байтами в таблиці.

Всі перетворення в шифрі мають строге математичне обґрунтування. Сама структура й послідовність операцій дозволяють виконувати даний алгоритм ефективно як на 16-бітних так і на 64-бітних процесорах.

У структурі алгоритму закладена можливість паралельного виконання деяких операцій, що на багатопроцесорних робочих станціях може ще підняти швидкість шифрування в 4 рази.

Алгоритм складається з деякої кількості раундів (від 10 до 14 – це залежить від розміру блоку й довжини ключа), у яких послідовно виконуються наступні операції :

ByteSub – Таблична підстановка 8x8 біт (рисунок 4.3).

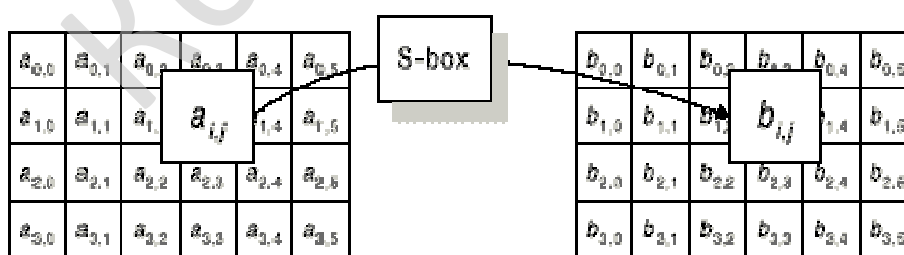


Рисунок 4.3 – Таблична підстановка 8x8 біт

ShiftRow – зрушення рядків у двовимірному масиві на різні зсуви (рисунок 4.4).

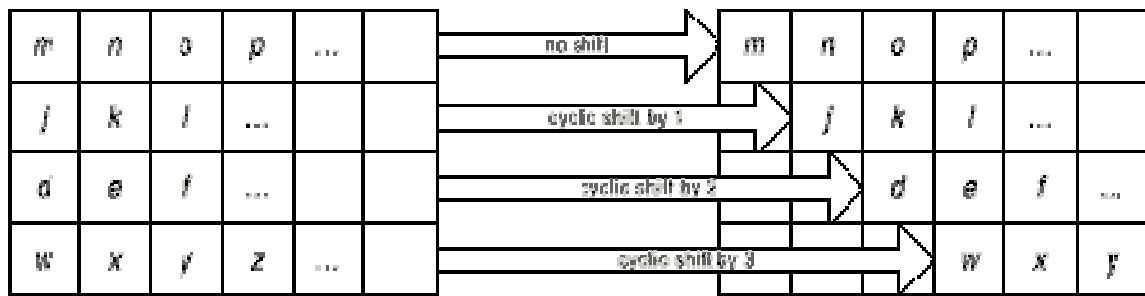


Рисунок 4.4 – Зрушення рядків у двовимірному масиві на різні зсуви

MixColumn – математичне перетворення, що перемішує дані усередині стовпця (рисунок 4.5).

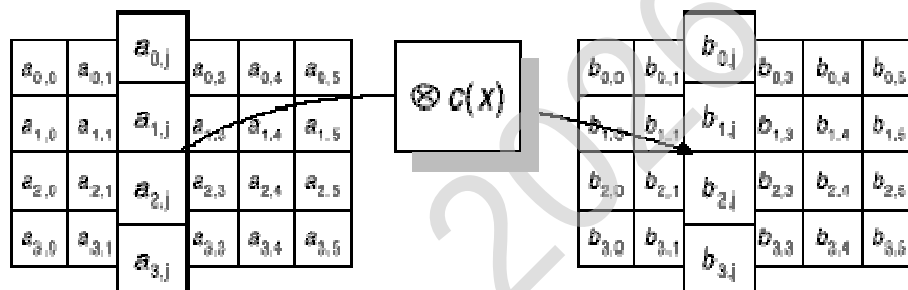


Рисунок 4.5 – Математичне перетворення, що перемішує дані усередині стовпця

AddRoundKey – додавання матеріалу ключа операцією XOR (рисунок 4.6).

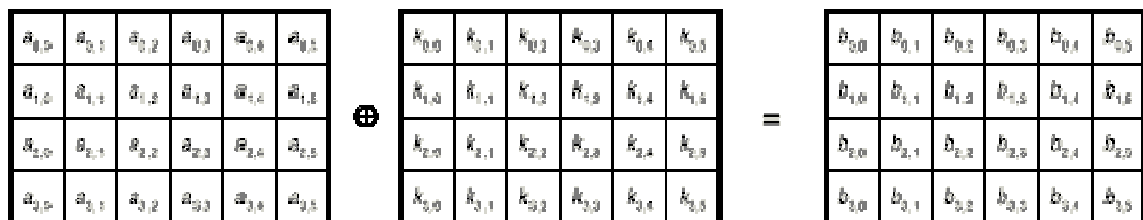


Рисунок 4.6 – Додавання матеріалу ключа операцією XOR

В останньому раунді операція перемішування стовпців відсутня, що робить всю послідовність операцій симетричною.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено розроблене у бакалаврської дипломної роботі система керування оптичною комунікаційною технікою ЦОД. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи: Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші; Функціональних кнопок ПЗ; Верхнього меню; Розділу обрання групи; Розділу виведення результату роботи системи.

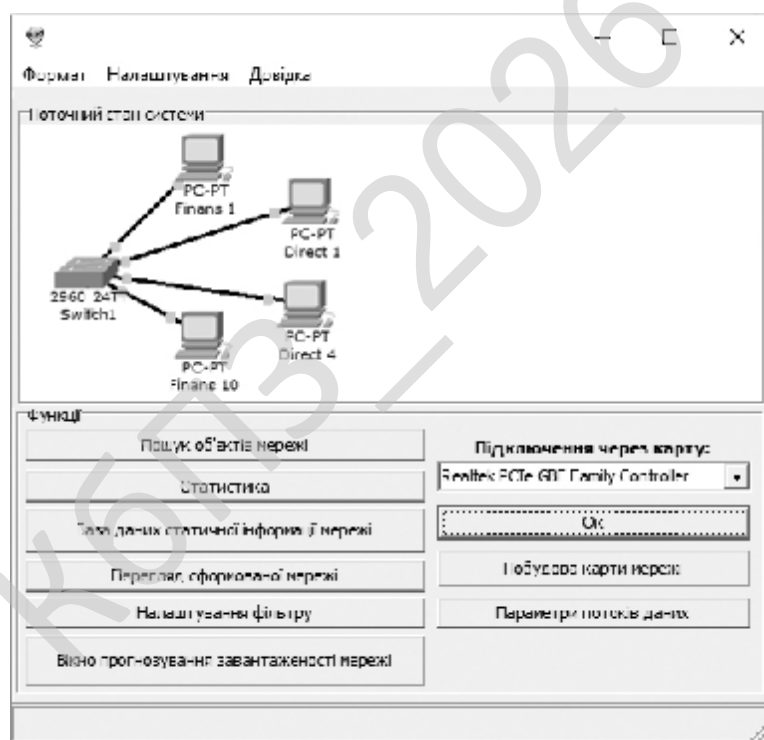


Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий. Якщо програма не видала ніяких

помилки, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

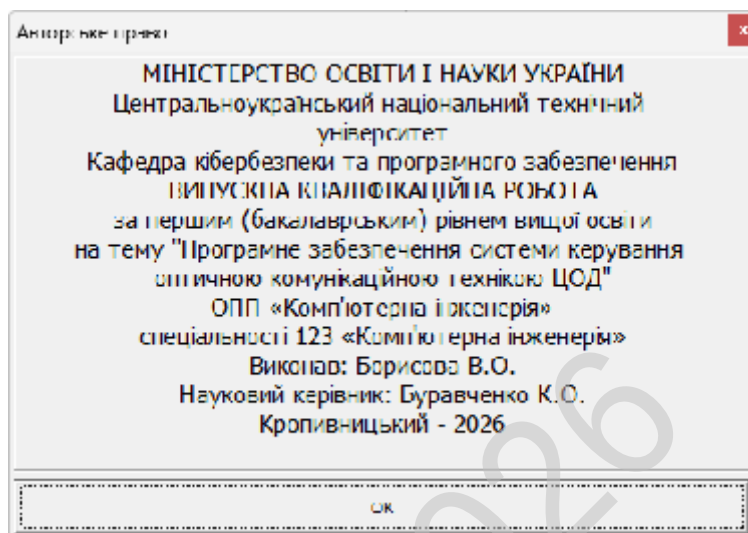


Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

- Випуск.
- Встановлення та активація.
- Деактивація.
- Адаптація.
- Оновлення.
- Вмонтування.
- Відстежування версій.
- Видалення.
- Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

- відповідності поставленим вимогам;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з ОС та стороннім ПЗ.

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

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

Проводилось тестування чорної скриньки.

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

- Як виконуються функції програми.
- Як приймаються вихідні дані.
- Як виробляються результати.
- Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^{10} . Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чію поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

- Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ІТс).

- Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його ОТ).

Будь-який спосіб тестування «чорної скриньки» повинен:

- Виявити такі вхідні дані, які з високою ймовірністю належать набору ІТс.
- Сформулювати такі очікувані результати, які з високою ймовірністю є елементами набору ОТ.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

- Некоректних чи відсутніх функцій.
- Помилки інтерфейсу.
- Помилки у зовнішніх структурах даних або в доступі до зовнішньої бази даних.
- Помилки характеристик (необхідна ємність пам'яті і т.д.).
- Помилки ініціалізації та завершення.

Обрано умови розповсюдження – Shareware. Під умовно-безплатним програмним забезпеченням можна розуміти спосіб або метод розповсюдження

комерційного ПЗ на ринку (тобто на шляху до кінцевого користувача), при якому випробувачеві пропонується обмежена за можливостями (не повнофункціональна або демонстраційна версія), терміном дії (тріал версія) або версія з вбудованим набридливим нагадуванням про необхідність оплати використання програми.

В угоді про використання (ліцензії для кінцевого користувача, EULA) також може бути обумовлена заборона на комерційне або професійне (не тестове) її використання.

Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно.

Звичайно користувач платить тільки за час завантаження файлів через Інтернет або за носій (CD диск, флешку, ключ). Протягом певного терміну, що становить зазвичай тридцять днів, він може користуватися програмою, тестувати її, освоювати її можливості.

Якщо після закінчення цього терміну користувач вирішить продовжити використання ПЗ, він зобов'язаний купити його (zareestruvatisya), заплативши авторові певну суму.

В іншому випадку користувач повинен припинити використання ПЗ та видалити його зі свого комп'ютера.

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи керування оптичною комунікаційною технікою ЦОД.

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

Рішення завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем керування оптичною комунікаційною технікою ЦОД.
- Досліджена система керування оптичною комунікаційною технікою ЦОД.
- На основі отриманих результатів досліджень створена програмна реалізація системи керування оптичною комунікаційною технікою ЦОД.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання керування оптичною комунікаційною технікою ЦОД.

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

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

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для системи керування оптичною комунікаційною технікою ЦОД. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		87

загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

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

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм AES.

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

К6ПЗ-2026

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		88

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Wendell Odom. «CCNA 200-301 Official Cert Guide, Volume 1». Cisco Press. 2020. – 848 p.
2. Wendell Odom. «CCNA 200-301 Official Cert Guide, Volume 2 Premium Edition eBook and Practice Test». Cisco Press. 2020. – 624 p.
3. Scott Jernigan «CompTIA Network+ Certification All-in-One Exam Guide, Eighth Edition». 2022. – 976 p.
4. Doug Lowe «Networking For Dummies 12th Edition». 2020. – 480 p.
5. Ramon Nastase «Computer Networking: The Beginner's guide for Mastering Computer Networking, the Internet and the OSI Model». 2018. – 186 p.
6. Russ White & Ethan Banks «Computer Networking Problems and Solutions: An Innovative Approach to Building Resilient, Modern Networks». 2017. – 832 p.
7. Вінтенко Б., Смірнов О., Миронець І., Смірнова Т., Смірнов С. «Імітаційна модель шляхів вхідних даних комп'ютерної інтелектуальної системи підтримки оператора енергоблоку АЕС». *Комбінаторні конфігурації та їхні застосування: Матеріали XXVII Міжнародного науково-практичного семінару, присвяченого 125-річчю Національного університету «Запорізька політехніка» (Запоріжжя-Кропивницький-Київ, 4-6 червня 2025 р.)*. Запоріжжя: НУ «Запорізька політехніка», 2025. С.82-91.
8. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Drieiev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». *International Review on Modelling and Simulations* 18 (1), 2025. pp. 32-42.
9. Смірнова Т.В., Коноплицька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка*. 2024. №4(24), С. 6-27.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		89

10. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології*, 2024, № 13, с. 28-35.
11. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.
12. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.
13. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchев, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.
14. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.
15. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувацького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.
16. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». *CEUR Workshop Proceedings* Volume 3156, 2022, Pages 390-399.
17. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління*, № 2(70). 2022. С. 28-37.

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		90

18. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 3(69). С. 93-98.

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

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

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

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

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

24. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable*

Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

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

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

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

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

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

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

31. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of*

Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

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

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

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

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

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

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

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

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		93

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

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

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

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

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

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

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

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

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		94

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

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

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

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

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

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

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

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

					ВКРБ-123.26.0032.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		95