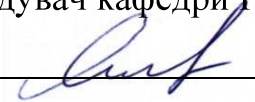


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ



к.т.н, доц. Михелєв І.Л.

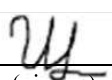
" ____ " _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "магістр"

на тему: «Дослідження технологій та впровадження мультиагентних систем для
управління матеріальними потоками»

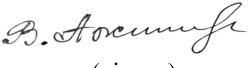
Виконав: студент групи 6145 м



Шилов Ф.А.
(підпис)

Керівник роботи:

доцент кафедри ІУСТ, к.т.н.,
(посада, науковий ступінь вчене звання)



Ажищев В.Ф.
(підпис)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

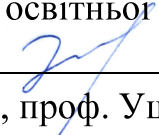
Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

Спеціальність 124 "Системний аналіз"

Освітня програма "Системний аналіз"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми


д.ф-м.н, проф. Ушкац М.В.

"___" "___" 2024 р.

*ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ*
на здобуття ступеня вищої освіти "магістр"

Студенту Шишлову Феліксу Артуровичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи: Дослідження технологій та впровадження мультиагентних систем для управління матеріальними потоками _____

Керівник роботи к.т.н., доцент Ажищев Віктор Федорович _____

Затверджені наказом ректора № 1055-уч. від "08" жовтня 2024 року.

2. Термін подання роботи: 15 грудня 2024 р. _____

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем, матеріали наукового стажування _____

4. Перелік питань, що належать до розробки (найменування розділів) _____

5. Перелік презентаційних матеріалів _____

6. Консультанти розділів роботи

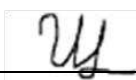
Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Ажищев В.Ф.	05.09.2023	22.12.2023
2	Ажищев В.Ф.	23.12.2023	15.06.2024
3	Ажищев В.Ф.	15.06.2024	25.09.2024
4	Ажищев В.Ф.	25.09.2024	14.11.2024
5	Ажищев В.Ф.	25.09.2024	14.11.2024
6	Ажищев В.Ф.	25.09.2024	14.11.2024

7. Дата видачі завдання 05 вересня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

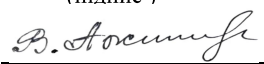
№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Дослідження та аналіз предметної галузі	22.12.2024	
2	Розробка концепції	15.03.2024	
3	Розділ з розробки проектних рішень по системі (та її частинах)	15.06.2024	
4	Реалізація проекту та розробка робочої документації	25.09.2024	
5	Обчислення економічної ефективності	14.11.2024	
6	Вирішення питань охорони праці та навколишнього середовища	14.11.2024	
7	Оформлення пояснювальної записки	14.11.2024	
8	Подання магістерської роботи на перевірку на плагіат	16.11.2024	
9	Подання магістерської роботи рецензенту	20.11.2024	
10	Подання магістерської роботи на державний захист	15.12.2024	

Студент


(підпис)

Шишлов Ф.А.
(ПІБ)

Керівник роботи


(підпис)

Ажищев В.Ф.
(ПІБ)

АНОТАЦІЯ

Темою роботи являється «Дослідження технологій та впровадження мультиагентних систем для управління матеріальними потоками».

Актуальність даної теми обумовлюється тим, що існуючі системи автоматизації управління матеріальним потоком не відповідають сучасним потребам в гнучкості, в адаптивності та в оперативній реакції на події в реальному часі. Об'єктом досліджень являються процеси управління матеріальними потоками на підприємстві. Предмет дослідження – підходи та механізми автоматизації процесів управління матеріальними потоками.

Метою роботи є підвищення ефективності управління матеріальним потоком підприємства шляхом автоматизації даного процесу за допомогою технології МАС. Для досягнення поставленої мети в роботі вирішено ряд задач по дослідженню теоретичних основ управління матеріальним потоком, аналізу класичних концепції організації даного процесу, дослідженню традиційних підходів до автоматизації задач управління ресурсами, та проведено дослідження технології МАС як засобу подолання визначених на попередніх етапах проблем. Кінцевою задачею являється розробка проекту МАС управління матеріальними потоками для вирішення конкретної прикладної задачі.

В якості прикладної предметної галузі прикладної задачі було обрано діяльність підприємства спеціалізованого на розкрої листових матеріалів.

Наукова новизна роботи полягає в автоматизації процесу управління матеріальним потоком підприємства за допомогою МА технологій. Практичним значенням результатів являється розробка проекту МАС управління ресурсами на етапі виробництва та збуту для підприємств листового розкрою.

На етапі дослідження використовуються методи системного аналізу, дослідження операцій, теорії прийняття рішень, лінійного програмування та інше. На етапі розробки застосовується методологія проектування UML.

Ключові слова: системний аналіз, мультиагентні системи, управління матеріальними потоками.

SUMMARY

The theme of the work is «Research into technologies and implementation of multi-agent systems for material flow management».

Actuality of this theme is caused by the fact that the existing system of automation of management of the material flow do not meet current needs for flexibility, adaptability and rapid response to events in real time. The object of research is the processes of materials management at the enterprise. Subject of research - the approaches and mechanisms to automate the management of material flows.

The aim of work is to improve the management of the material flow of the enterprise by automating this process with the help of the technology of the MASSES. To achieve this goal solved a number of tasks on research of theoretical bases of management of material flow, analysis of the classical concept of the organization of this process, the study of traditional approaches to automation problems of resource management, and conducted a survey of the technologies of the MASSES as a means to overcome identified at the previous stages of the problems. The ultimate goal is the development of the project MASSES of material management solutions for specific applied tasks.

As applied subject area of application was selected activity of the enterprise specialized in cutting of sheet materials.

Scientific novelty of work consists in the automation of management process of the material flow of the enterprise with the help of MA technologies. Practical value of the results is the development of the project of the MASSES resource management at the stage of production and sales of the enterprises of sheet cutting.

At the stage of research is used methods of system analysis, operations research, decision theory, linear programming, and others On the stage of the development of the methodology applies UML design.

Keywords: systems analysis, multi-agent systems, material flow management.

ЗМІСТ

СПИСОК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ ТА ТЕРМІНІВ

ВСТУП

1 ДОСЛІДЖЕННЯ МУЛЬТИАГЕНТНОЇ ТЕХНОЛОГІЇ В УПРАВЛІННІ МАТЕРІАЛЬНИМИ ПОТОКАМИ ПІДПРИЄМСТВА

- 1.1 Обумовлення актуальності вибраної теми та постановка проблем
- 1.2 Дослідження процесу управління матеріальними потоками
- 1.3 Аналіз існуючих систем управління матеріальними потоками
- 1.4 Аналіз проблем в управлінні матеріальними потоками за допомогою існуючих систем автоматизації
- 1.5 Дослідження можливості застосування МАС в управлінні матеріальними потоками
 - 1.5.1 Аналіз останніх досліджень та досягнень в області застосування мультиагентних технологій для вирішення задач логістики
 - 1.5.2 Дослідження технології МАС
 - 1.5.3 Характеристика мультиагентного підходу до вирішення задач управління матеріальними потоками
 - 1.5.4 Порівняльний аналіз мультиагентного підходу управління матеріальними потоками

2 ВИБІР ПРИКЛАДНОЇ ЗАДАЧІ ТА РОЗРОБКА ПРОЕКТУ МУЛЬТИАГЕНТНОЇ СИСТЕМИ ДЛЯ ЇЇ ВИРІШЕННЯ

- 2.1 Обумовлення актуальності вибраної теми та постановка проблем
- 2.2 Аналіз діяльності обраного підприємства
- 2.3 Виявлення проблем
- 2.4 Постановка задачі
- 2.5 Розробка концепції МАС управління матеріальними потоками підприємства на етапі виробництва та збуту
- 2.6 Загальносистемні рішення
 - 2.6.1 Розробка динаміки системи
 - 2.6.2 Розробка активності та логіки поведінки агентів системи

2.6.3 Розробка рішень з взаємодії агентів системи

2.7 Рішення з математичного забезпечення системи

2.8 Рішення з інформаційного забезпечення системи

2.9 Рішення з програмного забезпечення

2.9.1 Вибір та опис мультиагентної платформи

2.9.2 Діаграма компонентів системи

2.9.3 Розгортання системи

3 РОЗРОБКА РОБОЧОЇ ДОКУМЕНТАЦІЇ ПО ПРОЕКТУ

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ
РОЗРОБЛЕНОЇ СИСТЕМИ

5 ОХОРОНА ПРАЦІ КОРИСТУВАЧІВ СИСТЕМИ

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

ВИСНОВКИ

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

ДОДАТОК А – Технічне завдання

ДОДАТОК Б – Технічний опис системи

ДОДАТОК В – Інструкція користувача

ДОДАТОК Г – Програма та методика випробувань

ДОДАТОК Д – Програмний код

СПИСОК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ ТА ТЕРМІНІВ

- АРМ – автоматизоване робоче місце;
- АСУ – автоматизована система управління;
- БД – база даних;
- БЗ – база знань;
- ВМТП – відділ матеріально-технічного постачання;
- ІА – інтелектуальний агент;
- ІС – інформаційна система;
- МАС – мультиагентна система;
- МП – матеріальний потік;
- ПАР – програма автоматизованого розкрою;
- САПР – система автоматизованого проектування;
- ТМЦ – товарно-матеріальні цінності;
- BDI – beliefs, desires, intentions (переконання, бажання, наміри);
- BOM – bill of material (специфікація матеріалів);
- CRP – capacity requirements planning (системою планування потреби в потужностях);
- JIT – just in time (концепція «точно в строк»);
- MPS – master production schedule (головний календарний план виробництва);
- MRP – material requirements planning (система планування потреб в матеріалах);
- MRP II – manufacturing resource planning (планування виробничих ресурсів);
- OPT – optimized production technology (оптимізована виробнича технологія).

ВСТУП

При побудові сучасних інформаційно-управляючих систем масштабу підприємства окреме місце відводиться автоматизації процесу управління матеріальними потоками. Ефективність виробництва напряду залежить від оперативного управління матеріальними потоками, так як даний процес включає в собі вирішення всіх питань, зв'язаних з використанням виробничих ресурсів в часі та просторі.

Актуальність даної теми обумовлюється тим, що існуючі системи автоматизації управління матеріальним потоком, в повній мірі, не відповідають сучасним потребам в гнучкості, адаптивності та в оперативній реакції на події в реальному часі.

Сучасні умови ведення бізнесу приводять до змін в організації процесів управління ресурсами підприємства на всьому логістичному ланцюгу від постачальника до споживача. Відмічається зростання складності та динаміки прийняття управлінських рішень. Дані фактори викликають потребу в більш гнучких та адаптивних системах управління, які можуть швидко реагувати на зміни в потребах споживачів, зміни на ринках сировини та на непередбачені події на всіх етапах логістичного ланцюгу. Враховуючі такі сучасні вимоги традиційні системи логістики стають менш ефективними, так як не пристосовані до постійно змінюваних за об'ємом та змістом внутрішніх та зовнішніх потоків замовлень. Також окрім «жорсткості» та відсутності реакції у реальному часі на непередбачені події, в традиційних системах необхідно відмітити:

- надто великі відхилення кошторисного планування від реального положення справ, незважаючи на значні витрати на електронну обробку даних і систему в цілому;
- відсутність можливостей ефективно впливати на продуктивність, терміни проходження циклу і необхідний рівень запасів;
- недостатню свободу дій плануючих структур і пов'язаних з плануванням співробітників.

Отже, об'єктом проведених в роботі досліджень являються процеси управління матеріальними потоками на підприємстві. Предмет дослідження – підходи та механізми автоматизації даних процесів.

Шляхом подолання визначених вище проблем та задоволення сучасних вимог до систем автоматизації управління ресурсами підприємства є застосування в даній галузі мультиагентного підходу. Застосування мультиагентних технологій в області управління матеріальними потоками приводить до кардинальних змін в самій організації логістики виробництва. Централізація, жорстке управління та глобальна оптимізація класичних систем замінюється децентралізованою організацією системи, переговорами між агентами та пошуком локальних оптимумів. Такий підхід забезпечує високу ефективність, гнучкість, адаптивність та надійність систем підприємства.

Отже, метою роботи є підвищення ефективності управління матеріальним потоком підприємства шляхом автоматизації даного процесу за допомогою технології МАС.

В основі мультиагентних систем лежить поняття «агента». Характерними особливостями агентів є:

- колегіальність, тобто здатність до колективної цілеспрямованої поведінки в інтересах вирішення загальної задачі;
- автономність, тобто здатність самостійно вирішувати локальні завдання;
- активність, тобто здатність до активних дій заради досягнення загальних і локальних цілей;
- інформаційна та рухова мобільність, тобто спроможність активно переміщатися і цілеспрямовано шукати і знаходити інформацію, енергію та об'єкти, необхідні для кооперативного рішення загальної задачі;
- адаптивність, тобто здатність автоматично пристосовуватись до невизначених умов в динамічному середовищі.

Мультиагентні системи інтегрують в собі досягнення в сферах інформаційних технологій, штучного інтелекту, розподілених інформаційних систем, комп'ютерних мереж та комп'ютерної техніки.

В мультиагентному підході вирішення задач логістики усім об'єктам, які задіяні в процесі управління матеріальним потоком (вантажі, склади, транспортувальні засоби, тощо), ставиться у відповідність свій інтелектуальний агент, який діє в інтересах свого об'єкту, використовуючи різні стратегії. Інтелектуальні агенти вступають в взаємодію як і з навколишнім середовищем так і один з одним – обмінюються інформацією та вступають в переговори, находячи компроміс. Такий підхід сприяє всебічному вирішенню задач управління матеріальним потоком.

В якості базових агентів використовуються два типа: агент замовлення та агент ресурсу. Кожен агент замовлення повинен знайти ресурс, який його виконає максимально підходящим для нього способом, і навпаки, кожен агент ресурсу прагне найбільш вигідне для себе замовлення. Крім множини агентів замовлень і готових об'єктів, які постійно шукають зустрічі, мультиагентна система може передбачати і ряд інших більш традиційних додаткових агентів, наприклад: агенти виробника, складу, конвеєра та інші.

Використання мультиагентного підходу в управлінні матеріальними потоками підприємства має ряд своїх переваг:

- збалансований підхід до процесу планування та побудови розкладів;
- врахування індивідуальних вимог на рівні кожного окремо взятого замовлення та ресурсу;
- використання в процесі планування стратегій, інтуїтивно використовуваних людиною на базі досвіду;
- прозорість діяльності підрозділів підприємства;
- можливість ефективно і швидкої реакції на непередбачені події.

Для досягнення поставленої вище мети в роботі вирішені наступні задачі:

- Проаналізовано сучасні вимоги до систем автоматизації управління матеріальними потоками підприємства;
- Проведено аналіз існуючих проблем в управлінні матеріальними потоками за допомогою систем автоматизації;
- Визначено можливості та переваги застосування МАС в управлінні матеріальним потоком;

— Розроблено концепцію та проект МАС для вирішення обраної прикладної задачі.

В якості прикладної предметної галузі було обрано діяльність підприємства спеціалізованого на розкрої листових матеріалів. Обрана галузь характеризується умовами невизначеності та високою динамікою в прийнятті управлінських рішень.

1 ДОСЛІДЖЕННЯ МУЛЬТИАГЕНТНОЇ ТЕХНОЛОГІЇ В УПРАВЛІННІ МАТЕРІАЛЬНИМИ ПОТОКАМИ ПІДПРИЄМСТВА

1.1 Обумовлення актуальності вибраної теми та постановка проблем

Глобалізація світової економіки змушує підприємства зіткнутися з рядом нових проблем зв'язаних з ростом складності прийняття управлінських рішень[17]:

- Умови невизначеності: важко передбачити зміну попиту та пропозиції;
- Подієвість: інтенсивність виникнення непередбачених подій, які призводять до зміни планів;
- Ситуативність: рішення приймаються не заздалегідь а по ситуації;
- Багатофакторність: велика кількість критеріїв, переваг та обмежень в прийнятті рішень;
- Висока зв'язність (кореляція подій): прийняття одного рішення призводить до хвилі змін в інших планах;
- Індивідуальність: клієнти вимагають все більш індивідуального підходу, що знижує можливість узагальнення вирішення задач;
- Конфліктність: багато об'єктів з суперечливими інтересами;
- Трудомісткість: великий об'єм роботи по розрахунку наслідків прийняття рішення.

Такі умови організації управління матеріальними потоками підприємства значно посилюють саму динаміку прийняття рішень [17]:

- Стає необхідна дуже висока оперативність передачі/обробки інформації для прийняття рішення;
- Зменшується час на відповідь — рішення приймаються під зростаючим пресом часу;
- Стає необхідна постійна взаємодія з клієнтом, постачальником и т.д.

Зростаюча складність і динаміка прийняття управлінських рішень, вимагають нових методів та засобів в організації управління матеріальними

потоками. Нові рішення поставленої задачі повинні реалізовувати повний цикл оперативного управління [16]:

- Реакція на подію (в реальному часі);
- Динамічне планування/перепланування;
- Узгодження та перегляд планів «на ходу»;
- Моніторинг та контроль виконання планів.

Функціонування таких систем можна зобразити наступним чином (рисунок 1.1):

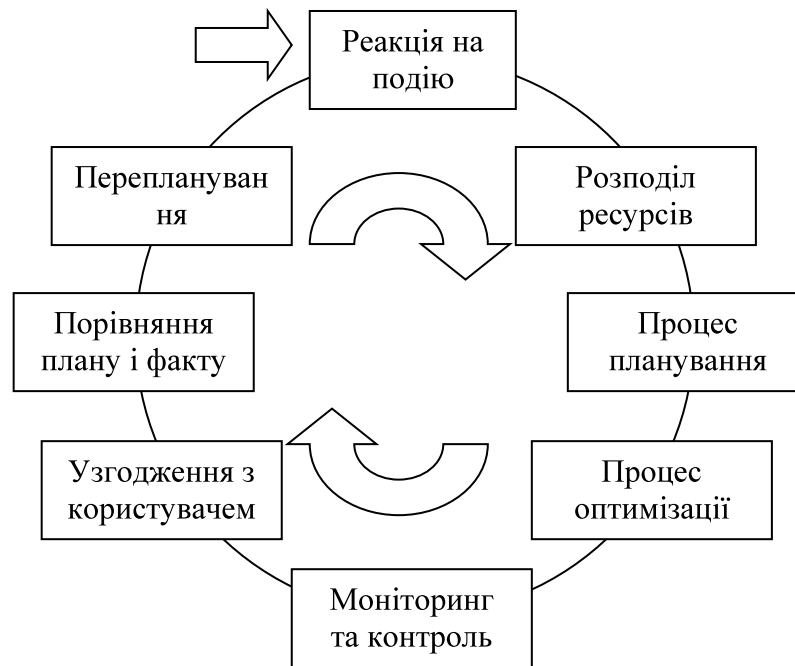


Рисунок 1.1 – Повний цикл управління ресурсами підприємства

«Реакція на подію» повинна відбуватися як і в ході виконання вже існуючого плану, де подіями можуть бути непередбачені: надходження нових замовлень, запізнення постачальників чи відмови верстатів, так і при плануванні з нуля, так як любий план можна розглядати як сукупність впорядкованих в часі та просторі подій, де подіями є: вже існуючі замовлення, очікуванні зупинки верстатів на профілактику та інше. Подія любого типу вимагає використання певних ресурсів, які необхідно виділити з їх загальної множини, яка вже має взаємозв'язок з сукупністю інших подій («розподіл ресурсів»). Наступним кроком є «процес планування» - зв'язування в часі та просторі подій та ресурсів між собою. Отриманий план підлягає «процесу оптимізації» - підвищення ефективності за заданими критеріями в інтересах поставлених цілей. Оптимізований план поступає на виконання та настає стадія «моніторингу та

контролю» за його виконанням. Вхіді виконання плану управління може стати необхідним, наприклад, змінити пріоритет події чи взагалі по деяким суб'єктивним/об'єктивним причинам внести зміни до оптимізованого плану («узгодження з користувачем»). Бувають ситуації коли управління і не необхідно вносити зміни до плану, але в ході плану, наприклад, працівники можуть відставати в часовому інтервалі від прийнятого графіку, «порівняння плану і факту» і визначає необхідність системи перейти в стадію «перепланування» за новими обставинами.

Отже, об'єктом проведених в роботі досліджень повинні являтися процеси управління матеріальними потоками на підприємстві. Предметом дослідження – підходи та механізми автоматизації даних процесів.

Шляхом задоволення сучасних вимог до систем автоматизації управління ресурсами підприємства є застосування в даній галузі мультиагентного підходу. Мультиагентні технології дозволяють будувати децентралізовані розподілені гнучкі системи які орієнтовані на досягнення локальних оптимумів і задоволення всіх об'єктів системи з урахуванням їх індивідуальних інтересів.

Отже, метою роботи є підвищення ефективності процесу управління матеріальним потоком підприємства шляхом автоматизації даного процесу за допомогою мультиагентних технологій.

Для досягнення мети необхідно вирішити ряд наступних задач:

- Провести дослідження процесу управління матеріальним потоком;
- Провести аналіз традиційних підходів вирішення задач управління матеріальними потоками;
- Провести дослідження технології мультиагентних систем;
- Визначити можливості та переваги застосування мультиагентних технологій в управлінні матеріальними потоками;
- Вибрати галузь прикладної задачі та розробити проект мультиагентної системи для її вирішення.

1.2 Дослідження процесу управління матеріальними потоками

Матеріальний потік – це сукупність товарно-матеріальних цінностей, яка розглядаються в процесі додатку до неї ряду логістичних операцій, які пов'язані

в часовому інтервалі [4]. На рисунку 1.2 зображена загальна схема протікання матеріальних потоків в межах логістичної системи управління.

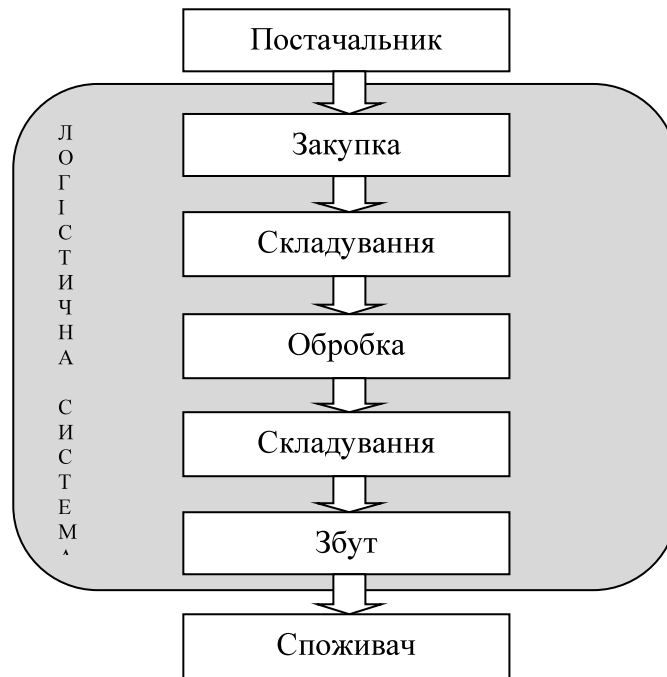


Рисунок 1.2 – Загальна схема протікання матеріальних потоків

Якщо етап складування являється суто обліковим і немає принципових проблем в своїй автоматизації, то автоматизація останніх етапів передбачає вирішення таких задач як: розподіл, планування, оптимізація та контроль за використанням ресурсів підприємства. Необхідно відмітити, що це складні багатокритеріальні задачі, які вимагають врахування індивідуальних особливостей замовлень, технології виробництва, можливостей робітників та станків, і ряду інших особливостей.

Управління матеріальним потоком являє собою процес ціле-направленого впливу на виробничі підрозділи, які зайняті просуванням матеріальних та інформаційних потоків з пункту виробництва в пункт споживання продукції [6]. Прийняття управлінських рішень здійснюється на основі даних о ході виконання виробничих заказів, інформації о ресурсах та запитів споживачів. На рисунку 1.3 зображено узагальнену схему даних для управління матеріальним потоком.

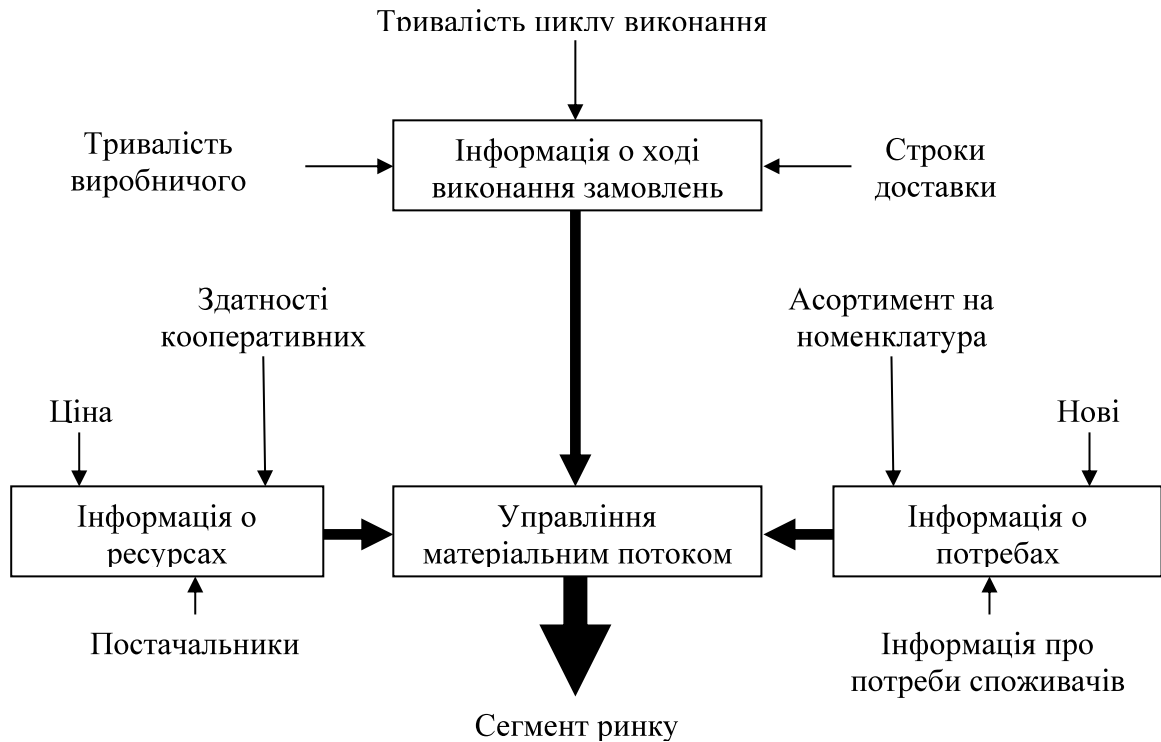


Рисунок 1.3 – Схема даних для управління матеріальним потоком

За допомогою інформації з ринку збуту та ринку постачання формується комплексний план виробництва, матеріального забезпечення та збуту готової продукції, згідно з яким організується діяльність по виконанню замовлень. В системі управління матеріальними потоками здійснюється циркуляція інформації та формується замкнутий контур управління з зворотнім зв'язком. Управління матеріальним потоком повинно забезпечувати постійне узгодження та взаємне корегування планів та дій ланок постачання, виробництва та збуту в середині підприємства [9]. В даному процесі вирішується низка задач з прогнозування попиту, виробництва, об'єму перевозок, визначення оптимальних об'ємів та напрямків матеріальних потоків.

На рисунку 1.4 зображена загальна схема управління матеріальним потоком підприємства.

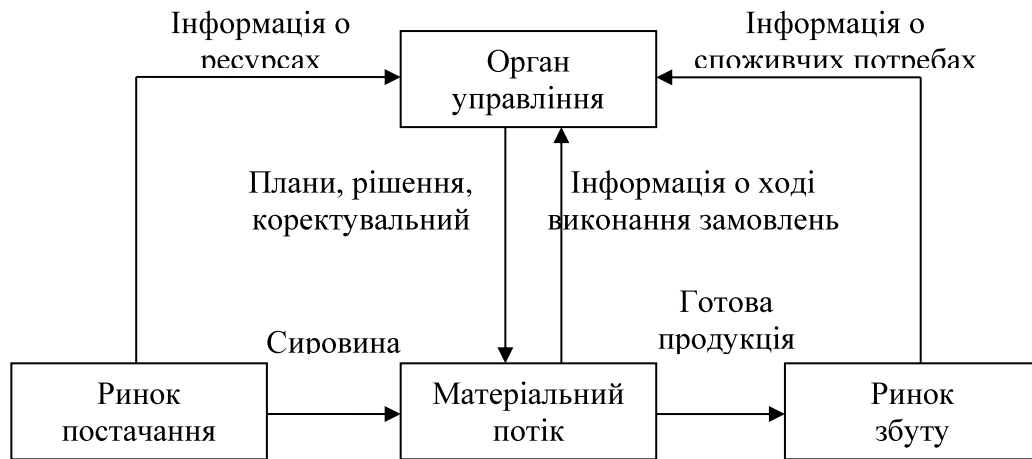


Рисунок 1.4 – Загальна схема управління матеріальним потоком

Управління матеріальними потоками у рамках внутрішньовиробничих логістичних систем може здійснюватися двома принципово різними способами: шляхом «виштовхування» чи «витягування» замовлення.

Система виштовхування ґрунтується на прогнозуванні розміру запасів сировини, матеріалів, деталей для кожної ланки логістичної системи. Виходячи з цього прогнозу, здійснюється керування всім багатоетапним процесом виробництва шляхом забезпечення виправданого об'єму матеріального запасу на кожній стадії обробки. При цій системі управління матеріальними потоками предмети труда переміщуються з однієї ділянки та іншу незалежно від її готовності до обробки та потреби в цих деталях [9].

На рисунку 1.5 зображено принципову схему системи виштовхування замовлення.

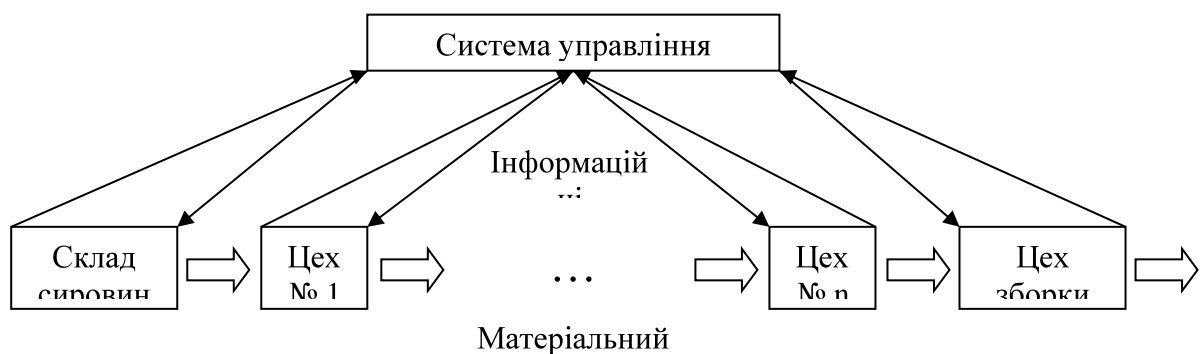


Рисунок 1.5 – Схема системи виштовхування

Даний спосіб управління дозволяє пов'язати складний виробничий механізм в єдину систему та максимально задіяти робочих та обладнання. Система виштовхування являє собою систему управління матеріальним потоком на всьому протязі логістичного ланцюга, в якому рішення о виконанні приймаються централізовано. При реалізації готової продукції система виштовхування проявляється як стратегія збуту, направлена на попереджувальне, відносно попиту, формування товарних запасів. Система виштовхування дозволяє вирішувати різноманітні задачі прогнозування. Для цього широко використовуються методи дослідження операцій та імітаційного моделювання.

Чим крупніше стає система виштовхування, тим чіткіше проявляються такі її проблемні місця [6, 9]:

- При різких змінах попиту або затримках в процесу виготовлення практично не можливо перепланувати виробництво для кожної стадії. Через це з'являються надлишкові запаси та затоварювання.
- Управлінському персоналу тяжко детально розібратися у всіх ситуаціях, пов'язаних з нормами виробітки та параметрами матеріальних потоків. Через це виробничий план повинен передбачати створення надлишкових страхових запасів.
- Любі оперативні, негайні зміни розміру партій виробів, а також тривалість виробничо-логістичних операцій викликають великі ускладнення, так як розрахувати в деталях оптимальні виробничі плани дуже важко.

На відміну від системи виштовхування, система витягування полягає в збереженні мінімального рівня запасів на кожному етапі виробництва та рух замовлень від наступної ділянки до попередньої, де наступна ділянка замовляє матеріали в відповідності з нормами та часом споживання своїх виробів. Основні цілі системи витягування замовлення [9]:

- Попередження розповсюдження зростання коливань попиту або об'єму продукції від наступного процесу до попереднього.
- Зведення до мінімуму коливання параметрів запасів між технологічними операціями.

- Максимальне спрощення управління запасами в процесі виробництва шляхом його децентралізації, підвищення рівня оперативного цехового управління.

Система витягування не потребує всебічної комп'ютеризації виробництва. Вона вимагає високу дисципліну, дотримання усіх параметрів поставок, а також підвищену відповідальність персоналу усіх рівнів, особливо виконавців [4].

На рисунку 1.6 зображена принципова схема системи витягування.

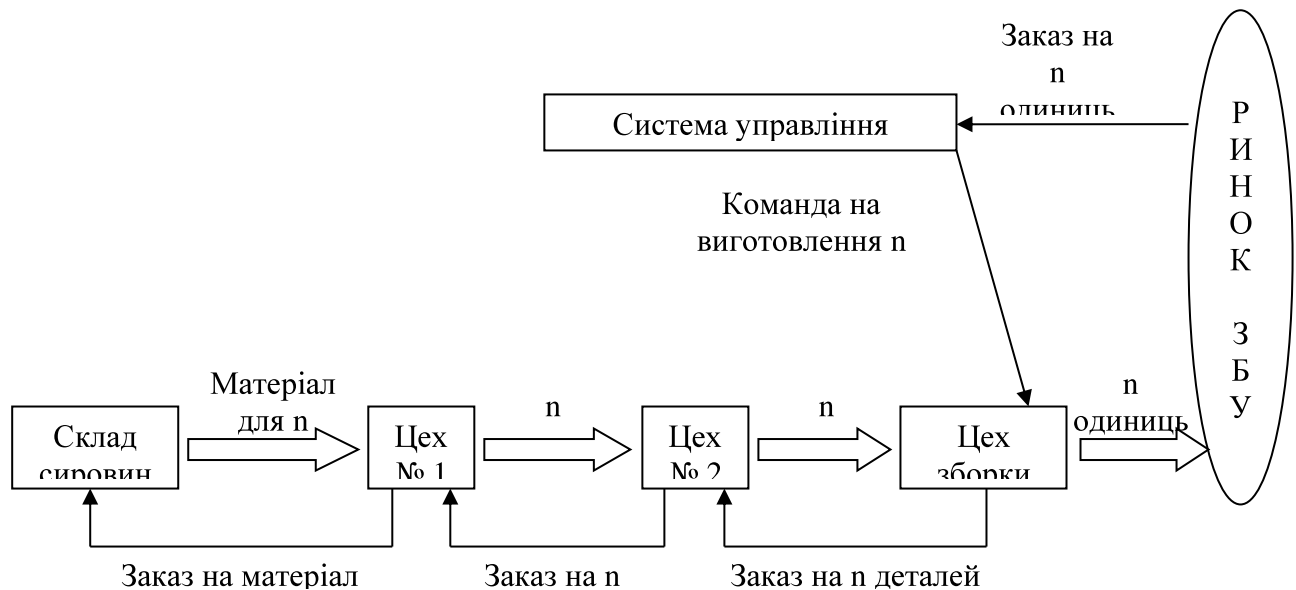


Рисунок 1.6 – Схема системи витягування

Не зважаючи на всі свої переваги, система витягування має ряд недоліків[6]:

- Залежність від партнерів з поставок;
- Зайві транспортні розходи;
- Простій робочих;
- Не визначеність майбутніх планів;
- Чуттєвість до непередбачених подій.

1.3 Аналіз існуючих систем управління матеріальними потоками

Під системою управління матеріальним потоком розуміють організаційний механізм формування та регулювання матеріальних потоків у рамках внутрішньовиробничої логістичної системи [19].

Початковим етапом в розвитку систем управління ресурсами підприємства була розробка системи MRP (Material Requirements Planning), системи планування потреб в матеріалах. В своїй схемі функціонування система використовує наступні блоки [9]:

- Інформація про попит. Потреба в výroбах. Які номенклатурні позиції потрібні споживачам. При цьому для продажу можуть бути як реальними замовленнями, так і прогнозами, - головне в цій методиці спланувати певний рівень виробництва у відповідності з планом продажів. Цей план потім ляже в основу формування головного календарного плану виробництва (Master Production Schedule, MPS);
- Дані про запаси складальних одиниць та наявності матеріалів для виробництва. Запаси готової продукції. Так як рідко яка продукція складається тільки з однієї деталі або виготовляється в одну стадію, то в цьому модулі враховуються стан і напівфабрикатів, і зібраних вузлів і т.п.
- Дані про склад виробів і норми витрат сировини, матеріалів і компонентів на одиницю виміру готової продукції - BOM (Bill of Material, «специфікація»).

MRP-система видає два масиви повідомлень: планові замовлення (planned orders) і рекомендації (action messages). Перший масив описує замовлення - його номер, дату запуску, дату виконання. А другий визначає управлінські інструкції: «виконати замовлення», «перепланувати замовлення» або «відмінити замовлення».

Ключовим недоліком таких систем було те, що вони не враховували можливостей виробництва, вважаючи виробничі потужності необмеженими [9]. Тому наступним кроком стала розробка систем, що дозволяють аналізувати завантаження виробничих потужностей і враховувати ресурсні обмеження

виробництва. Система MRP доповнювалась системою CRP (Capacity Requirements Planning) – системою планування потреби в потужностях. CRP-системи містять три основні блоки [9]:

- Календарний план виробництва (MPS), розроблений MRP-системою і містить відомості про виробничі замовлення.
- Дані про виробничі центри. Виробничий центр - це група типового взаємозамінного устаткування на виробничій ділянці. Такий ділянка розглядається як одна виробнича одиниця.
- Дані про технологічні маршрути - порядок виконання технологічних операцій, їх характеристики (час виконання, персонал тощо).

На рисунку 1.7 зображено блок-схему MRP/CRP-систем.

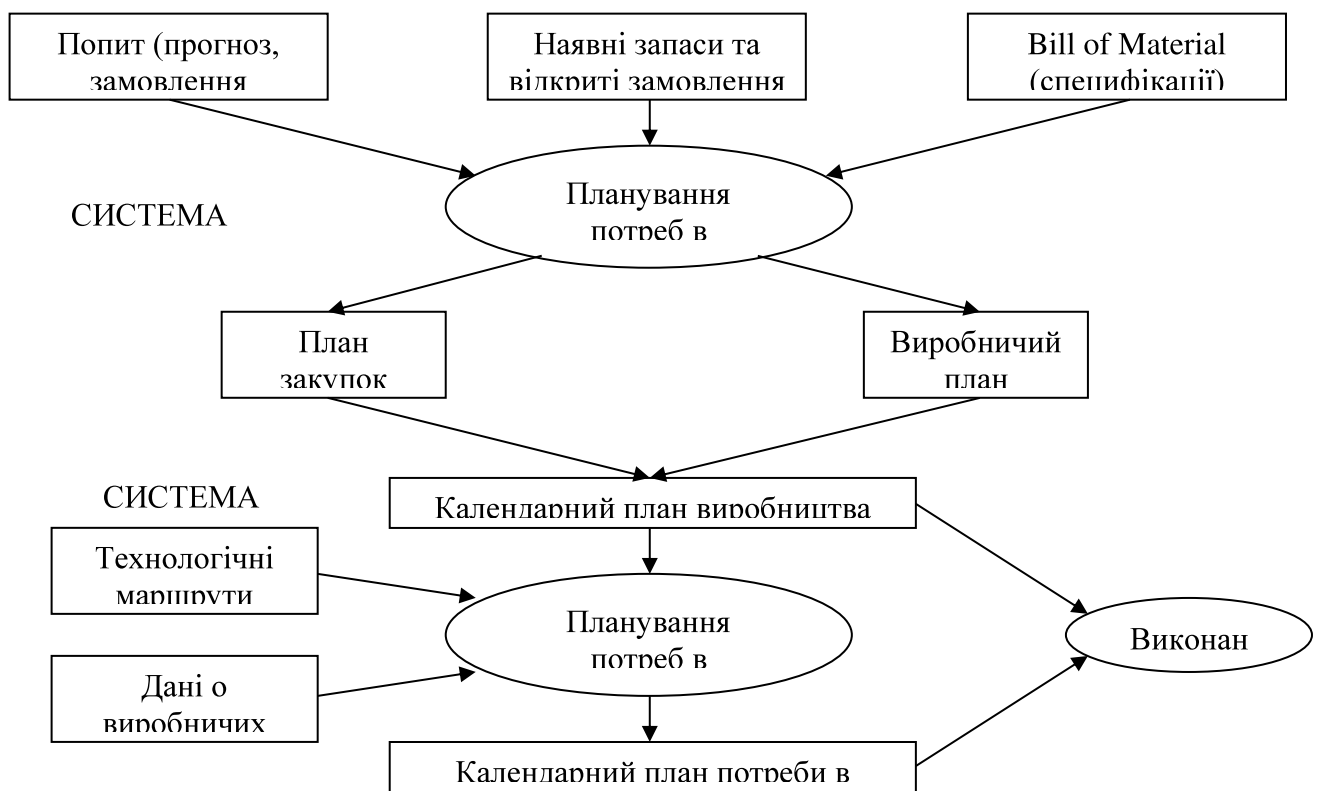


Рисунок 1.7 – Блок-схема MRP/CRP-систем

Системи MRP II (manufacturing resource planning - планування виробничих ресурсів) є подальшим розвитком систем MRP і орієнтовані на ефективне планування всіх ресурсів виробничого підприємства. Система MRP II дозволяє виконувати наступні бізнес-функції [9]:

- Планування розвитку бізнесу - оцінка прибутку і необхідні фінансові ресурси. Фактично це побудова бізнес-плану.

- Планування продажів - обсяг і динаміка продажів.
- Планування виробництва - виробничий план.
- Планування потреб у матеріалах і послугах - план закупівель.
- Планування виробничих потужностей - детальний план завантаження кожної виробничої одиниці у відповідності з виробничим планом.

На рисунку 1.8 зображено блок-схему системи MRP II.

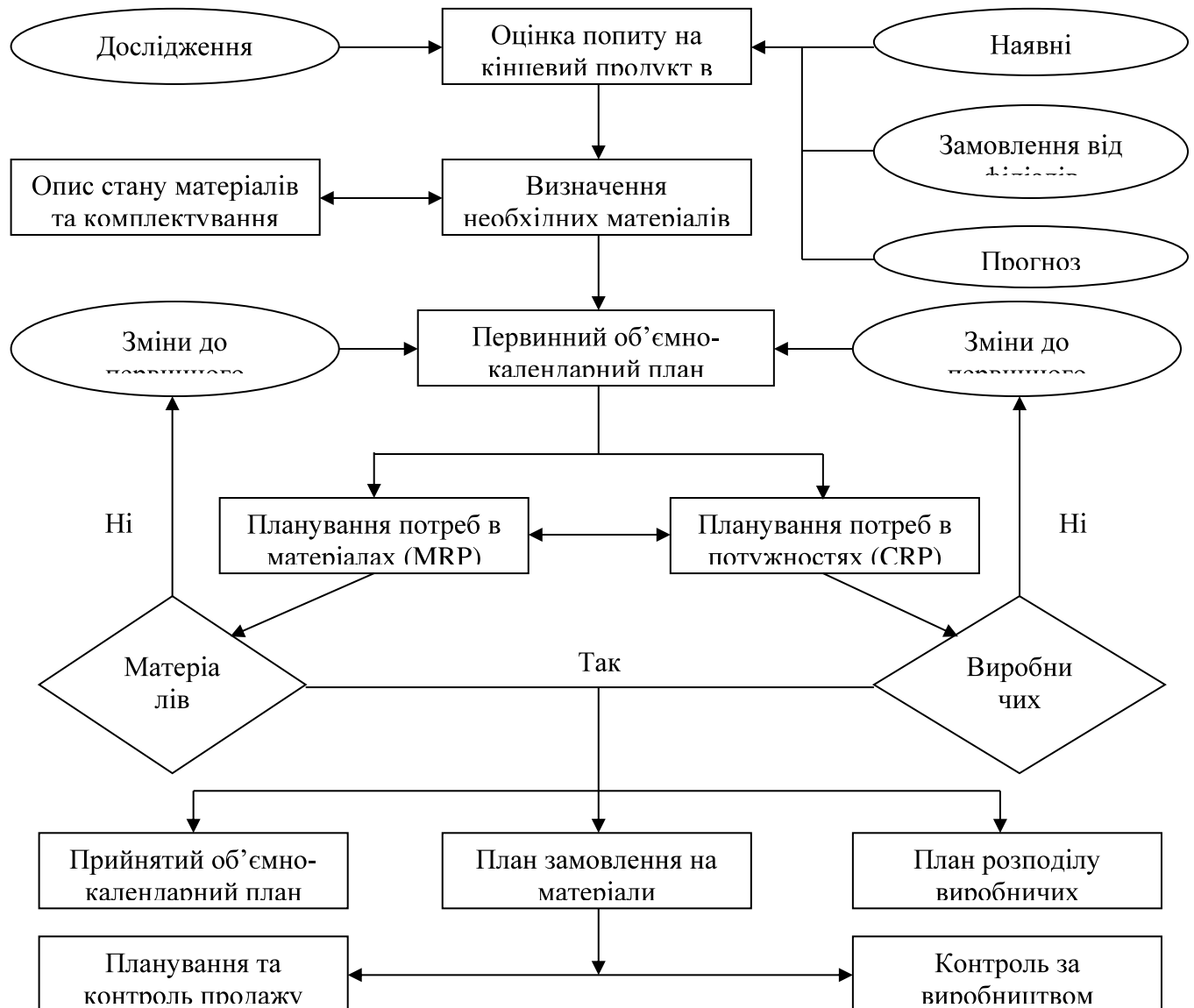


Рисунок 1.8 – Блок-схема MRP II-систем

Основним недоліком «штовхаючих» MRP систем є необхідність створення і підтримання значних буферних запасів між виробничими підрозділами та етапами технологічного циклу.

Протилежним до концепції MRP є підхід JIT (just in time) – «точно в строк». Основна ідея концепції полягає в наступному: якщо виробничий розклад задано, то можна так організувати рух матеріальних потоків, що всі матеріали,

компоненти і напівфабрикати будуть надходити в необхідній кількості, у потрібний місце і точно до призначеного терміну для виробництва, складання або реалізації готової продукції [19]. При цьому страхові запаси, які заморожують кошти фірми, не потрібні.

Реалізацією підходу JIT є «тягнуча» система KANBAN. Основними принципами функціонування системи є своєчасна поставка продукції замовнику, посилений контроль за якістю продукції на всіх етапах виробництва, ретельне налагодження устаткування, виключення випуску браку, скорочення числа постачальників комплектуючих виробів, максимальне наближення підприємств суміжників до місця розташування головного, як правило, складального заводу. Сутність системи KANBAN – на всі виробничі ділянки заводу, включаючи лінії кінцевого складання, строго за графіком поставляється саме та кількість сировини, матеріалів, комплектуючих деталей і вузлів, що дійсно необхідно для ритмічного випуску певного обсягу продукції [9]. Засобом передачі інформації в системі є спеціальна картка «kanban». Поширено два види карток: відбору і виробничого замовлення. У картці відбору зазначається кількість деталей (компонентів, напівфабрикатів), яку потрібно взяти на попередній ділянці обробки (складання), у той час як у картці виробничого замовлення – кількість деталей, яку потрібно виготовити (скласти) на попередній виробничій ділянці. Ці картки циркулюють як усередині підприємства, так і між підприємством та компаніями-співробітниками, а також на підприємствах-філіях.

На рисунку 1.9 зображено схему роботи системи KANBAN.

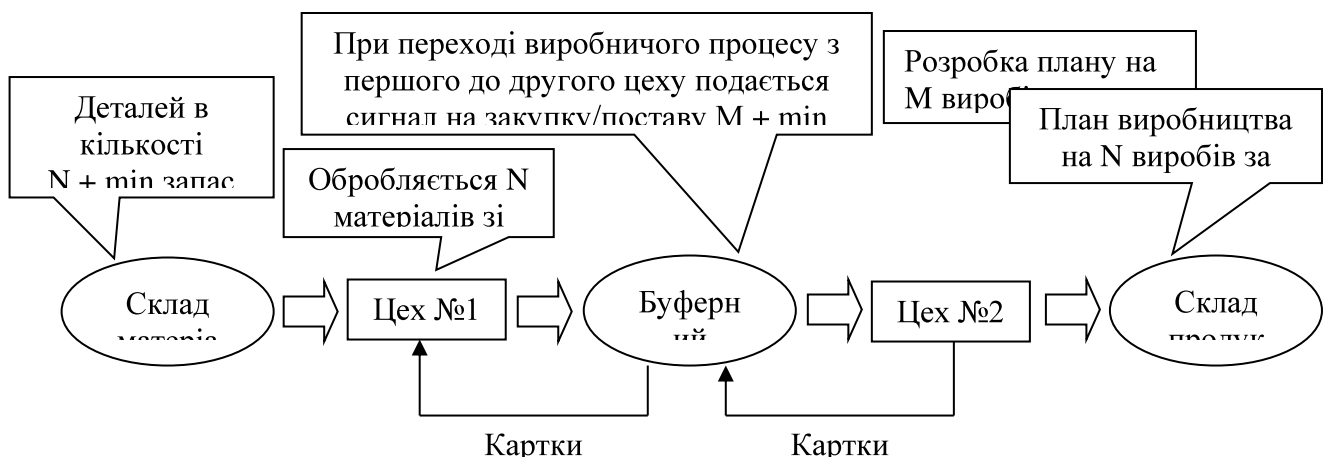


Рисунок 1.9 – Схема роботи системи KANBAN

Система «KANBAN» не вимагає тотальної комп'ютеризації виробництва, однак вона вимагає високу дисципліну поставок, а також високу відповідальність персоналу, так як центральне регулювання внутрішньовиробничого логістичного процесу обмежене. «Тягнучі» системи типу «KANBAN», усуваючи зайві запаси, можуть ефективно працювати лише при відносно коротких виробничих циклах, точному прогнозуванні попиту та ряду інших сприятливих виробничо-технологічних умовах.

Для виправлення недоліків, властивих обома типам систем, були зроблені спроби їх об'єднання в єдиному планово-виробничому і диспетчерському комп'ютерному комплексі. Одним з найбільш вдалих прикладів синтезу у виробництві продукції ключових елементів MRP і KANBAN на основі сучасних інформаційно-комп'ютерних технологій стала система «Optimized Production Tehnology» - OPT (оптимізована виробнича технологія) [9].

Система OPT належить до класу «тягнучих» систем, інтегруючих процеси постачання і виробництва. Основним принципом роботи цієї системи є виявлення у виробничому процесі так званих «вузьких» місць. Комп'ютерний розрахунок виробничих розкладів виконується на зміну, день, тиждень і т.д. Вирішуються також завдання контролю відвантаження запасів готової продукції споживачам, пошуку альтернативних ресурсів, видачі рекомендацій по повноцінних замінах у разі відсутності необхідних матеріальних ресурсів. При формуванні графіка виробництва використовуються критерії: ступінь задоволення потреб виробництва в ресурсах; ефективність використання ресурсів; кошти, іммобілізовані в незавершеному виробництві; гнучкості.

Для формування виробничого розкладу з бази даних OPT використовуються файли замовлень, технологічних карт, ресурсів, прогнозів збуту та ін. Дані файли матеріалів і комплектуючих виробів обробляються паралельно з даними файлів технологічних карт, внаслідок чого формується технологічний маршрут, який обробляється за допомогою програмного модуля, який ідентифікує критичні ресурси. В результаті з'являється можливість оцінити інтенсивність використання ресурсів і ступінь їх завантаження та відповідним чином впорядкувати їх. На цьому етапі технологічний маршрут розгалужується. Гілка критичних ресурсів включає всі «вузькі» місця і подальші пов'язані з ними

логістичні активності. Після пошуку та виправлення помилок процес повторюється.

На рисунку 1.10 зображено схему функціонування комбінованої системи «KANBAN – MRP II».

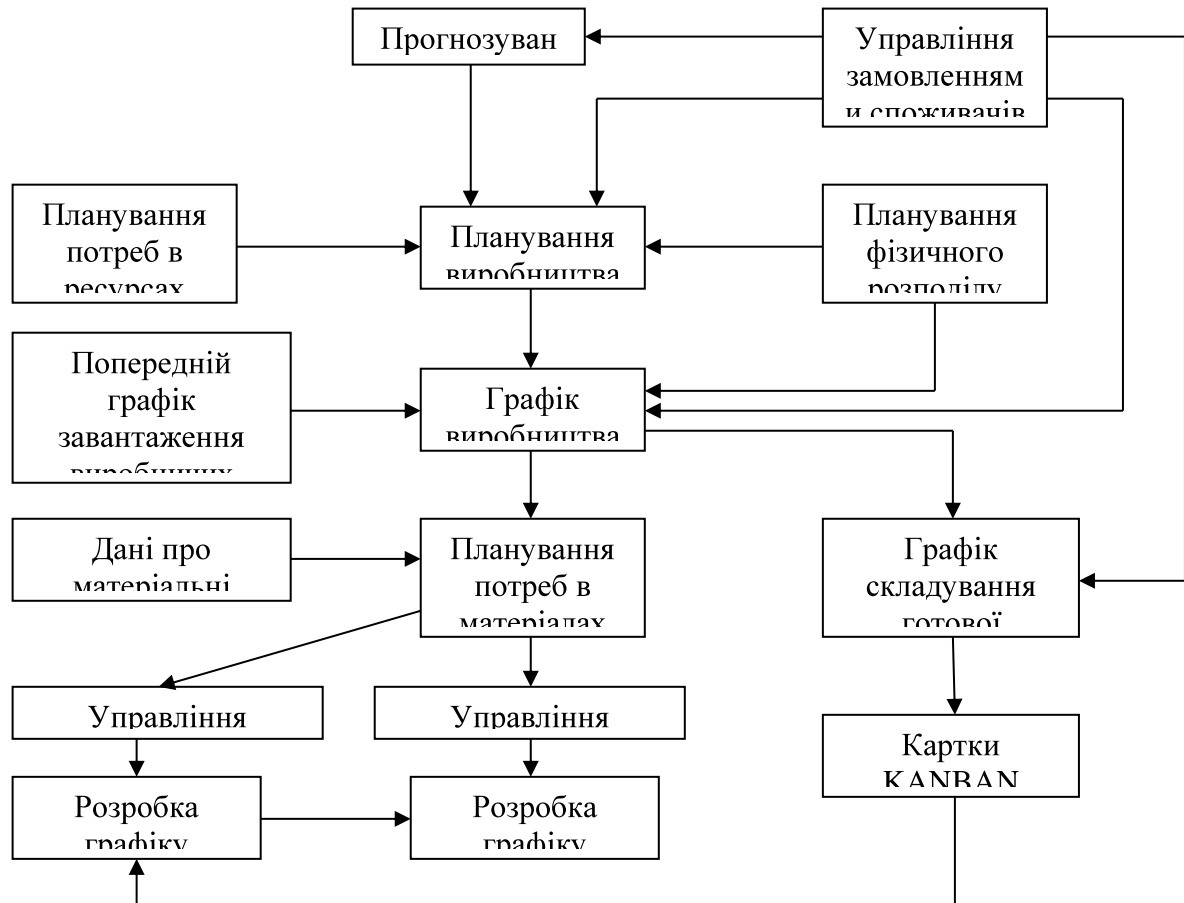


Рисунок 1.10 – Схема комбінованої системи «KANBAN – MRP II»

В таблиці 1.1 представлені широко-застосовані логістичні концепції для автоматизації процесу управління матеріальним потоком.

Таблиця 1.1 – Концепції автоматизації управління матеріальним потоком

Назва	Спосіб управління	Опис	Недоліки
MRP – планування потреб в матеріалах	Виштовхування замовлень	Автоматизована система планування потреб виробництва в необхідних матеріальних ресурсах. MRP дозволяє здійснювати пріоритетне планування виконання замовлень в реальному масштабі часу з урахуванням виникаючих відхилень від планових завдань, а також поточне регулювання і контроль виробничих запасів. Основними цілями MRP є: гарантійне задоволення потреби в матеріальних ресурсах, підтримання мінімально можливого рівня запасів; підвищення точності планування виробництва, поставок і закупівель матеріалів	— Великий об'єм розрахунків та попередньої обробки даних; — Зростання логістичних витрат на обробку замовлень і транспортування при прагненні фірми зменшити рівні запасів або перейти на випуск продукції в малих обсягах з високою періодичністю

			— Не чуттєвість до змін попиту; — Велика розмірність та складність системи.
DPR – планування розподілу ресурсів	Виштовхування замовлень	Автоматизована система управління вихідними товарами. Вона є дзеркальним відображенням MRP, використовує ту саму логіку, засоби та методи. До числа основних функцій системи належать: планування постачань і запасів на різних рівнях ланцюга розподілу (центрального - периферійні склади), інформаційне забезпечення розподілу продукції, а також планування транспортних перевезень. Основний графік виробництва формується у відповідності з даними незалежного попиту (прогнозу попиту). Таким чином, DPR дозволяє пов'язати функції виробництва і збуту продукції, а також оптимізувати логістичні витрати за рахунок скорочення транспортних витрат і витрат товароруку	— Потреба в точному та координованому прогнозі відправок та поповнень для кожного центру та в каналі розподілу графіку виробництва в логістичній системі; — Потреба в високій надійності здійснення логістичного циклу; — Часті зміни в виробничому завданні.
ЛІТ – управління матеріальними та інформаційними потоками за принципом «точно в час»	Витягування замовлень	Являє собою процес виробництва та пов'язані з ним постачання і збут як єдиний безперервний виробничий потік. Управління матеріальними потоками в єдиній системі здійснюється на основі зворотного планування термінів. Виробник не має завершеного плану та графіка роботи, він тісно пов'язаний не з загальним, а з конкретним замовленням споживача цієї продукції і оптимізує свою роботу в межах цього замовлення. Для всіх підрозділів розробляються тільки усереднені плани (на місяць), а їх деталізація по декадах (днях, годинах) проводиться безпосередніми виконавцями робіт з урахуванням термінів здачі деталей (складальних одиниць) та обсягу отриманого завдання	— Залежність від партнерів з поставок; — Зайві транспортні розходи; — Підвищення витрат на виробництво внаслідок невеликих партій випуску; — Нездатність справлятися з непередбаченими обставинами (поломки, страйки працівників, зрив поставок та ін.).
KANBAN – інформаційне забезпечення оперативного управління МП за принципом «точно в час»	Витягування замовлень	Картка замовлення «KANBAN» містить всю необхідну інформацію про запити споживача. Як правило, така інформація включає: найменування і шифр деталі; специфікацію ємностей із зазначенням їх типу та кількості укладаються в них деталей; найменування ділянки-виробника і ділянки-споживача продукції; час доставки, визначений з урахуванням тривалості виготовлення деталі. Кожен попередній по технологічній ланці ділянка працює у відповідності з отриманою замовленням, зазначеним у картці «KANBAN». Контроль за ходом виробництва здійснюється шляхом реєстрації карт, що перебувають в обігу.	— Важко забезпечити високу злагодженість між стадіями виробництва; — Великий ризик зриву виробництва та реалізації продукції.
OPT – оптимізована технологія виробництва	Витягування замовлень	Основним принципом роботи є виявлення у виробничому процесі вузьких місць або критичних ресурсів. OPT є комп'ютеризованою версією KANBAN з тією різницею, що система OPT перешкоджає виникненню вузьких місць у логістичній мережі «постачання - виробництво», а система KANBAN дозволяє ефективно усувати вже виникли вузькі місця. В якості критичних ресурсів, що роблять вплив на ефективність логістичної системи, можуть виступати запаси сировини і матеріалів, розмір незавершеного виробництва, технологія виготовлення, персонал та інші	— Висока вартість програмного, інформаційного та матеріально-технічного забезпечення; — «Заморожування» матеріальних ресурсів, установка зайвого обладнання.

		Підприємства, що використовують систему ОРТ, не прагнуть максимально завантажити персонал, що виконує некритичні операції, так як це викликає небажане зростання запасів незавершеного виробництва. Ефективність системи ОРТ з логістичних позицій полягає у збільшенні випуску продукції, зниженні виробничих і транспортних витрат, зменшенні запасів незавершеного виробництва.	
--	--	--	--

1.4 Аналіз проблем в управлінні матеріальними потоками за допомогою існуючих систем автоматизації

Характерною рисою сучасних систем автоматизації процесу управління ресурсами підприємства є централізація управління та глобальна оптимізація. В їх основі лежать методи дослідження операцій, імітаційного моделювання, методи PERT та інші методи мережевих розрахунків. Традиційні математичні алгоритми даних систем базуються на методах динамічного програмування та constraint-програмування [15]. Але на зважаючи на достатньо потужні механізми вирішення задач управління, в розглянутих системах, прийняття рішень все ж потребує великих за об'ємом розрахунків, попередньої обробки великої кількості даних та ручної доводки результатів.

При прийнятті рішень управлінець повинен врахувати велику кількість обмежень та вимог, які найчастіше задані в неявному вигляді, що призводить до збільшення трудомісткості вирішення задачі та обмежує застосування до неї класичних математичних алгоритмів.

Серед недоліків існуючих систем також можна відмітити, що такі системи мають вже вбудовану жорстку модель логістичних операцій, яку важко або взагалі неможливо змінити чи доповнити [15].

Системи розраховані на роботу в такому режимі, де спочатку заявки накопичуються, а потім в визначений час (наприклад на початку робочого дня) вводяться в систему та розплановуються (пакетний режим роботи). Така організація процесу ускладнює застосування розкладів в реальному часі та врахування непередбачених подій – наприклад поломки, запізнення, відміни заявок [7].

Розклади побудовані в таких системах вимагають обов'язкової ручної доводки. Для внесення навіть невеликих змін необхідно змінювати вхідні дані та заново з початку запускати програму, через що втрачається велика кількість часу [16, 17].

В традиційних системах неможливо врахувати індивідуальні особливості (стратегії, обмеження, переваги) окремих об'єктів, клієнтів чи замовлень. Не реалізуються «людські евристики» - підходи основані на досвіді управлінця. Програма не може подолати обмеження моделі, які можна було б обійти в силу їх необов'язковості. Так як традиційні методи не зберігають інформацію о неподоланих обмеженнях, планувальник не повертається до відкинутих варіантів, в тих випадках коли порушення обмеження було мінімальним [15].

Традиційно в класичному підході побудова розкладу «з нуля» передбачає процес послідовного планування, яке являє собою перебір варіантів у певному напрямку. Однак у складних моделях логістики кількість розкладів дуже велика, і для того, щоб побудувати розклад в розумний час, необхідно обмежити глибину пошуку, в результаті чого отриманий розклад виявляється неефективним. Повний перебір варіантів у таких умовах неможливий, тому більшість систем використовує підхід послідовного застосування обмежень. Такий спосіб, хоч і прийнятний, все ж малоефективний, оскільки передбачає перебір дуже великої кількості варіантів, які зрештою відсіваються в результаті накладання обмежень [15].

1.5 Дослідження можливості застосування МАС в управлінні матеріальними потоками

1.5.1 Аналіз останніх досліджень та досягнень в області застосування мультиагентних технологій для вирішення задач логістики

Дослідження застосування мультиагентних технологій для побудови автоматизованих інформаційних систем управління останнім часом набули значного поширення.

По-перше, необхідно відмітити основоположні теоретичні праці кінця дев'яностих років: різні класи агентів та методи їх взаємодії розглядалися в роботах С. Рассела та П. Норвига [22], В. Хорошевського та С. Городецького [8], В. Тарасова [18] та ряду інших авторів [20, 21]. Разом з тим, в цих роботах методам колективної та узгодженої взаємодії агентів у відкритих системах приділялося недостатньо уваги, а існуючі застосування охоплювали сфери е-комерції, інформаційного пошуку та деякі інші.

На початку двохтисячних років В. А. Виттихом, П. О. Скобелевим та Г. А. Ржевським [5] був розроблений апарат «мереж потреб та можливостей» (ПМ-мережі) для побудови самоорганізуючихся систем, який став основою теоретичних досліджень в області мультиагентних систем управління для ряду науковців: К. А. Аксенов [3, 14], А. В. Попов, Н. В. Гончарова [7], інші.

В серединні двохтисячних роботи набувають практичного значення. Необхідно відмітити дослідження П. О. Скобелева [15, 17], К. В. Івкушкіна [12], С. В. Батищева [1, 15] в області вирішення задач транспортної та розподільчої логістики за допомогою мультиагентного підходу. Апарат ПМ-мереж був програмно реалізований у вигляді технології MAGENTA, яка знайшла застосування в сімействі прикладних інтелектуальних систем планування [17]:

- МАС для управління танкерами для компанії Tankers International;
- МАС для управління вантажівками для компанії GIST;
- МАС для управління таксі для компанії Addison Lee;
- МАС для управління здачею машин в оренду для компанії Avis.

З кінця двохтисячних галуззю досліджень використання мультиагентних систем в області логістики стає управління виробничими цехами в реальному часі. Тут необхідно відмітити роботи А. В. Іващенко [10, 11], П. О. Скобелева [16], М. В. Андрєєва [2].

Результати впровадження розроблених мультиагентних систем [11]:

- Вирішується складне завдання автоматизації виробничого планування, контролю та прогнозування роботи ресурсів;
- Підвищується ефективність роботи підприємства;
- Забезпечується оперативна реакція на непередбачені події в реальному часі;

- Робота стає більш ефективною - роботи плануються та реалізуються на основі детального розрахунку, аналізу і зіставлення варіантів;
- План роботи є гнучким і відкритим до будь-яких змін «на льоту», що враховують індивідуальні особливості замовлень і ресурсів;
- План набагато краще відображає реальність, оскільки коректується в ході щоденного виконання робіт за фактом;
- Забезпечується повна прозорість роботи;
- Скорочується трудомісткість планування, контролю і прогнозування результату роботи підприємств;
- Підвищується дисципліна і відповідальність за результат;
- Зменшується людський фактор (помилки, викривлення даних і т.д.).

1.5.2 Дослідження технології МАС

Шляхом подолання визначених вище проблем є побудова автоматизованих логістичних систем на основі мультиагентних технологій. Мультиагентна система (МАС) – це система, яка основана на взаємодії декількох інтелектуальних агентів. Характерними рисами таких систем є [22]:

- Автономність: агенти в системі хоча б частково незалежні;
- Обмеженість представлення: агенти не мають представлення о всій системі в цілому, так як система може бути досить складна, щоб знання о ній мали б практичне застосування для агента;
- Децентралізація: нема агентів, які управляють всією системою.

МАС інтегрують в собі досягнення в сферах інформаційних технологій, штучного інтелекту, розподілених інформаційних систем, комп'ютерних мереж та комп'ютерної техніки.

Основні компоненти МАС [18]:

- Множина системних одиниць, в якій виділяється підмножина активних одиниць – агентів, які маніпулюють підмножиною пасивних одиниць – об'єктів;
- Середовище – деякий простір, в якому існують агенти та об'єкти;

- Множина задач (функцій, ролей), які поручаються агентам;
- Множина відносин (взаємодій) між агентами;
- Множина організаційних структур (конфігурацій), які формуються агентами;
- Множина дій агентів.

Інтелектуальний агент (ІА) – програмно або апаратно реалізована система, яка діє в інтересах досягнення цілі та має наступні властивості [8]:

- Автономність: агенти можуть функціонувати без прямого втручання користувача та мають здатність контролювати свої дії та внутрішній стан;
- Способи спілкування: взаємодія агентів при колективному рішенні задачі – агенти можуть обмінюватись інформацією з оточуючим середовищем чи іншими агентами;
- Реактивність: агенти можуть сприймати оточуюче середовище (яким може бути фізичний світ, користувач, інші агенти, чи все разом) и адекватно реагувати на зміни які в ньому відбуваються;
- Активність: здатність агента генерувати ціль і раціонально діяти для її досягнення;
- Інтелектуальна поведінка: поведінка агента включає здатність до навчання, логічної дедукції чи побудови моделі оточуючого середовища для того, щоб знаходити оптимальні способи поведінки.

В «сильному» понятті ІА, також володіє ще й такими властивостями як знання та обов'язки [8]:

- Знання. Це постійна частина знань агента про себе, середовищі і інших агентів, тобто та частина, яка не змінюється в процесі його функціонування;
- Зобов'язання по відношенню до іншим агентам. Завдання, які агент бере на себе за проханням (дорученням) інших агентів у рамках кооперативних цілей або цілей окремих агентів у рамках співробітництва.

Інші автори вважають [20, 21] що агенти повинні мати ще такі властивості: мобільність, раціональність, доброзичливість, правдивість:

- Мобільність. Здатність агента мігрувати по мережі в пошуках необхідної інформації для вирішення своїх завдань, при кооперативному вирішенні завдань спільно або за допомогою інших агентів і т.д.;
- Доброзичливість. Готовність агентів допомогти один одному і готовність агента вирішувати саме ті завдання, які йому доручає користувач, що передбачає відсутність у агента конфліктуючих цілей;
- Правдивість. Властивість агента не маніпулювати інформацією, про яку йому завідомо відомо, що вона помилкова;
- Раціональність. Властивість агента діяти так, щоб досягти своїх цілей, а не уникати їх досягнення, принаймні, в рамках своїх знань і переконань.

Також кожному агентові відповідають такі ментальні категорії як переконання, бажання і наміри (beliefs, desires, intentions – BDI) [7, 8, 20]:

- Переконання – уявлення про поточний стан середовища;
- Бажання – стан середовища, в який агент прагне середовище перевести;
- Наміри – безліч обраних, сумісних і досяжних бажань.

Розділяти наміри і бажання необхідно, так як агент може мати несумісні бажання або бажання можуть бути недосяжні. Оскільки агент обмежений у ресурсах і не може досягти всіх бажань одночасно, природно обирати найбільш значущі цілі – наміри.

Можна виділити три аспекти, що характеризують агентів: вони містять знання, вони відчують навколишнє середовище, вони виконують дії.

На рисунку 1.11 схематично зображено поведінку інтелектуального агента в системі (середовищі).

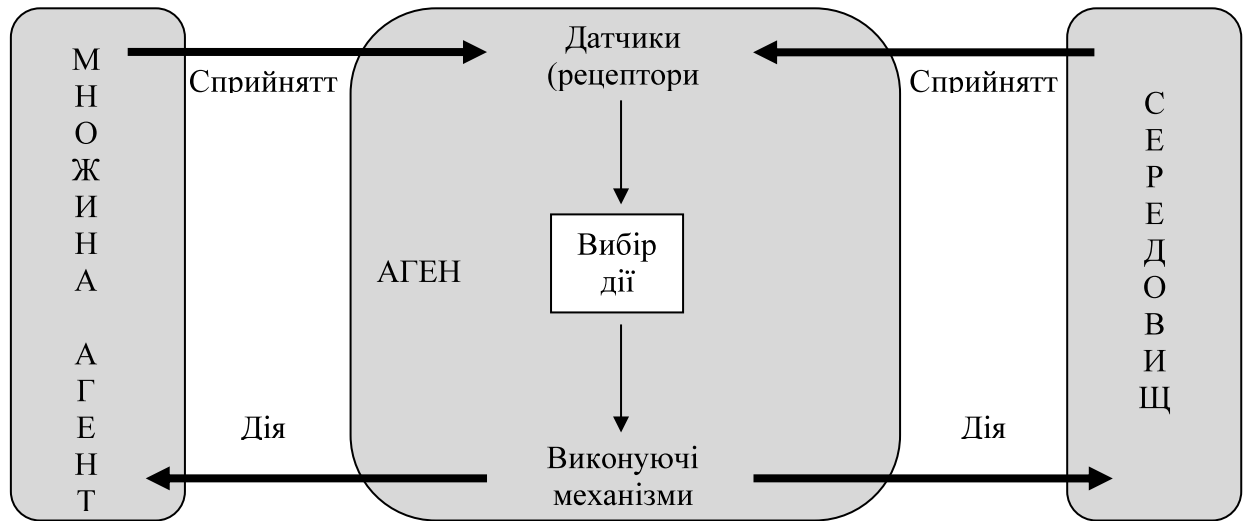


Рисунок 1.11 – Поведінка інтелектуального агента в системі

У загальній конструкції агента можна виділити три основних рівня (рисунок 1.12) [1]:

- Рівень комунікації. На цьому рівні виділяються компоненти і функціональні можливості підтримки комунікації між агентами. Класичний спосіб взаємодії агентів – обмін повідомленнями.
- Рівень даних. На цьому рівні ізолюються всі властивості і дані агентів: пам'ять, досвід, значення онтологічних атрибутів і т.д. Рівень даних підтримує управління історією змін: скасування змін, збереження змін як окремої сутності, застосування змін.
- Рівень поведінки (управління): рівень поведінки є активною складовою агента. На цьому рівні агента визначаються ролі агента. Роль агента – незалежна активна сутність, яка може:
 - Ініціювати відсилання повідомлення;
 - Чекати (знаходиться в стані очікування) і реагувати на повідомлення або подію зміни часу;
 - Отримувати інформацію, правила прийняття рішень і інші складні структури (цілі, стратегії) з рівня даних;
 - Змінювати інформацію в рівні даних;
 - Реагувати на зміну інформації в рівні даних.

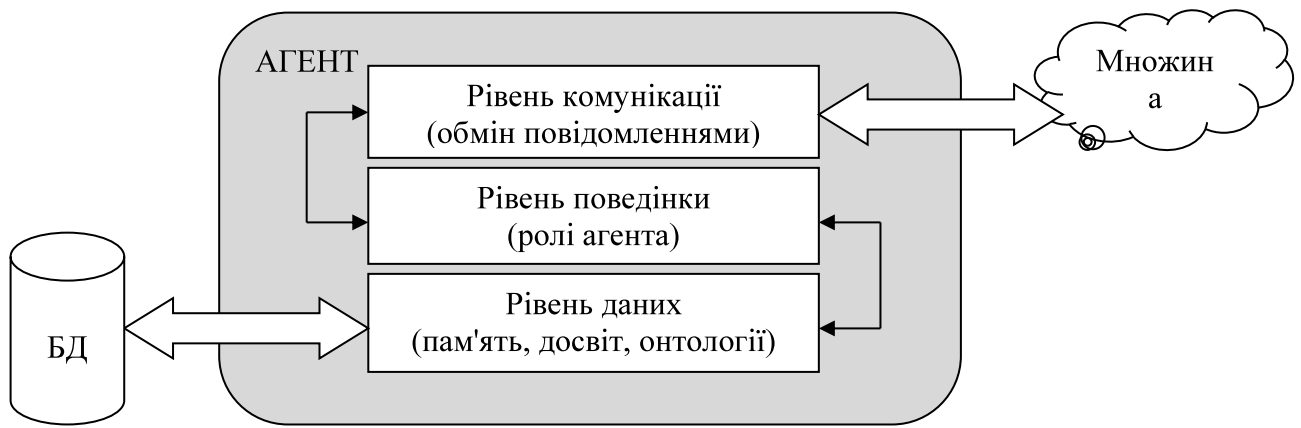


Рисунок 1.12 – Конструкція агента

Активність агента в системі узагальнено можна зобразити за допомогою діаграми станів (рисунок 1.13).

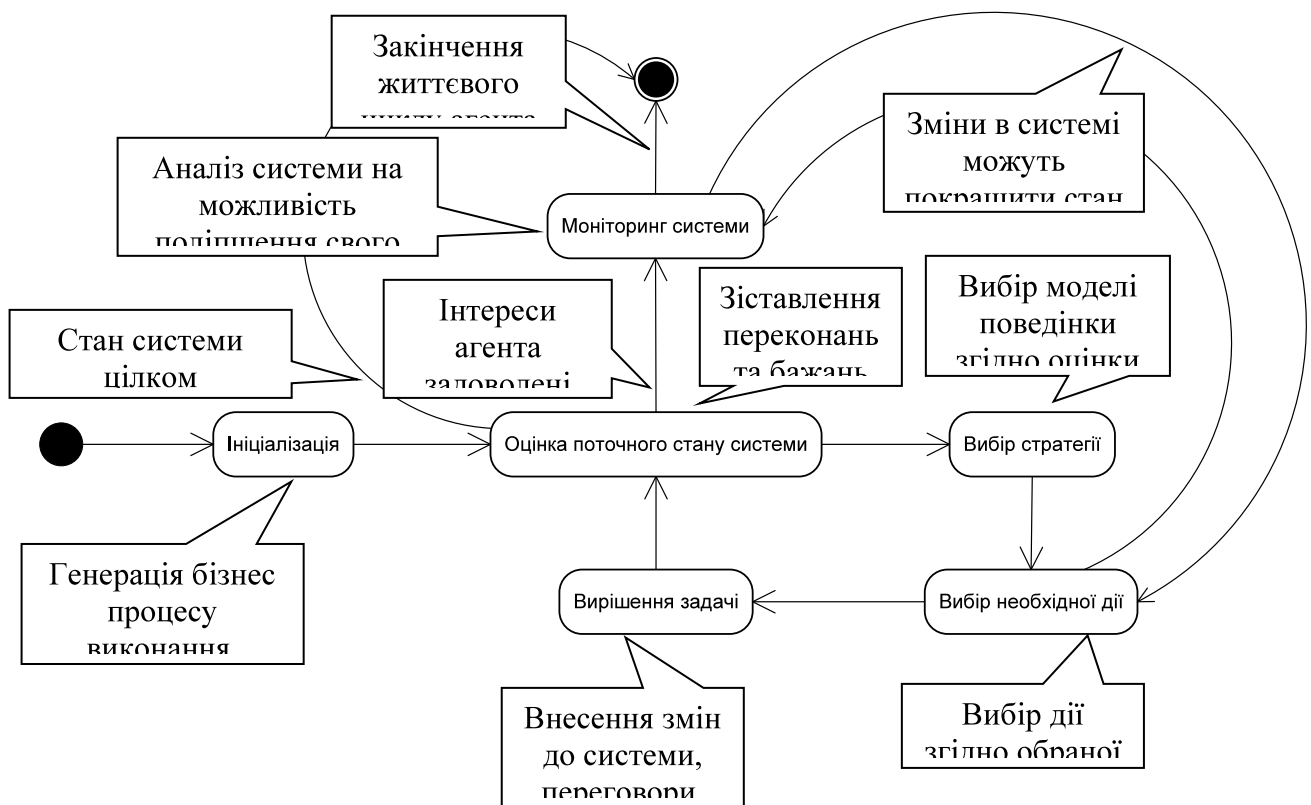


Рисунок 1.13 – Активність агента

В залежності від концепції, прийнятої при розробці МАС, можливі різні варіанти її архітектури, серед яких виділяють три базових типи [20]:

— Архітектури, засновані на методах роботи зі знаннями;

- Архітектури, в яких використовуються поведінкові моделі «стимул-реакція»;
- Гібридні архітектури.

В архітектурі першого типу для подання та обробки знань використовуються традиційні моделі, методи та засоби штучного інтелекту, а прийняття рішень здійснюється на основі механізмів формальних міркувань. В перших системах такого типу для подання та обробки знань використовувалася логіка предикатів першого порядку. Розвиток досліджень в цій сфері призвело до появи спеціальних розширень логічних числень, орієнтованих на врахування таких властивостей агентів, як переконання, бажання, наміри та зобов'язання. Основний недолік архітектури першого типу – складність або принципова неможливість побудови достатньо повних баз знань, які є необхідною частиною створюваних систем. Зокрема, інтелектуальний агент може мати архітектуру типовою для продукційної системи, яка здатна сприймати інформацію із зовнішнього середовища і здійснювати ті чи інші дії в результаті обробки цієї інформації. Головні відмінності агента від звичайної продукційної експертної системи пов'язані з наявністю механізму формування цілей і модуля комунікації, який забезпечує взаємодію з іншими агентами. Агент з такою архітектурою здатний до міркувань, але не здатний до навчання. Адаптивну поведінку агента дозволяє реалізувати архітектура на основі класифікацій систем Холланда. Найважливішими відмінностями класифікуючих систем від продукційних є: можливість формування нових правил із застосуванням генетичного алгоритму та наявність механізму заохочень.

В архітектурах другого типу, які називають реактивними, не використовуються традиційні для штучного інтелекту символічні моделі подання знань. Моделі поведінки агентів представлені або наборами правил, які дозволяють вибрати дію, відповідну ситуації, або кінцевими автоматами, або іншими засобами, що забезпечують формування адекватних реакцій агента на виникаючі у системі стимули. Системи цього типу, як правило, мають високий ступінь спеціалізації і суворі обмеження на складність вирішуваних завдань.

Найбільш перспективними вважаються гібридні інтелектуальні мультиагентні системи, які дозволяють використовувати можливості

інтелектуальних і реактивних архітектур. Прикладом може служити архітектура з ієрархічною базою знань, яка містить структуровану базу знань, робочу пам'ять, модуль управління комунікацією і людино-машинний інтерфейс. Агент з подібною архітектурою має здатність до міркувань і реактивної поведінки. Його база знань містить три рівня:

- знання предметної області;
- знання про взаємодії, які дозволяють приймати рішення в умовах невизначеності;
- керуючі знання.

Інтелектуальна поведінка агента забезпечується здатністю приймати рішення, а реактивна - системою контролю за вмістом робочої пам'яті, яка функціонує за принципом глобальної дошки оголошень. Агент взаємодіє з користувачем, використовуючи людино-машинний інтерфейс. У загальному випадку гібридні архітектури є багаторівневими і відрізняються один від одного структурою і змістом рівнів, які можуть відповідати різним рівням управління, абстракції або окремим функціональним властивостям агента.

1.5.3 Характеристика мультиагентного підходу до вирішення задач управління матеріальними потоками

В мультиагентному підході усім об'єктам, які задіяні в процесі управління матеріальним потоком (вантажі, склади, транспортувальні засоби, тощо), ставиться у відповідність свій інтелектуальний агент, який діє в інтересах свого об'єкту, використовуючи різні стратегії [12]. Інтелектуальні агенти вступають в взаємодію як і з навколишнім середовищем так і один з одним – обмінюються інформацією та вступають в переговори, находячи компроміс. Такий підхід сприяє всебічному вирішенню задач управління матеріальним потоком.

В якості базових агентів використовуються два типа: агент замовлення та агент ресурсу [15]. Кожен агент замовлення повинен найти ресурс, який його виконає максимально підходящим для нього способом, і навпаки, кожен агент ресурсу прагне найбільш вигідне для себе замовлення. У процесі побудови

розкладу «з нуля» усі ресурси вільні і жодне замовлення не обслуговане, тому кожне замовлення може обрати собі той ресурс, який йому більше підходить. Однак скоро вільні ресурси закінчується, і замовлення починають конфліктувати. Для вирішення конфліктів замовлення та ресурси вступають в переговори з метою знайти компромісний варіант. Особливо слід відзначити, що об'єднання індивідуальних інтересів об'єктів в кінцевому підсумку відображає інтереси всієї системи, тобто формується розклад, що задовольняє умовам системи і, нехай не обов'язково оптимальний, але перебуває в стані внутрішньої гармонії, оскільки побудовано на взаємному поєднанні інтересів [13, 15]. Таким чином, планування не являє собою послідовний перебір варіантів, як в класичних системах, а пошук йде «з усіх сторін», в результаті чого отриманий розклад не страждає «однобокістю». Крім того, такий підхід дозволяє на рівні стратегій агентів підтримати «людські евристики», які неможливо врахувати при послідовному переборі варіантів. Практика показує, що у виконанні будь-якого плану завжди відбуваються збої, і тому навіть план, складений оптимальним чином, але який не передбачає вірогідних змін, гірше і менш ефективний, ніж план, не настільки оптимальний, але який дозволяє внести без великого збитку деякі зміни, викликані непередбаченими обставинами [13].

Мультиагентні системи забезпечують можливість адаптивної побудови і виконання планів, коли план не будується щоразу заново при виникненні непередбачених подій, а лише коректується [15].

Мультиагентний підхід дозволяє побудувати децентралізовані системи, які орієнтовані на досягнення локальних оптимумів, шляхом виявлення конфліктів і досягнення компромісів з допомогою переговорів агентів. Типовий приклад переговорів агентів можна зобразити наступним чином [12]: надходить замовлення на деталь, і тут же створюється агент замовлення для проведення переговорів про прийняття його у виробництво. Агент замовлення знаходить вже існуючу готову деталь на одному зі складів. Він пропонує певну ціну за дану деталь, а агент деталі, в свою чергу, може прийняти цю пропозицію або відмовитися від нього в залежності від запропонованої ціни. Припустимо, що агент деталі приймає пропозицію і деталь резервується. Припустимо, що далі в систему надходить інше замовлення, і другий агент замовлення намагається

зарезервувати для себе ту ж деталь, оскільки іншої підходящої не знаходить. Розуміючи, що деталь вже зайнята, агент замовлення пропонує відмінити бронювання за певну компенсацію. В результаті агент деталі знову звертається до першого агента замовлення з пропозицією прийняти компенсацію. Перший агент замовлення може погодитися і прийняти компенсацію, якщо тільки це не вплине на терміни його виконання, або дата поставки може бути перенесена на пізніший термін. Якщо перший агент замовлення згоден, то деталь перебронюється заново, тепер вже для другого замовлення.

В ході переговорів агенти можуть реалізовувати різні стратегії. Наприклад [12], «обережний» агент замовлення послідовно обходить весь «ринок» пропозицій виробників, торгується з ними і в результаті знаходить найкращу пропозицію (яке, втім, до моменту прийняття рішення вже може бути «зайнято»), а «рішучий» - негайно бронює першу ж пропозицію, якщо вона задовольняє його вимогам. У підсумку, вступаючи в переговори, агенти можуть конкурувати між собою, так і кооперуватися для досягнення заданої мети.

На рисунку 1.14 представлена узагальнена схема мультиагентної системи управління.

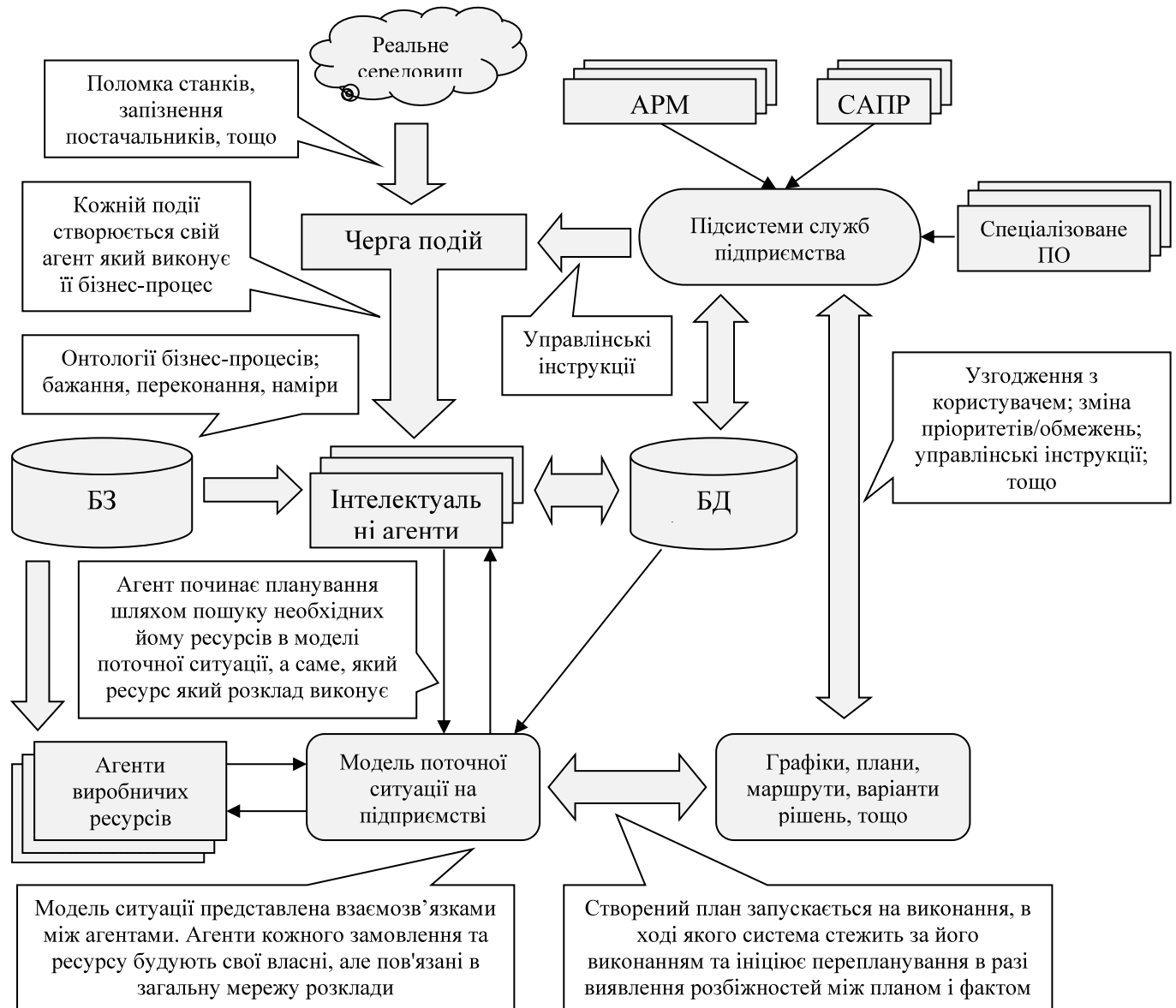


Рисунок 1.14 – Мультиагентна система управління

Крім множини агентів замовлень і готових об'єктів (ресурсів), які постійно шукають зустрічі, мультиагентна система може передбачати і ряд інших більш традиційних додаткових агентів, наприклад: агенти виробника, складу, конвеєра та інші. Агент складу може наділятися такими функціями: стягує з замовлень плату за зберігання частин і постійно до-замовляє частини, постійно оцінюючи співвідношення між мінливим попитом та пропозицією. Агент конвеєра управляє критеріями складання і при необхідності може перемикає переговори замовлень з отримання максимального прибутку, наприклад, на резервування часу з ремонту складальної лінії.

Узагальнено поведінку мультиагентної системи управління можна зобразити за допомогою діаграми станів (рисунок 1.15).

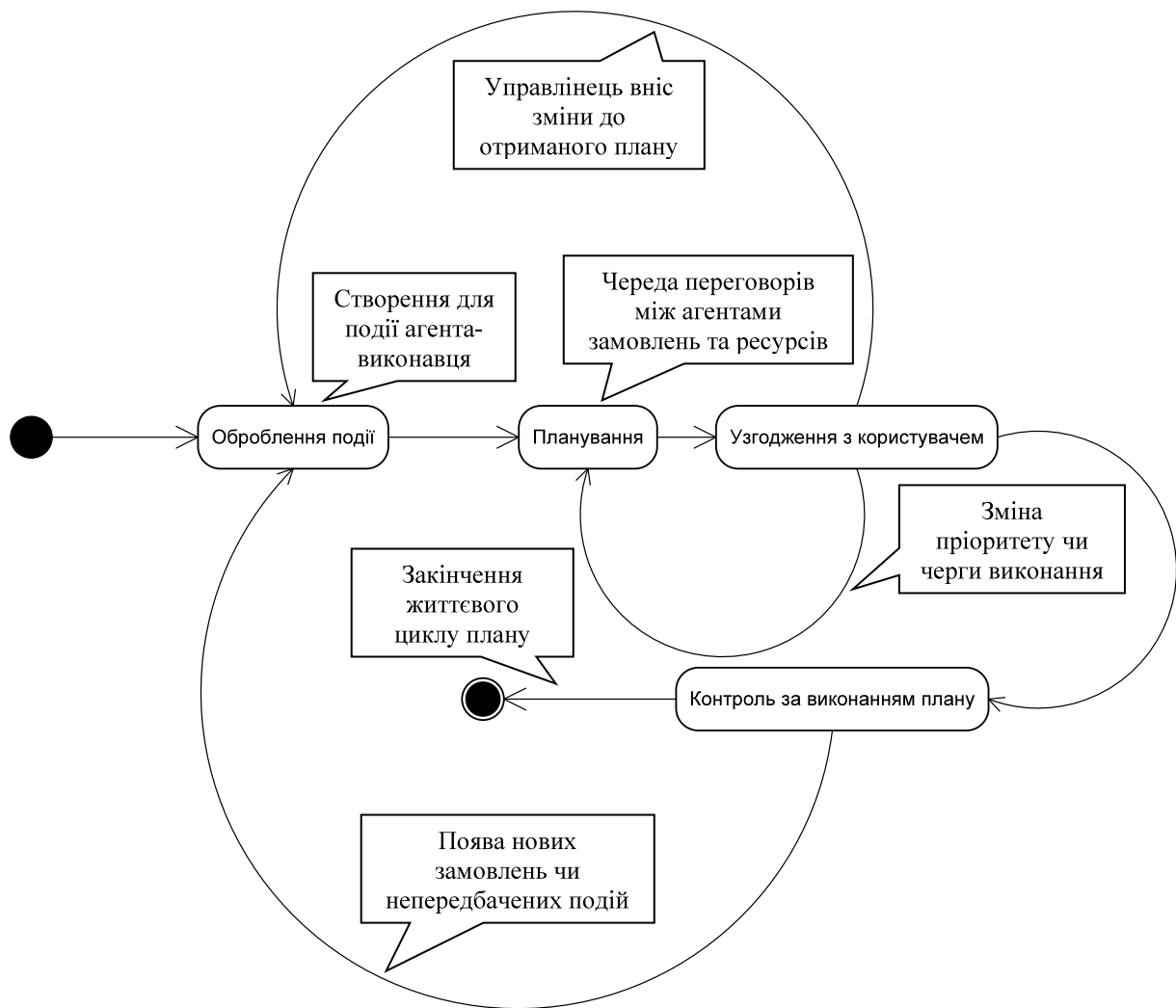


Рисунок 1.15 – Поведінка МАС управління

1.5.4 Порівняльний аналіз мультиагентного підходу управління матеріальними потоками

Отже, на основі проведеного в роботі дослідження можна стверджувати, що використання мультиагентного підходу в управлінні матеріальними потоками підприємства має ряд своїх переваг:

- збалансований підхід до процесу планування та побудови розкладів;
- врахування індивідуальних вимог на рівні кожного окремо взятого замовлення та ресурсу;
- використання в процесі планування стратегій, інтуїтивно використовуваних людиною на базі досвіду;

— можливість ефективної і швидкої реакції на непередбачені події.

Принципову різницю між класичними та мультиагентними системами можна продемонструвати зобразивши їхню узагальнену структуру (рисунок 1.16):

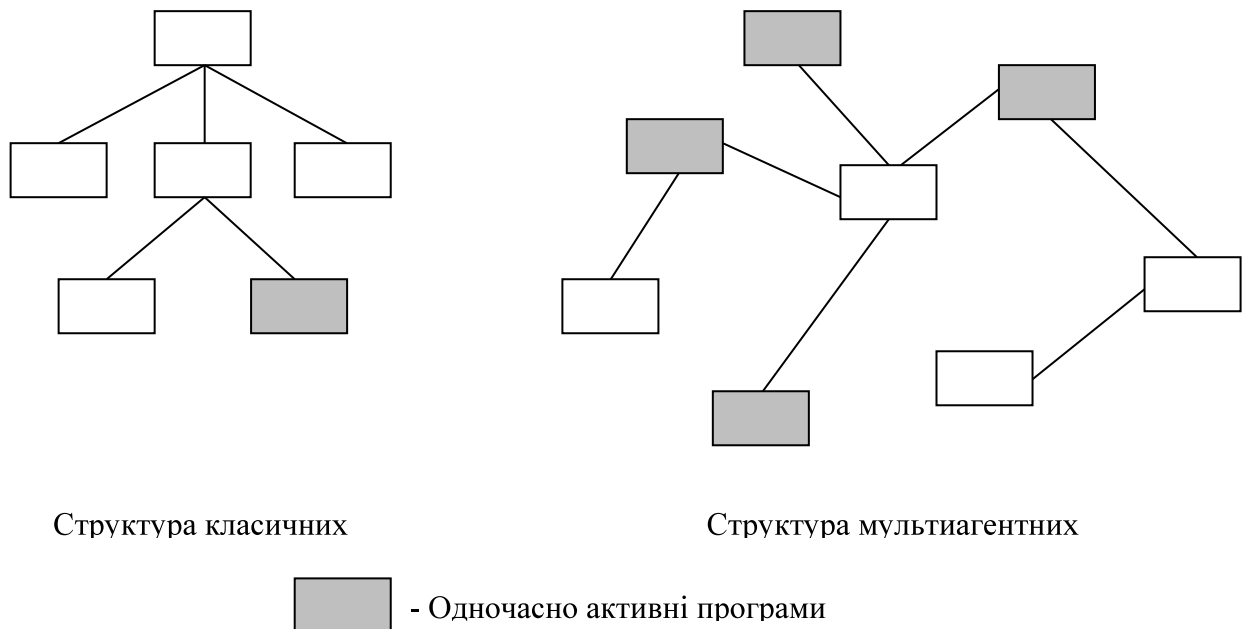


Рисунок 1.16 – Порівняння структур класичних та мультиагентних систем

Принципова різниця в структурі класичних та мультиагентних систем породжує цілий ряд відмінностей між системами. Дослідники пропонують такий варіант порівняння класичних та мультиагентних систем на основі різниці в їх структурах (таблиця 1.2) [17]:

Таблиця 1.2 – Порівняння традиційного і мультиагентного підходу

Класичні системи	Мультиагентні системи
Ієрархії великих програм	Великі мережі малих агентів
Послідовне виконання операцій	Паралельне виконання операцій
Інструкції з гори до низу	Переговори
Централізовані рішення	Розподіленні рішення
Управляються даними	Управляються знаннями
Передбачуваність	Самоорганізація
Стабільність	Еволюція
Прагнення зменшення складності	Прагнення нарощування складності
Тотальний контроль	Умови для розвитку

В подальшому порівнянні з класичними системами можна виділи ще ряд характерних особливостей мультиагентних систем. Окрім відмінностей викликаних принципово різною структурою систем, мультиагентні системи характеризуються принципово іншими як і підходами так і механізмами вирішення задач, що призводить до значного розширення області застосування, але це не виключає породження нових недоліків і нових шляхів їх подолання (таблиця 1.3) [13].

Таблиця 1.3 – Порівняння класичних та мультиагентних систем

Критерій	Класичні системи	Мультиагентні системи
Основні характеристики	Передбачуваність, повторюваність, ієрархічна структура	Гнучкість, чуттєвість, самоорганізація
Механізми досягнення основних характеристик	Класичні математичні алгоритми, пам'ять, інтеграція	Можливість будувати припущення, знання, навчання, робота в мережі
Основні недоліки	Жорсткість	Ризик здійснити помилку
Механізм вирішення недоліків	Модуляції	Розподілення інтелекту, навчання на опиті, повноцінне використання знань
Області застосування	Стабільні середовища, масове виробництво, довгострокове виробництво	Непередбачене середовище, часто змінювані виробництва, індивідуальні виробництва, короткострокові періоди освоєння нової продукції

«Основні характеристики» викликані основою систем – їхньою структурою. Ієрархічна структура класичних систем дає можливість очікування та передбачення результатів їх роботи. Мережева структура мультиагентних систем навпаки викликана самоорганізацією елементів системи. Самоорганізація робить систему більш гнучкою і як наслідок більш чуттєвою на зміни різного роду як і в навколишньому середовищі так і в середині системи. Ієрархічність та передбачуваність класичних систем визначають механізми вирішення їх задач («механізми досягнення основних характеристик»). В системах з такими

характеристиками доречно використовувати класичні математичні алгоритми та інтеграцію елементів системи спеціалізованих на вирішенні задач конкретної області. Мережева структура мультиагентних систем не дає елементам системи (агентам) представлення о всій системі в цілому, тому агенти діють роблячи припущення, навчаючись та спілкуючись з іншими елементами системи. Такий підхід дає можливість роботи в умовах невизначеності, роботи з неповними, некоректними даними. Механізми вирішення задач є і причиною «основних недоліків» систем. Класичні алгоритми, передбачуваність, ієрархічність класичних систем вимагають роботу з повними, стабільними даними, що породжує жорсткість таких систем. Обмеження представлення про систему та достатня свобода дій елементів мультиагентних систем породжує ризик здійснити помилку – так як мультиагентні системи будують не оптимальні плани, а плани приближені до оптимальних з задоволенням інтересів всіх об'єктів системи задля досягнення внутрішньої гармонії. «Механізмом вирішення недоліків» в класичних системах є модуляції – коректування та зміна параметрів/вхідних даних системи. В мультиагентних системах недоліки вирішуються розподіленням інтелекту та навчанням. Отже, такі характеристики систем і визначають «область їх застосування». Класичні системи знаходять застосування в стабільних середовищах – масове/довгострокове виробництво. Мультиагентні системи можуть діяти в умовах невизначеності, в непередбачених середовищах – наприклад індивідуальне виробництво.

Отже, основуючись на приведених вище порівняннях класичних та мультиагентних систем та враховуючи їхні особливості з перевагами та недоліками можна представити звідну таблицю порівняння з визначенням переваг які отримує підприємство від мультиагентного підходу автоматизації управлінських процесів (таблиця 1.4):

Таблиця 1.4 – Мультиагентний підхід в порівнянні

Класичні системи	Мультиагентні системи	Переваги для підприємства
Намагання зменшити складність	Врахування реальної складності виробництва	Отримання більш ефективних та адекватних рішень
Пакетний режим роботи	Робота по подіям в реальному часі	Зменшення дефіциту, простою, холостого пробігу, тощо
Великий набір критеріїв для кожного замовлення та ресурсу	Індивідуальний підхід до кожного замовлення та ресурсу	Можливість побудови більш живучих планів

Прагнення загальної типізації та інструкції «зверху вниз»	Самоорганізація	Більша продуктивність та задоволеність всіх учасників
Комбінаторна оптимізація	Пошук балансу інтересів	Злагодженість рішень та їх гнучка адаптація по ситуації
Підтримання жорстких бізнес-процесів	Підтримання мережових взаємодій	Природність і злагодженість в роботі
Необхідна висока точність даних	Підтримання неповних, некоректних, збійних даних	Відповідність до реальної ситуації

Однак, незважаючи на усі переваги мультиагентних систем, нові можливості мультиагентного підходу приводять до ряду нових проблем:

- В умовах постійних змін важко оцінити наскільки поточне рішення системи відрізняється від «оптимального»;
- Рішення системи залежить від історії виникнення подій;
- «Ефект метелика»: невеликі зміни на вході можуть призвести до, неочікуваних для спостерігача, великих змін на виході;
- Реакція системи може непередбачено сповільнюватися для спостерігача у разі виникнення довгого ланцюга змін та переговорів (потрібен контроль активностями в умовах обмеженого часу на відповідь);
- При повторному запуску системи, при тих самих вхідних даних рішення системи може бути іншим;
- Неможливо «відкотити» рішення назад, так як ситуація кардинально змінюється;
- Рішення системи іноді буває складно пояснити користувачу, в силу складних зв'язків та залежностей між агентами.

2 ВИБІР ПРИКЛАДНОЇ ЗАДАЧІ ТА РОЗРОБКА ПРОЕКТУ МУЛЬТИАГЕНТНОЇ СИСТЕМИ ДЛЯ ЇЇ ВИРІШЕННЯ

2.1 Вибір прикладної задачі

В якості прикладної задачі для розробки проекту мультиагентної системи було обрано задачу автоматизації процесу управління матеріальними потоками підприємства спеціалізованого на розкрії листових матеріалів. Обрана галузь характеризується умовами невизначеності та високою динамікою в прийнятті управлінських рішень.

Перш за все, серед проблем обраної галузі відмічається складність ведення закупок матеріалів на підприємство. Менеджеру по закупкам необхідно балансувати між великими плановими закупками у виробника та малими закупками під замовлення на місцевих оптових базах. В першому варіанті матеріали закупаються по мінімальним цінам виробника великими партіями періодично, що обумовлює відносно великі витрати на транспортування (так як склади виробника можуть бути достатньо віддалені від підприємства). Але великі партії можуть не виправдатись, так як в даній галузі складно передбачити попит в силу індивідуальності замовлень на підприємстві. Передбачення попиту ускладнює також великий асортимент матеріалів, з якими працює підприємство, так як листи з одними і тими самими параметрами (текстура, фактура, колір, тощо) але від різних виробників відрізняються одні від одних та не є взаємно замінюваними. Другий варіант ведення закупок передбачає закупівлю матеріалів під замовлення по трохи дорожчим оптовим цінам але з меншою ціною за транспортування (так як місцеві оптові бази знаходяться поруч з підприємством і об'єм вантажу невеликий). Але цей варіант теж себе не виправдовує, так як замовлення за об'ємом відносно невеликі, а за частотою достатньо інтенсивні, що призводить до великих транспортних затрат.

Проблеми іншого роду виникають через характерні риси замовлень які поступають на підприємство. Треба відмітити що замовлення на підприємство поступають достатньо інтенсивно і з непередбаченою динамікою. Замовлення різні за об'ємом, складністю та індивідуальні за конструкцією і специфікацією.

Така характерна інтенсивність та індивідуальність значно ускладнює процес побудови раціональних планів та графіків виробництва.

Визначені умови ведення процесу управління матеріальним потоком виробництва вимагають високої оперативності в обробці/передачі інформації та значно знижують час на прийняття рішення.

Для обраної галузі також характерна проблема роботи з залишками матеріалу. Зазвичай замовнику доводиться платити за весь лист матеріалу, хоча на замовлення необхідно використати лише частину його площі. Звісно залишок матеріалу повертається замовнику, але в більшості випадків він йому просто непотрібен. І складається така ситуація де над нікому непотрібним залишком приходится проводити різні операції, а він не тільки займає місце, вантажомісткість, але й сили, час та ресурси робітників. Проблеми роботи з залишками після виробництва:

- Ускладнення складського обліку залишків в силу їх нестандартної форми;
- Подальше накопичення залишків та несанкціоноване їх використання;
- Перевантаження транспортної системи в середині виробництва;
- Використання залишків з попереднього замовлення в наступному ускладнює задачу побудови раціональних та ефективних виробничих розкладів.

В умовах жорсткої конкуренції вирішення задачі скорішого ефективного подальшого використання залишків виробництва стає принциповим.

Отже, в якості предмета автоматизації, для доцільного застосування мультиагентних технологій, було обрано процес побудови та управління виконавчими графіками підприємства.

2.2 Аналіз діяльності обраного підприємства

Підприємство в своїй діяльності спеціалізується на розкрої листових матеріалів (деревостружкові плити, метал, скло, пластик, тощо) та на застосуванні меблевої фурнітури (допоміжних частин та деталей необхідних для

виготовлення кінцевого продукту: ручки, замки, напрямні, інше). Матеріали на підприємство закуповуються як планово у виробника, так і по ситуації на місцевих оптових базах. Деревостружкові плити (листи) характеризуються лінійними розмірами (кожен виробник виготовляє стандартні листи зі своєю висотою, шириною, та лінійкою розмірів товщини), кольором, текстурою – структурою поверхні листа, та фактурою – малюнком, який імітує деяку обрану породу дерева.

Замовлення на розкрій листів поступають від магазинів-представництв підприємства та магазинів-партнерів, які можуть знаходитись як і в одному місті з підприємством так і в сусідніх.

Структурно підприємство можна розділити на проектні, виробничі та облікові підрозділи (рисунок 2.1):

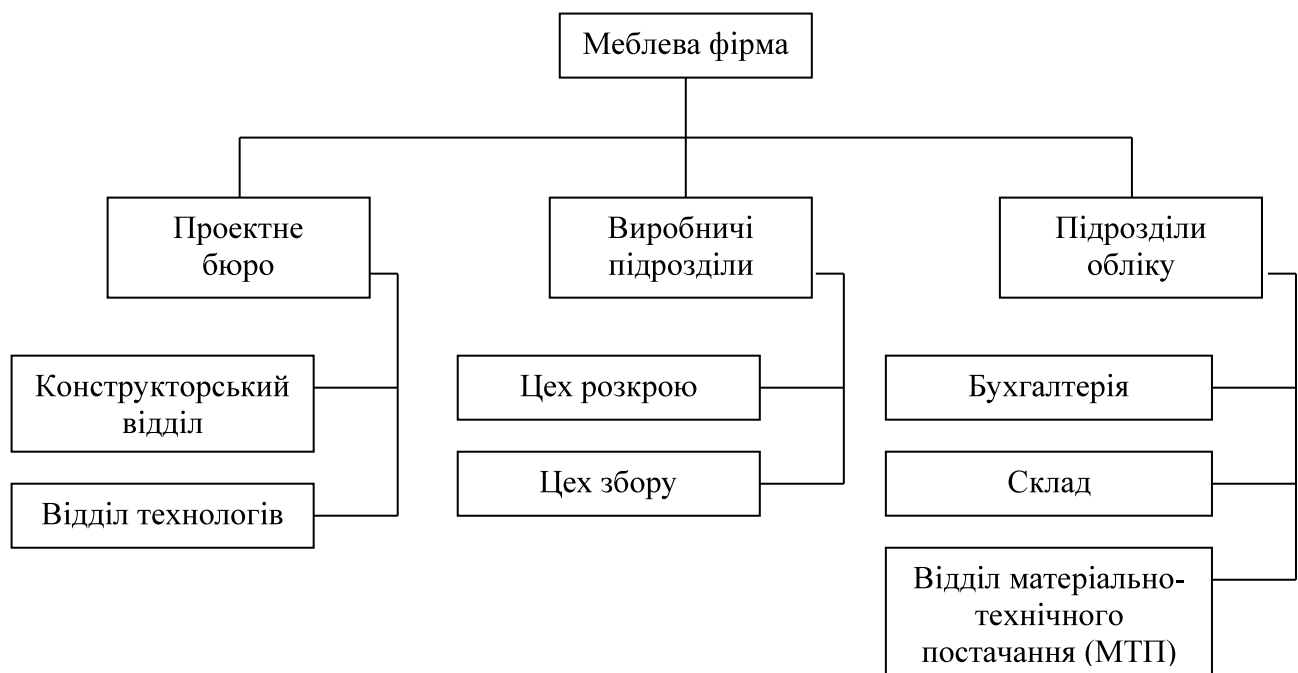


Рисунок 2.1 – Структурна схема підприємства

Визначені структурні одиниці вирішують на підприємстві наступні завдання:

Конструкторський відділ:

- узгодження з замовником технічного завдання (ТЗ);
- розробка дизайн-проекту та креслення продукції;
- розробка специфікації на матеріали та обладнання.

Відділ технологів:

- створення карт розкрою та свердлень;
- контроль за виконанням робіт по розкрою та зборці.

Виробничі підрозділи:

- підготовка виробництва;
- одержання необхідних матеріалів та фурнітури зі складу;
- безпосереднє виконання робіт по розкрою та збору кінцевої продукції;
- звіт по виконанню робіт, їх якості, та затрачених ресурсів;
- видача готового виробу та залишків.

Бухгалтерія:

- облік та контроль фінансових операцій підприємства;
- облік надходження та розходу матеріалів.

Склад:

- прийом, видача, розміщення, зберігання та облік матеріалів;
- звіт по прийому, видачі та залишку матеріалів.

Відділ МТП:

- вибір постачальників, узгодження цін на матеріали та обладнання;
- закупівля необхідних матеріально-технічних ресурсів;
- формування графіку закупок.

Початковим етапом процесу виробництва стає робота конструктора-дизайнера з замовником. Після чого підприємство отримує від замовника платіж на підготовку та старт виконання замовлення (закупку матеріалу, тощо).

Робота починається з вимірів приміщення. Перед початком вимірів конструктор з'ясовує, проектування яких меблів планує замовити клієнт, їх розташування, розміри, специфічні вимоги до конструкції, а також які матеріали будуть використовуватися. Якщо замовник не орієнтується в сучасних матеріалах, що використовуються для виробництва меблів, то конструктор знайомить його з ними згідно з плановим асортиментом складу підприємства.

Конструктор фіксує габаритні розміри приміщення, розміщення дверних і віконних прорізів та розташування електророзеток. При проектуванні кухонних

меблів додатково заміряється розташування водяних, каналізаційних і газових комунікацій і обладнання та приладів які їх обслуговують.

Після всіх замірів приміщення, конструктор складає технічне завдання на проектування меблів.

На наступному етапі згідно з затвердженим технічним завданням конструктор-дизайнер створює проект замовленого виробу. Розроблюється конструкція майбутнього виробу з урахуванням його розмірів і місця розташування. Опрацьовується запас міцності виробу і його безпечність в експлуатації. Далі, враховуючи побажання замовника, розміщуються всілякі елементи виробу: полиці, висувні ящики та інше насичення, визначаються їх функціональні розміри. При цьому враховується зручність у використанні як окремих елементів, так і виробу в цілому. Після створення моделі виробу, опрацьовуються колірні та текстурні рішення, враховуючи загальний дизайн приміщення. Якщо меблі складається з декількох виробів, то вибирається їх оптимальне розташування з урахуванням ергономічності і функціональності.

Отже, результатом роботи конструктора стає наступна технічна документація:

- технічне завдання;
- загальне креслення виробу;
- креслення окремих збірних одиниць;
- креслення по зборці;
- специфікація по деталям та матеріалам;
- фурнітурний лист.

Конструкторські креслення зазвичай розроблюється в програмах типу 3D Max Studio чи AutoCAD та передаються до відділу технологів. У відділі технологів, на основі створених конструктором креслень та специфікацій, розроблюється:

- карти розкрою матеріалу;
- карти нанесення кромкового матеріалу на торці деталей;
- креслення фрезерування криволінійних деталей;
- карти свердлень деталей під кріплення.

Для розробки технічної документації технолог використовує спеціалізоване програмне забезпечення – програму автоматичного розкрою, яка забезпечує алгоритми оптимального розміщення деталей на листі, розрахунок мірних залишків та набір функцій для оформлення карт розкрою та специфікацій. Підібрані технологом листи для розкрою резервуються під замовлення та транспортуються зі складу в виробничі підрозділи. Для цього технолог формує «вимогу-накладну» де зазначається перелік та об'єм необхідних матеріалів. «Вимога-накладна» у трьох екземплярах передається до відділу МТП. Якщо матеріали присутні на складі «вимогу-накладну» затверджують та видають технологу «забірну картку» з фактичною інформацією о матеріалах які передаються в виробництво. Один екземпляр «вимоги-накладної» залишається у відділі МТП, другий передається на склад а третій до бухгалтерії. Після створення карт розкрою та свердлень уся технічна документація по проекту та «забірна картка» на матеріали передається в виробничі підрозділи. Якщо матеріали, які зазначені в «вимозі-накладній» відсутні на складі чи в відділі МТП було прийнято рішення не використовувати резерв складу, менеджер по ТМЦ замовляє матеріали у постачальника, а «забірна карта» затверджується на дату поставки, яка вказана в договорі між постачальником та підприємством. Обробляючи інформацію з договорів відділ МТП формує графік планових поставок та веде графік надходжень матеріалів по факту.

Одержавши зі складу по «забірній картці» необхідні матеріали та фурнітуру, виробничі підрозділи керуючись розробленою технічною документацією по проекту закінчують виготовлення виробу. Координує та організовує виробничий процес майстер цеху – розподіляючи наряди на розкрій між виробничими дільницями/станками та віддаючи інструкції по руху матеріалів. Технологічний процес виготовлення замовлення можна зобразити наступним чином (рисунок 2.2):

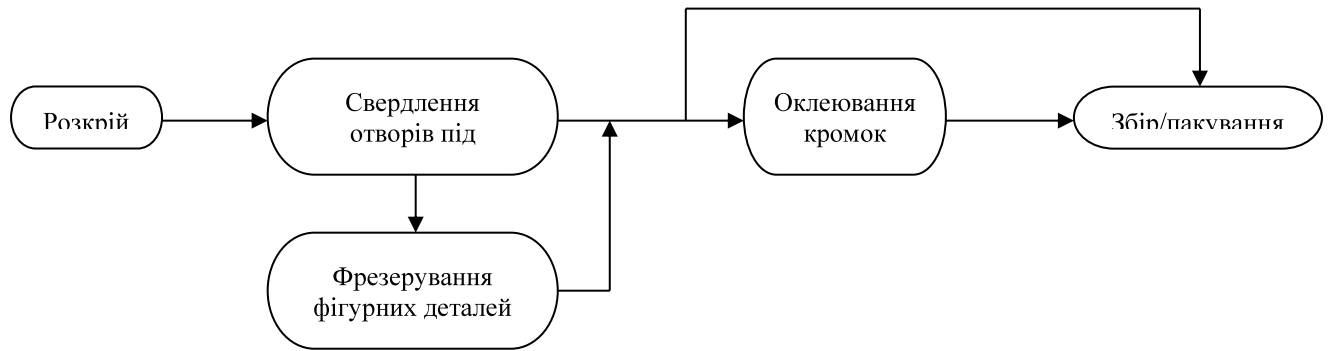


Рисунок 2.2 – Технологічний процес виготовлення замовлення

Як видача так і прийом матеріально-технічних ресурсів на складі ведеться по накладним, та звітується до бухгалтерії і відділу МТП. Прийом матеріалів здійснюється на підставі отримання від постачальника «товарно-транспортної накладної». Отримавши матеріали комірник фіксує факт надходження в «ордері прийому». Видача ведеться по пред'явленню «забірної картки». Комірник перевіряє наявність затвердженої «вимоги накладної» на дане замовлення та видає матеріали оформлюючи «акт списання». «Ордер прийому» та «акт списання» оформлюються комірником в трьох екземплярах. Один екземпляр залишається на складі для реєстрації в «картці складського обліку», другий передається до відділу МТП а третій до бухгалтерії.

Готові замовлення підлягають розвозці по магазинам-замовникам. Накопиченні за минулий робочий день заявки на перевезення оброблюються диспетчером та розроблюються маршрути розвозу для вантажівок. Водієві передається гарантійний талон, документи на отримання товару та на прийом залишку оплати. Водій доставляє замовлення, забирає решту оплати і підписує документи у клієнта. Після доставки замовлення водій здає документи і гроші бухгалтеру, який контролює правильність розрахунків і оформлення.

Враховуючи визначений процес функціонування підприємства можна зобразити наступну функціональну схему (рисунок 2.3):

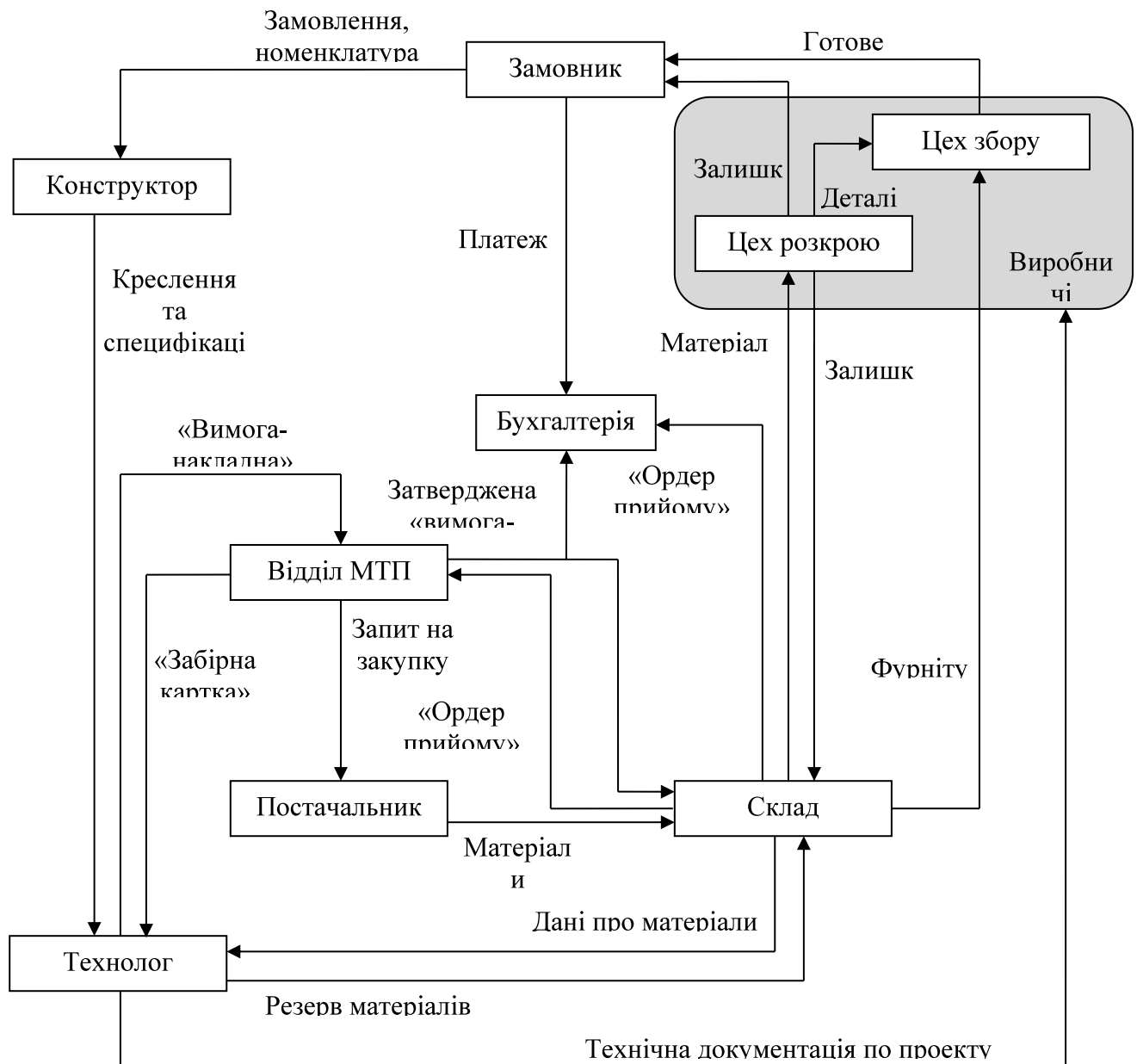


Рисунок 2.3 – Функціональна схема підприємства

2.3 Виявлення проблем

Комп'ютеризація та автоматизація на підприємстві процесів управління матеріальним потоком має фрагментарний характер. Через це зразу виникають такі проблеми в обробці інформації:

- Низька швидкість передачі інформації;
- Зберігання інформації одночасно в декількох місцях;
- Повторення операцій обробки інформації;
- Необхідність надмірного спілкування між співробітниками;

Дані фактори суттєво впливають на актуальність, вірогідність та на можливість втрати і викривлення інформації. Значно знижується оперативність управлінської діяльності.

Окрім проблем викликаних частковим характером автоматизації на підприємстві можна виділити ще й такі проблемні місця:

- Зберігання однієї і тієї ж інформації в різних місцях у різному вигляді. В бухгалтерському обліку та в більшій частині облікової документації, матеріали представляються в метрах квадратних (m^2), що не дає однозначного представлення про дані. Технологу та іншому конструкторському і виробничому персоналу необхідна інформація реальних параметрів – лінійних розмірів листів;
- Можна відмітити ситуативний характер ведення та побудови графіку виробництва замовлень;
- Відсутній налагоджений механізм побудови маршрутів розвозу готових замовлень;
- Також відмічається визначені вище проблеми організації закупок матеріалів на підприємство та проблеми роботи з залишками матеріалів по закінченню виробництва.

Визначені проблеми та недоліки в інформаційній системі управління матеріальними потоками підприємства напряду впливають на ефективність його роботи та раціональне використання ресурсів.

2.4 Постановка задачі

В ході аналізу предметної галузі було визначено ряд проблем в організації управління матеріальними потоками на підприємстві. Метою розробки інформаційної системи – є підвищення ефективності даного процесу. Розроблена система орієнтована на вирішення задачі побудови та управління виконавчими графіками, що дасть змогу забезпечити актуальність інформації та високу оперативність в обробці/передачі даних в прийнятті управлінських рішень.

Для досягнення поставленої задач розроблена система повинна забезпечувати наступні функції:

- Автоматизація побудови черги виконання замовлень на розкрій;
- Управління отриманим графіком виконання замовлень на розкрій;
- Автоматизація побудови розкладів маршрутів перевезень;
- Управління розкладом маршрутів.

Програмне забезпечення повинне реалізовувати графічний інтуїтивно зрозумілий інтерфейс користувача. Апаратне забезпечення повинне максимально повно охоплювати існуючий парк комп'ютерної техніки на підприємстві.

2.5 Розробка концепції МАС управління матеріальними потоками підприємства на етапі виробництва та збуту

Запропонований варіант вирішення проблеми управління ресурсами на підприємстві на етапі виробництва та збуту являє собою автоматизовану підсистему побудови та управління виконавчими графіками.

Мультиагентний підхід вирішення задачі побудови виконавчих графіків передбачає розробку агентів-замовників та агентів-виконавців, які ставляться у відповідність реальним об'єктам предметної галузі представляючи їх інтереси. Черода переговорів даних типів агентів буде шляхом побудови виконавчого графіку. Агент-замовник починає планування шляхом пошуку необхідних йому ресурсів в моделі поточної ситуації, а саме, який агент-виконавець який розклад виконує. Модель ситуації представлена взаємозв'язками між агентами. Агенти кожного замовлення та ресурсу будують свої власні, але пов'язані в загальну мережу розклади.

Для побудови графіку виробництва необхідно розробити агентів замовлення для представництва в черзі виконання інтересів кожного замовлення та агентів виробничих ділянок для формування в інтересах своєї ділянки графіку виконання. Також необхідне АРМ майстра цеху, для корегування отриманого графіку та фіксації факту виконання замовлень.

За даною концепцією вантажі на етапах закупки та збуту об'єднуються в один маршрут. Даний підхід полягає в тому, що завдяки застосуванню мультиагентної технології стає можливе динамічне перепланування маршрутів вантажівок підприємства в «реальному часі». Суть ідеї в тому, що вантажівки які розвозять готові замовлення будуть «підхоплювати» замовлення на доставку матеріалів з місцевих оптових баз на підприємство. Такий підхід дозволить відмовитись від великих за об'ємом планових поставок, які можуть не виправдатись, і при цьому не створити великих зайвих транспортних розходів на малі закупки під замовлення.

Візуально концепцію можна представити наступним чином (рисунок 2.4):

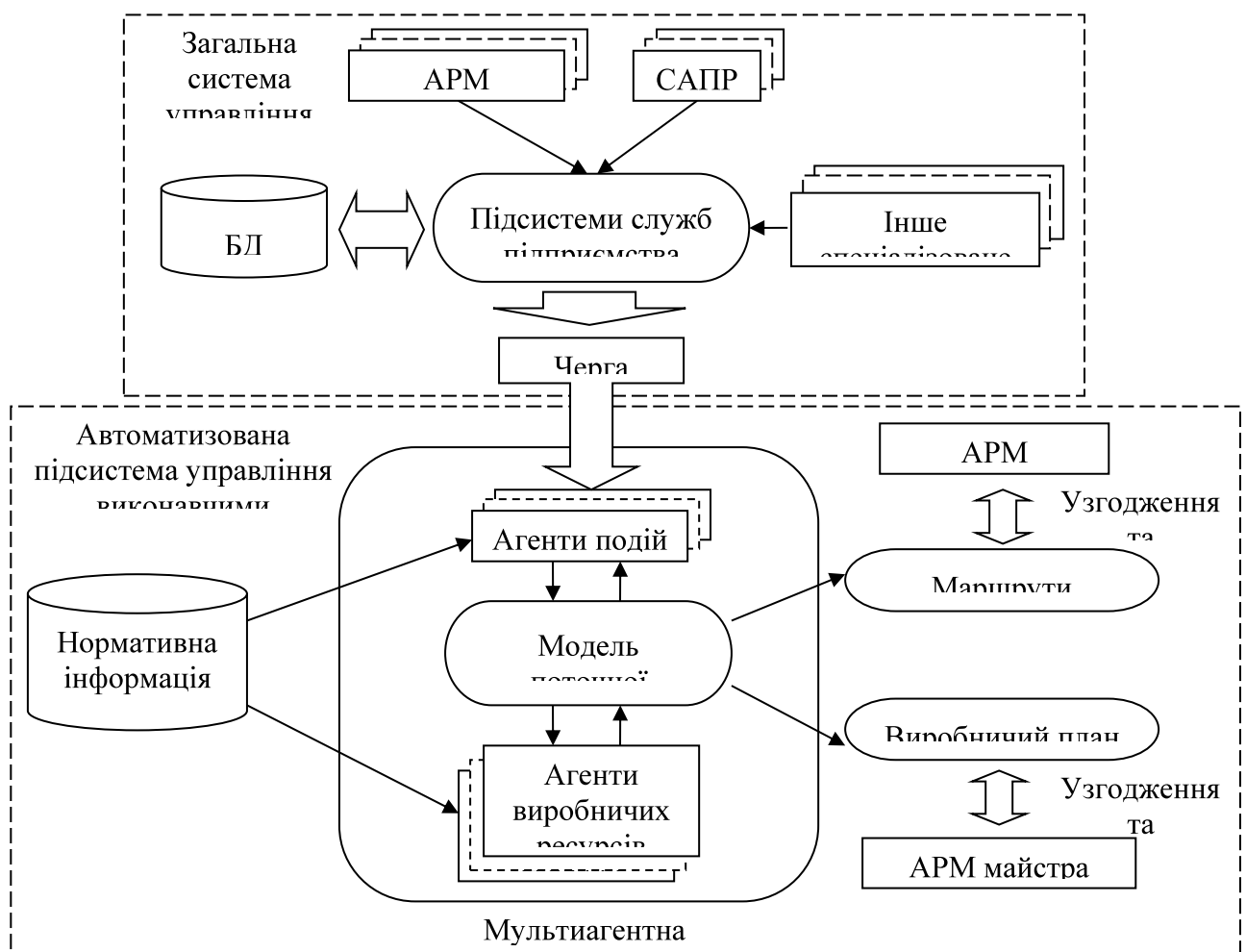


Рисунок 2.4 – Концепція МАС управління матеріальними потоками

2.6 Загальносистемні рішення

2.6.1 Розробка динаміки системи

Поведінку автоматизованої підсистеми управління виконавчими графіками можна зобразити у вигляді діаграми станів наступним чином (рисунок 2.5):

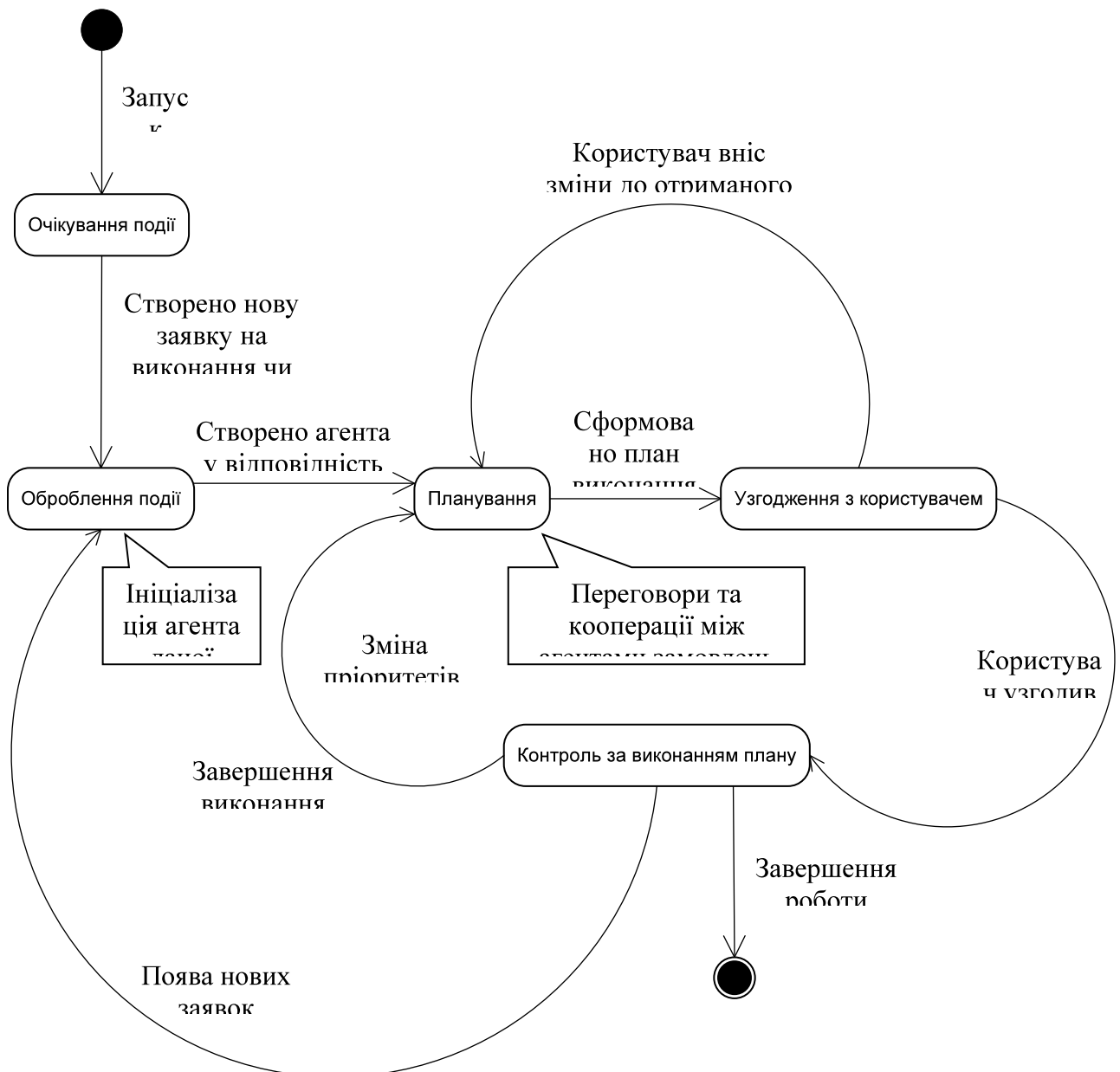


Рисунок 2.5 – Діаграма станів автоматизованої підсистеми управління виконавчими графіками

Після запуску система знаходиться в режимі очікування події – надходження в систему нового наряду на розкрій чи замовлення на перевезення. Потім, при надходженні в систему заявки на виконання проходить ініціалізація агента, який буде поставлений їй у відповідність. Агент починає планування шляхом пошуку необхідних йому ресурсів в моделі поточної ситуації, а саме, який ресурс який розклад виконує. Після чого відбувається узгодження з користувачем. Зміни користувача в отриманому плані переводять систему знову в стан планування. Після узгодження користувачем плану система переходить в стан контролю за його виконанням. Створений план запускається на виконання, в ході якого система стежить за його виконанням та ініціює перепланування в разі виявлення розбіжностей між планом і фактом. Зміна пріоритетів в ході виконання переведе систему на етап планування. Завершення виконання деякого замовлення також може перевести систему в режим перепланування, так як вихід з черги даного замовлення може дати можливість іншому замовленню поліпшити свій стан. При появі в системі нової заявки на виконання викличе ініціалізацію для неї агента та даний цикл буде повторено.

2.6.2 Розробка активності та логіки поведінки агентів системи

Отже, за концепцією в системі діють агенти наступних чотирьох типів:

- Агент замовлення: планує своє виконання – пошук підходящих виробничих дільниць для виконання наряду на розкрій;
- Агент виробничої дільниці: формує чергу свого виконання та управляє своїм розкладом роботи;
- Агент вантажу – пошук підходящої вантажівки для транспортування до місця призначення;
- Агент вантажівки: формує свій маршрут розвезень готових замовлень та доставок матеріалів від постачальників.

Поведінку агента визначають його ментальні категорії (BDI): переконання (уявлення про поточний стан системи), бажання (набір бажаних цілей які переслідує агент – стани системи в які агент намагається її перевести) і наміри

(всі бажання одночасно можуть бути несумісними чи недосяжними, тому необхідно вибирати самі значущі – наміри). Ментальні властивості агентів представлені в таблиці 2.1.

Таблиця 2.1 – Ментальні категорії агентів системи

Назва	Агент замовлення	Агент виробничої ділянки	Агент вантажу	Агент вантажівки
Переконання	Час початку свого виконання	Очікуваний прибуток на даний момент часу	Вартість свого перевезення. Час доставки	Очікуваний прибуток на даний момент часу. Витрати бензину. Завантаженість машини
Бажання	Бути виконаним найшвидше	Отримати максимальний прибуток	Бути перевезеним найдешевшим способом.	Отримати максимальний прибуток. Максимально використати вантажомісткість
Намір	Бути виконаним найшвидше	Отримати максимальний прибуток	Бути перевезеним найдешевшим способом.	Отримати максимальний прибуток

Розглянемо життєвий цикл кожного агента.

Агент замовлення. Збереження технологом в системі нового наряду на розкрій ініціює створення йому у відповідність агента замовлення. Спочатку агент визначає місцезнаходження матеріалів:

1 Якщо всі матеріали забираються зі складу, то агент починає шукати агента виробничої ділянки з найсприятливішим графіком – пошук ділянки з найменшим часом можливого початку виконання. Вставши в чергу виконання ділянки агент переходить в стан очікування свого виконання та моніторингу системи на зміни – на іншій ділянці можуть бути виконані чи відмінені замовлення, що зменшить на цій ділянці можливий час початку виконання. При таких умовах агент перейде на нову ділянку. Під час очікування агент також може бути «виштовхнутим» з черги виконання іншим агентом замовлення – цей факт переводить агента знову на етап пошуку найсприятливішої ділянки. «Виштовхування» відбувається в тому разі якщо агент не використовує залишки іншого замовлення і згоден перейти на іншу ділянку і прийняття нового агента замовлення для агента виробничої ділянки є більш вигідним. Якщо агент

використовує залишки іншого замовлення – він відмовляється звільнити своє місце в черзі.

2 Якщо на початку агентом було визначено зв'язок з іншим замовленням – використання залишків іншого замовлення яке ще перебуває в черзі на виконання, то агент намагається «виштовхнути» з черги ділянки наступне замовлення після пов'язаного. Якщо «виштовхнути» дане замовлення не вдалося – агент починає шукати інші замовлення початок виконання яких максимально наблизений до кінця виконання пов'язаного замовлення і намагається «виштовхнути» їх.

3 Якщо на початку агентом було визначено зв'язок з декількома замовленнями, то агент визначає час закінчення останнього з пов'язаних замовлень і діє як в пункті 2. Після чого визначає попередні пов'язані замовлення і «просить» їх змінити час початку виконання максимально наближено до часу початку виконання останнього пов'язаного з ним замовлення. Агенти цих замовлень погоджуються змінити час свого виконання шляхом «виштовхування» інших замовлень якщо не використовують залишки інших замовлень.

Фіксація майстром цеху виконання замовлення закінчує життєвий цикл агента замовлення.

Даний життєвий цикл агента замовлення у вигляді діаграми станів представлено на рисунку 2.6

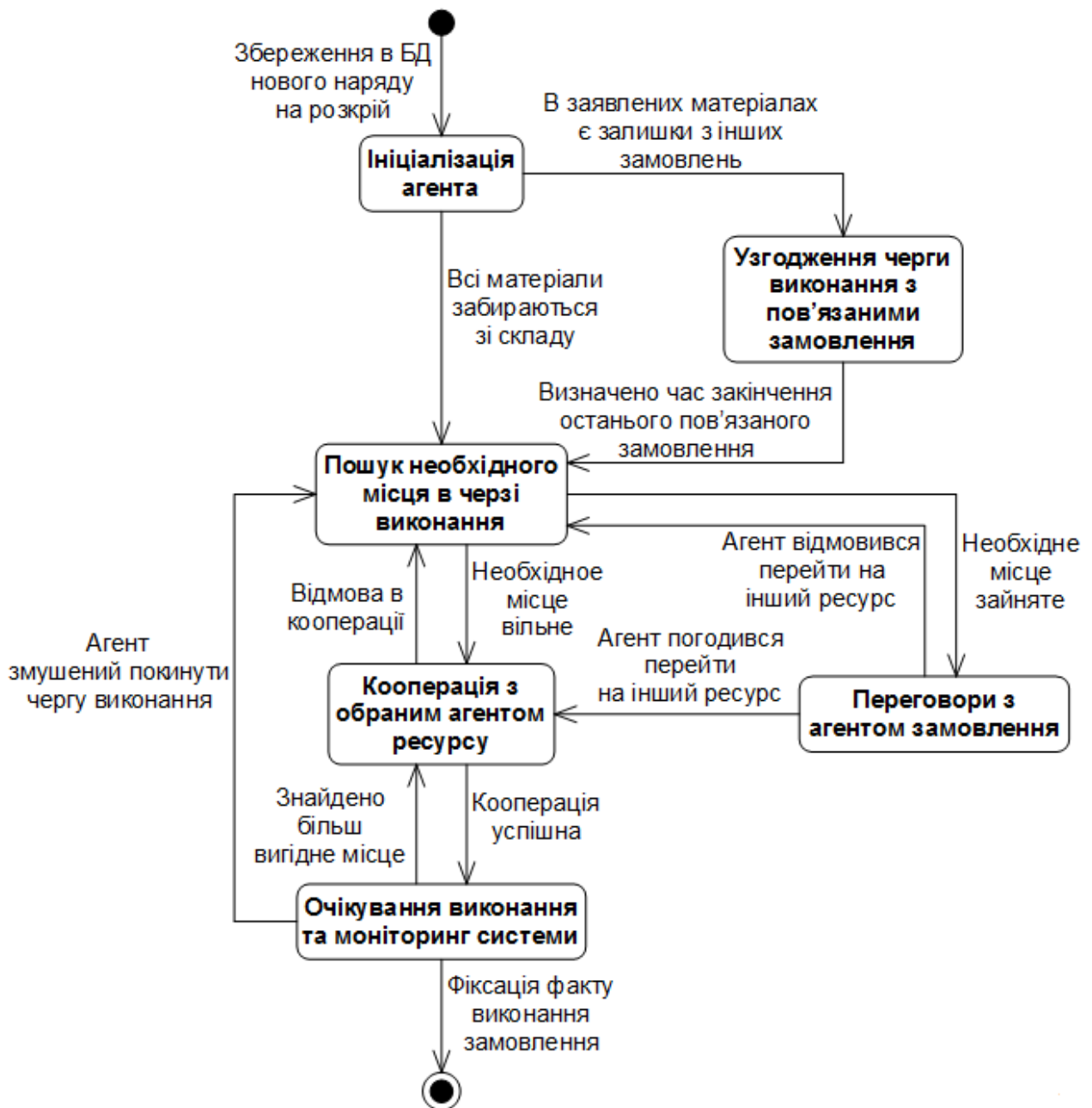


Рисунок 2.6 – Діаграма станів агента замовлення

Побудова графіку виконань нарядів на розкрій за даним алгоритмом діяльності агентів замовлень віршує проблему накопичення залишків виробництва та скорішого їх використання, а також значно розвантажує транспортну систему в середині підприємства. Так як пов'язані залишками замовлення будуть кроїтися приблизно одне за одним.

Агент виробничої дільниці. Створення агента виробничої дільниці ініціюється при запуску системи. Після своєї ініціалізації агент переходить в стан очікування заявок на виконання від агентів замовлень. Якщо заявка на виконання приходить на вільний часовий інтервал – агент виробничої дільниці оцінює

можливість прийняття даного замовлення і включає його в чергу свого виконання. Але, якщо заявка приходить на часовий інтервал, який зайнятий іншим замовленням – агент приймає рішення: якщо нове замовлення більш вигідне – прийняти його в чергу виконання, а старе замовлення піде на інший ресурс, а якщо навпаки – відмовити новому замовленню.

Діаграма станів агента виробничої дільниці представлена на рисунку 2.7

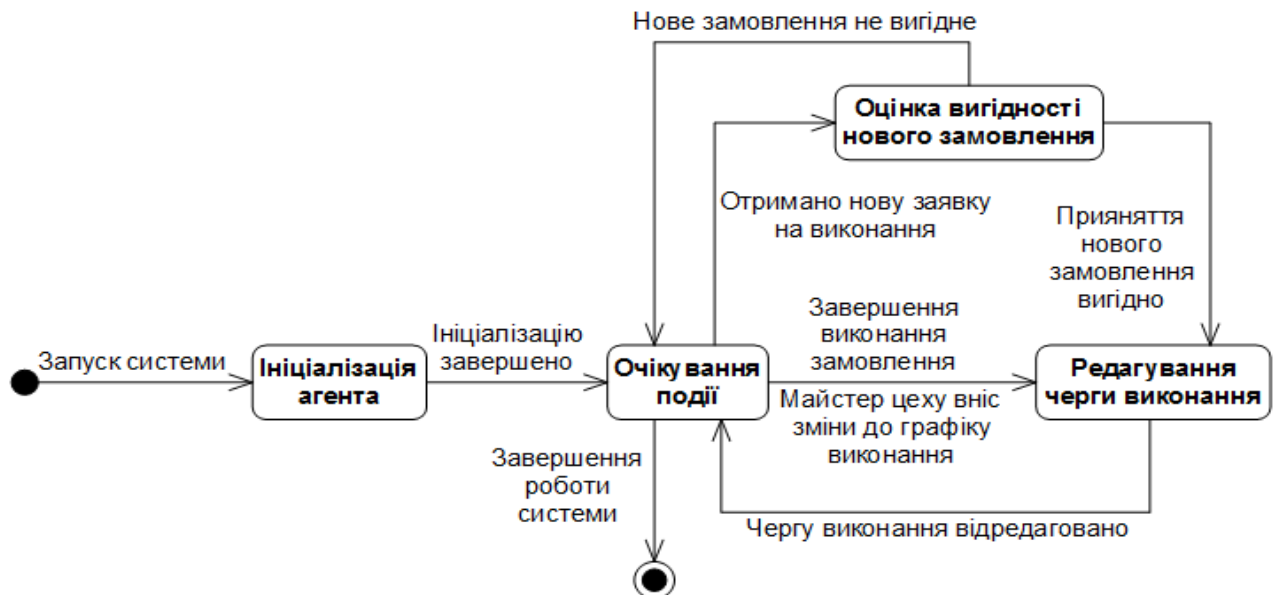


Рисунок 2.7 – Діаграма станів агента виробничої дільниці

В ході переговорів для планування маршрутів перевозок агенти повинні дотримуватись наступних правил:

- 1 Агенти вантажів повинні віддавати перевагу агентам вантажівок для яких вже заплановано операції завантаження/розвантаження в пунктах близьких до пунктів завантаження/розвантаження даного замовлення. Такий варіант може бути більш дорогим, але все одно більш «виграшним» ніж варіант планування на нову вантажівку.
- 2 При виборі набору вантажів агенти вантажівок повинні враховувати, що в разі непередбачених подій вантажівці прийдеться вертатись на базу не закінчивши маршруту. Тому в якості першого пункту розвантаження агент повинен вибирати найбільш віддалену від бази точку, щоб після неї вантажівка верталась на базу природнім шляхом. Такий підхід також забезпечить «підхват» замовлень на перевезення матеріалів з оптових баз. Оптові бази матеріалів місцевого характеру

територіально розташовуються відносно близько від бази підприємства, і тому коли вантажівка буде вертатись на базу з віддаленого пункту, вона вже буде достатньо розвантажена для «підхвату» даного замовлення.

Агент вантажу. Збереження диспетчером в системі інформації про новий вантаж ініціює створення нового агента вантажу. Після ініціалізації агент починає пошук най вигіднішої (найближчої) точки маршруту вантажівок до свого місця доставки. Якщо маршрут вантажівки на даний момент час «пустий», в ньому все одно знаходиться одна точка маршруту – склад самого підприємства. Якщо вибір агент випадає на таку вантажівку, то агент відкидає даний варіант і планує своє перевезення на менш вигідну для нього вантажівку (правило 1). До варіанту планування свого перевезення на «пусту» вантажівку агент повертається якщо він не зміг домовитись з іншими вантажівками.

Діаграму станів агенту вантажу представлено на рисунку 2.8.

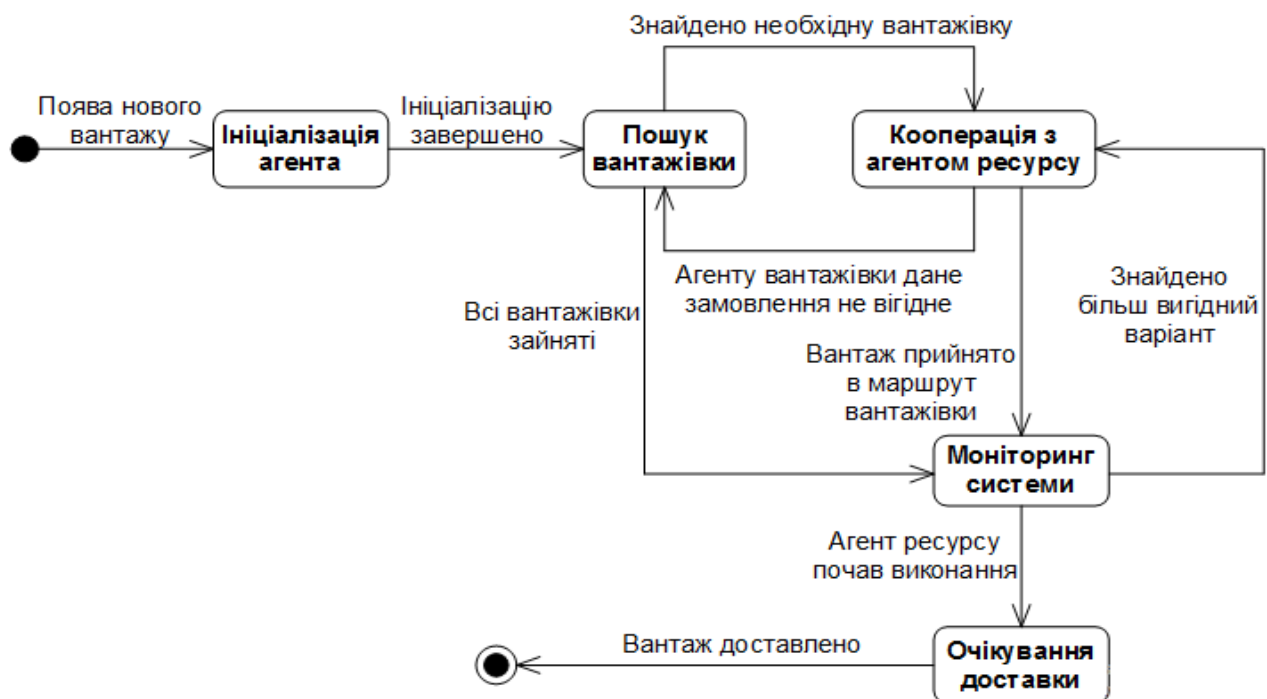


Рисунок 2.8 – Діаграма станів агенту вантажу

Агент вантажівки. Створення агента вантажівки ініціюється при запуску системи. Після своєї ініціалізації агент переходить в стан очікування заявок на перевезення від агентів вантажу. Вантаж вважається не вигідним якщо витрати на зміну маршруту більші ніж прибуток який буде отримано за доставку вантажу. З доданням до маршруту нового замовлення маршрут коректується:

першим пунктом стає найвіддаленіша точка, наступна – найближча до неї, і так далі (правило 2). Коли завантаженість машини набуває достатнього рівня – агент вантажівки переходить в стан виконання і приймає замовлення на перевезення тільки на доставку матеріалів з оптових баз.

Діаграма станів агента вантажівки представлена на рисунку 2.9.

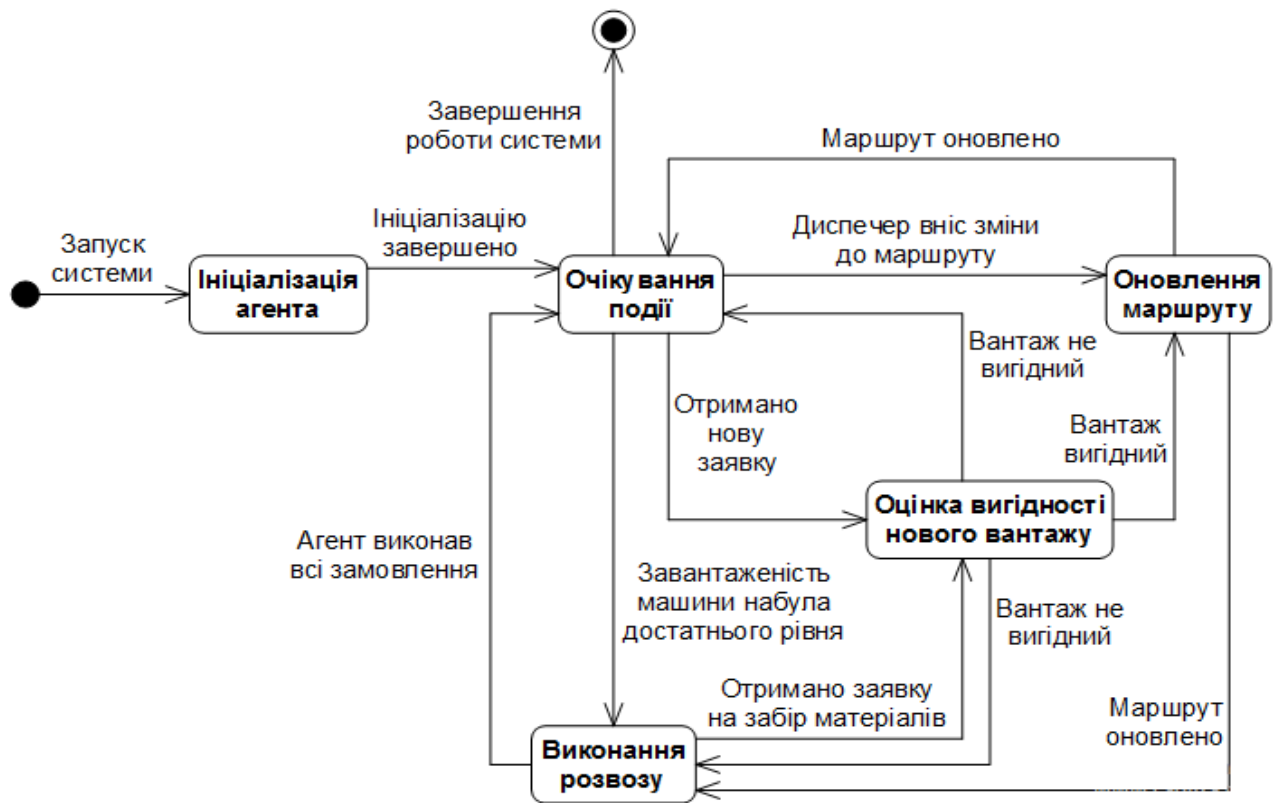


Рисунок 2.9 – Діаграма станів агента вантажівки

2.6.3 Розробка рішень з взаємодії агентів системи

Взаємодію «агента-замовника», «агента-виконавця» та користувача системи представлено у вигляді діаграми послідовності на рисунку 2.10.

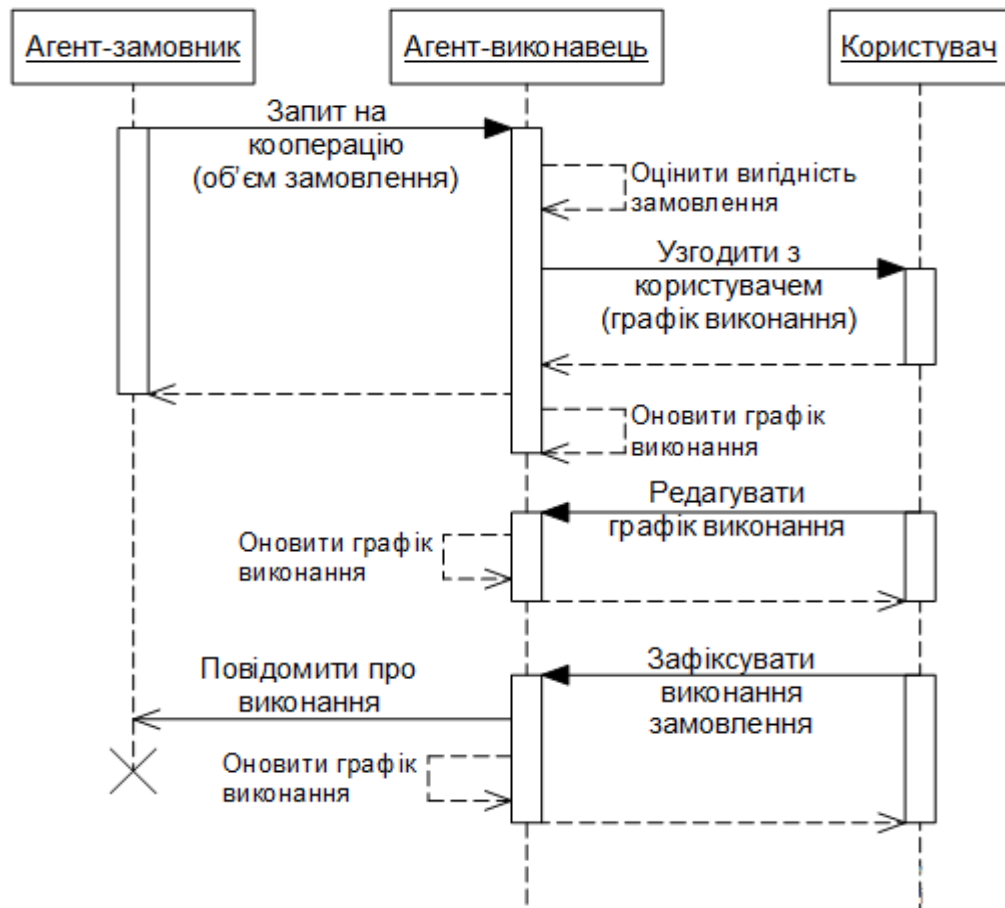


Рисунок 2.10 – Взаємодія агентів та користувача

Знайшовши необхідний ресурс, агент-замовник звертається з запитом на кооперацію до відповідного агента-виконавця. Агент-виконавець оцінює вигідність прийняття на виконання даного замовлення, узгоджує рішення з користувачем та надсилає підтвердження кооперації агенту-замовнику. Також в ході виконання графіку користувач може бути змушений внести і нього зміни. Користувач надсилає нові параметри, агент-виконавець згідно них оновлює свій графік та повертає підтвердження. Аналогічно у випадку коли користувач зафіксував завершення виконання певного замовлення – агент-виконавець повідомляє агента-замовника про виконання та повертає оновлений графік виконання.

Окрема необхідно розглянути переговори агентів замовлень у випадку коли одне замовлення «виштовхує» інше та коли одне замовлення простить змінити час виконання інше (рисунок 2.11).

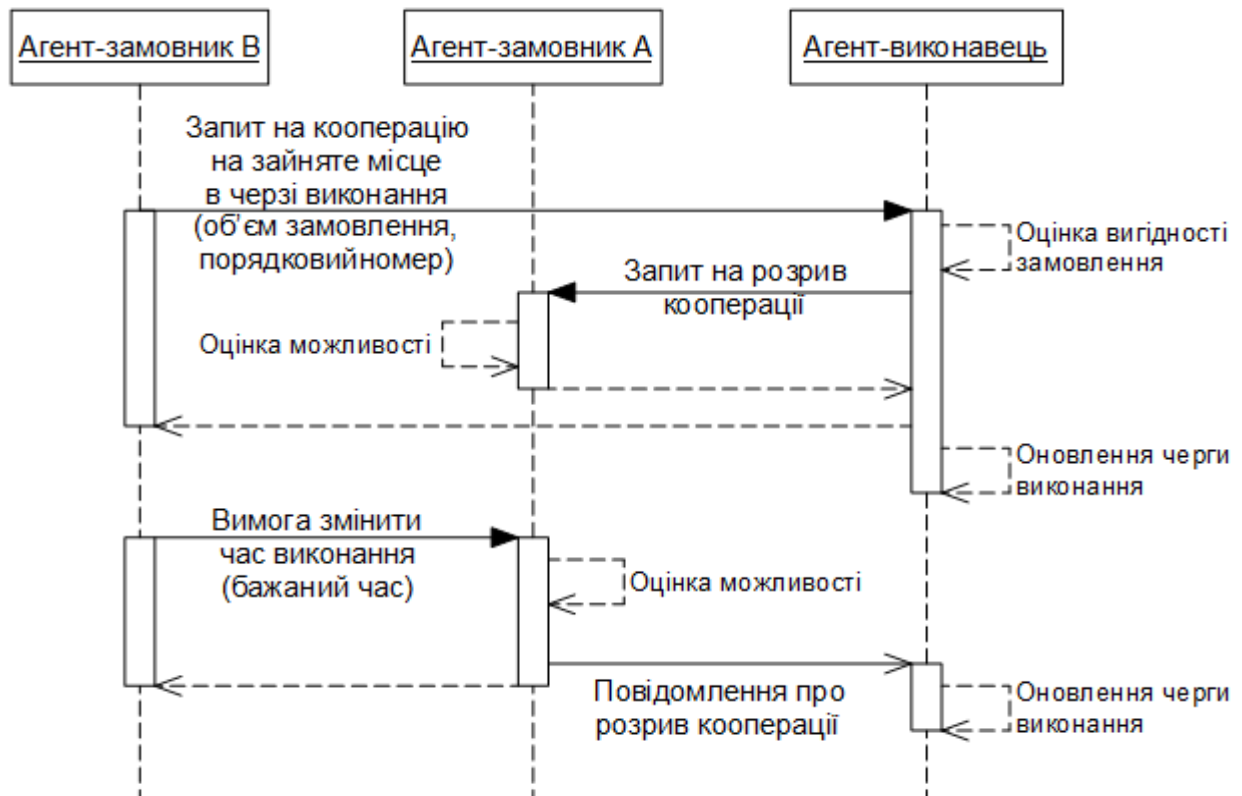


Рисунок 2.11 – Переговори агентів

Якщо агенту замовлення стає необхідним «виштовхнути» деяке інше замовлення, він спочатку звертається до відповідного агента-виконавця з запитом на кооперацію на зайняте місце в черзі виконання надсилаючи об'єм свого замовлення та бажаний порядковий номер в черзі. Агент-виконавець оцінює вигідність пропозиції, посилає запит на розрив кооперації агенту, якого виштовхують, і якщо той в змозі покинути чергу виконання – посилає підтвердження кооперації новому замовленню.

Також розглядається такий варіант, коли агент замовлення просить змінити час початку виконання іншого агента надсилаючи йому бажаний час виконання. Даний агент оцінює можливість виконання вимоги, надсилає повідомлення агенту-виконавцю про розрив кооперації та відправляє підтвердження першому агенту.

2.7 Рішення з математичного забезпечення системи

Як зазначалося вище, поведінку агента визначають його ментальні категорії. Міра близькості до мети агента отримується шляхом зіставлення

переконань і бажань – поточного та бажаного стану системи. Бажання і наміри зазвичай формуються математично у вигляді задач оптимізації тих чи інших функціоналів якості або досягнення якихось заданих множин.

Побудуємо математичну модель побудови графіка виконання замовлень на розкрій:

$Z = \{z_i\}_{i=1}^n$ - множина замовлень на дільниці.

Кожне замовлення характеризується набором наступних параметрів:

$$\forall z \in Z : (t_{call}, t_{max}, V, t_{end})$$

t_{call} – час надходження замовлення;

t_{max} – максимальний час виконання;

V – об'єм замовлення (загальна довжина (l) та кількість різів (k))

t_{end} – приблизний час виконання

($t_{end} = k \cdot t_{work}$, t_{work} – норма часу виконання робіт розкрою)

$A = \{a_j\}_{j=1}^m$ - множина агентів виробничих дільниць.

Агент намагається здобути максимальний прибуток своїй виробничій дільниці. Цільова функція j -го агента виробничої дільниці має наступний вигляд:

$$f_j(Z) = \sum_{i=1}^n S(z_i) \rightarrow \max$$

$S(z_i)$ - сума за виконання i -го замовлення. Оплата за виконання нарізних робіт залежить від загальної довжини різів в замовленні (l) відповідно до встановленого тарифу підприємства (q) (зазвичай в Миколаєві – 1,5-2 гривні за погонний метр). Але треба враховувати що робота виробничої дільниці обмежена часом робочого дня, тому цільова функція набуває наступного вигляду:

$$\begin{cases} f_j(Z) = \sum_{i=1}^n z_i(l) \cdot q \rightarrow \max \\ \sum_{i=1}^n z_i(t_{work}) \leq 8 \end{cases}$$

Тепер розробимо математичну модель побудови маршрутів перевезень готових замовлень на поставок з оптових баз:

$Z = \{z_i\}_{i=1}^n$ - множина замовлень на перевезення.

Кожне замовлення характеризується набором наступних параметрів:

$$\forall z \in Z : (p(x; y), m)$$

p – координати пункту призначення;

m – вага вантажу.

$A = \{a_j\}_{j=1}^m$ - множина агентів вантажівок.

Кожний агент має набір властивостей, характерний його фізичній реалізації:

$$\forall a \in A : (M, V_{сер}, R)$$

M – загальна вантажомісткість машини;

$V_{сер}$ – середня швидкість руху вантажівки;

R – розхід палива.

Агент вантажу намагається бути перевезеним найдешевшим способом. Тобто відстань між пунктом призначення і однією з точок маршруту вантажівки повинна бути мінімальною. Відстань розраховується за формулою:

$$d = \sqrt{(x_a - x_z)^2 + (y_a - y_z)^2} \rightarrow \min$$

За виконання кожного замовлення агент вантажівки отримує винагороду. Сума виплати за перевезення залежить від відстані між підприємством та пунктом призначення в відповідності до встановленого підприємством тарифу (q).

Функція доходу від виконання i -го замовлення:

$$S(z_i) = \sqrt{(x_0 - x_{zi})^2 + (y_0 - y_{zi})^2} \cdot q$$

$(x_0; y_0)$ – координати складу підприємства.

Виконання замовлень на перевезення призводять до певних витрат ресурсів – розходу бензину. Функція витрат вираховується як добуток довжини маршруту на розхід та ціну (c) палива. Перед розрахунком множина замовлень повинна бути упорядкована починаючи з самого віддаленого і закінчуючи найближчим пунктом призначення. Функція витрат:

$$L(Z) = \left(\sum_{i=n}^{n-1} \sqrt{(x_{zi} - x_{zi+1})^2 + (y_z - y_{zi+1})^2} + \sqrt{(x_0 - x_{z1})^2 + (y_0 - y_{z1})^2} \right) \cdot R \cdot c$$

Цільова функція агента вантажівки направлена на максимізацію прибутку (різниця між доходом та витратами) з обмеженням в часі виконання розвозу, так як робочий день складає 8 годин, та з обмеження вантажомісткості машини:

$$F = \sum_{i=1}^n S(z_i) - L(Z) \rightarrow \max$$

$$\left\{ \begin{array}{l} \frac{\left(\sum_{i=n}^{n-1} \sqrt{(x_{zi} - x_{zi+1})^2 + (y_z - y_{zi+1})^2} + \sqrt{(x_0 - x_{z1})^2 + (y_0 - y_{z1})^2} \right)}{V_{cep}} \leq 8 \\ \sum_{i=1}^n z_{im} \leq M \end{array} \right.$$

2.8 Рішення з інформаційного забезпечення системи

В рішеннях з інформаційного забезпечення необхідно визначити повний об'єм інформації який забезпечить функціонал агентів.

Агент замовлення представляє в системі інтереси свого наряду на розкрій. Агент повинен мати свій ідентифікатор, знати про свій порядковий номер в черзі та зберігати дату надходження замовлення та час початку свого виконання. Також агент замовлення повинен надати агенту ресурсу інформацію про загальну довжину різів в замовлені та їх кількість, для оцінки своєї вигідності. Також агент повинен зберігати інформацію про інструкцію щодо забору матеріалів – забирати всі матеріали зі складу, чи деякі листи повинні знаходитись в виробничих підрозділах. Окрім інструкції необхідно вказувати примітку щодо здачі залишків – за для розвантаження транспортної системи підприємства доречніше залишки, які будуть кроїтись в наступних замовлення, залишити на виробничій дільниці.

Агент вантажу також має свій ідентифікатор та зберігає інформацію про дату надходження заявки на перевезення та порядковий номер в маршруті розвозу. Крім цього агент оперує інформацією про вантаж – вага, пункт завантаження та пункт розвантаження.

Кожен агент-виконавець також має свій ідентифікатор та тариф за виконання заявок. Агент виробничої дільниці також повинен мати дані про норми часу на виконання робіт по розкрою. Агент вантажівки зберігає таку інформацію: вантажомісткість машини, розхід палива, та середня швидкість.

Розроблена модель даних для забезпечення функціоналу агентів представлена у вигляді діаграми класів на рисунку 2.12.

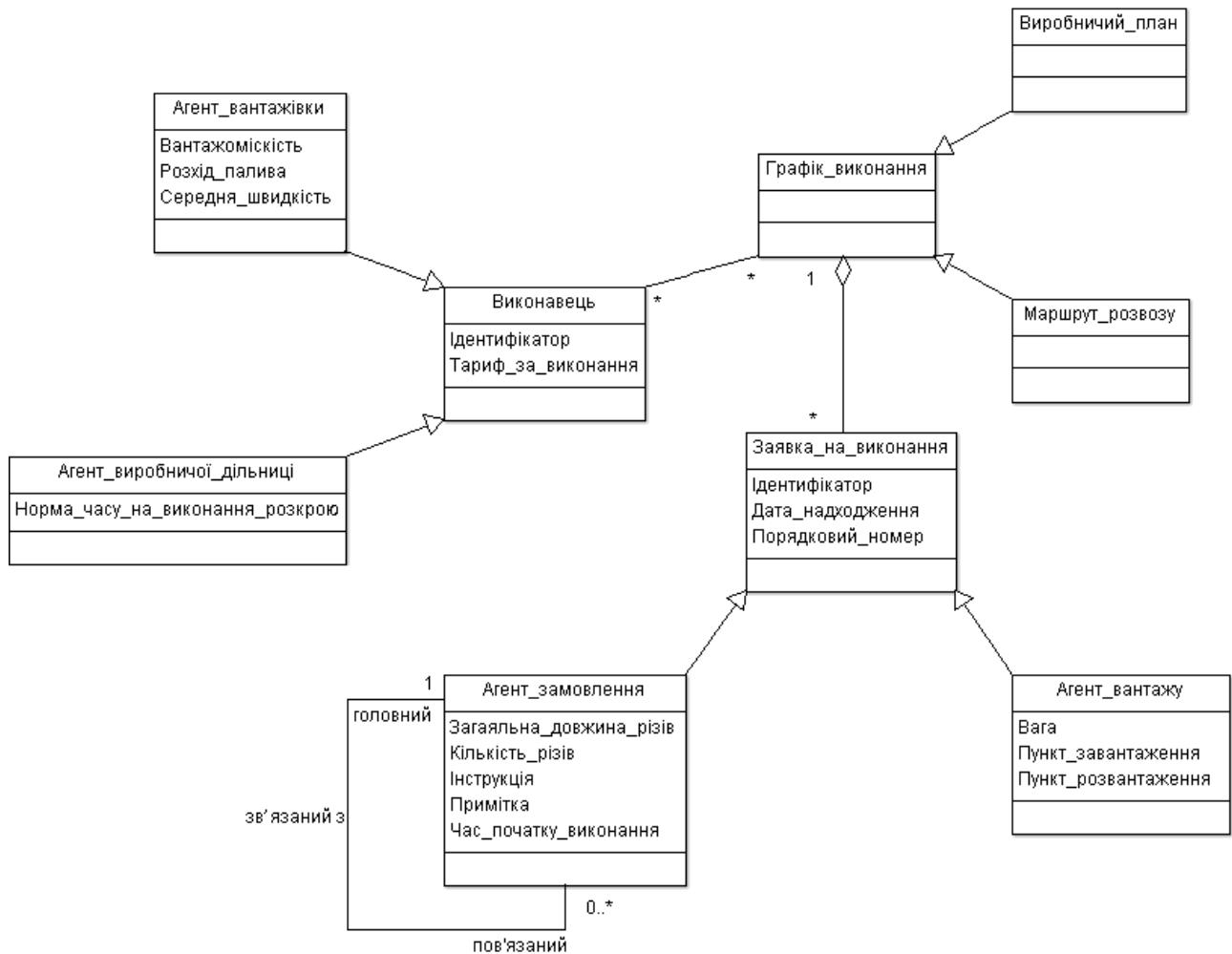


Рисунок 2.12 – Діаграма класів агентів

2.9 Рішення з програмного забезпечення

2.9.1 Вибір та опис мультиагентної платформи

Середовищем розробки та існування МАС є мультиагентні платформи. Було розроблено безліч програмних реалізацій мультиагентних платформ, кожна з яких має свої особливості, переваги і недоліки. Багато з них успішно існують у вигляді комерційних проектів або проектів, які позиціонують як проекти з відкритим вихідним кодом.

У 90-х роках виникла необхідність створення єдиних стандартів на розробку агентних систем. У цей період були засновані дві організації MASIF

(Mobile Agent System Interoperability Facility) і FIPA (Foundation of Physical Intelligent Agents). В результаті їх роботи з'явилися стандарт MASIF і стандарт FIPA, які дають рекомендації по створенню систем мобільних агентів і систем, інтелектуальних агентів.

Однією з найбільш популярних агентних платформ в даний час є платформа JADE (Java Agent DEvelopment Framework). Проект JADE розробляється компанією Telecom Italia Лабораторії з 2000 р. Слід відмітити що дана платформа знаходиться в вільному доступі та більш за інші орієнтована на вирішення задач планування та логістики.

Агентна платформа JADE є типовим Middleware, тобто програмним забезпеченням середнього рівня (рисунок 2.13), яке являє собою набір засобів для створення і управління системою з безліччю агентів.

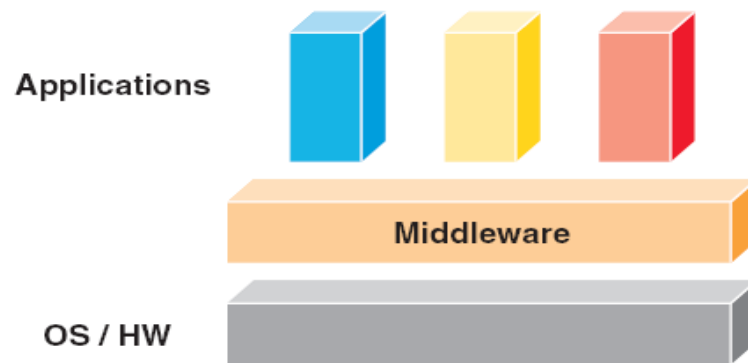


Рисунок 2.13 – Місце Middleware в структурі ПЗ

Платформа розробки мультиагентних систем JADE включає в себе динамічне середовище, де можуть «жити» JADE агенти; бібліотеку класів, яку програмісти можуть використовувати для розробки власних агентів; набір графічних інструментів, що дозволяють керувати активністю запущених агентів.

JADE надає програмісту - розробнику агентних систем наступний набір засобів:

- FIPA-compliant Agent Platform - агентную платформу, що базується на стандарті FIPA і включає обов'язкові типи системних агентів, які автоматично активується при запуску платформи.
- Distributed Agent Platform - розподілену агентную платформу, яка може використовувати декілька комп'ютерів (вузлів), причому на кожному вузлі запускається тільки одна Java Virtual Machine. Агенти виконуються як Java-поток. Для доставки повідомлень між агентами, залежно від їх

місцезнаходження, використовується відповідний транспортний механізм - Multiple Domains support - безліч заснованих на FIPA-специфікаціях спеціалізованих агентів, які можуть об'єднуватися у федерацію, реалізуючи таким чином мультидоменну агентну середу.

- Multithreaded execution environment with two-level scheduling. Кожен JADE-агент має власний потік управління, але він також здатний працювати в многопоточковому режимі. Java Virtual Machine проводить планування завдань, виконуваних агентами або одним з них.
- Object-oriented programming environment. Більшість концепцій, властивих FIPA-специфікації, представляються Java-класів, що формують інтерфейс користувача.
- Library of interaction protocols. Використання стандартних інтерактивних протоколів fipa-request і fipa-contract-net. Для того щоб створити агента, який міг би діяти згідно з таким протоколом, розробникам прикладних програм потрібно тільки імплементувати специфічні доменні дії, в той час як вся незалежна від прикладної програми протокольна логіка буде здійснюватися системою JADE.
- Administration GUI. Прості операції управління платформою можуть виконуватися через графічний інтерфейс, який відображає активних агентів і контейнери агентів. Використовуючи GUI, адміністратори платформи можуть створювати, знищувати, розбивати і відновлювати дії агентів, створювати ієрархії доменів та мультиагентні федерації.

Платформа JADE написана на мові програмування Java з використанням Java RMI, Java, CORBA IDL, Java Serialization і Java Reflection API. Вона спрощує розробку мультиагентних систем завдяки використанню FIPA-специфікацій та інструментів (tools), які підтримують фази виправлення помилок (debugging) і розгортання (deployment) системи. Ця агентна платформа може поширюватися серед комп'ютерами з різними операційними системами, і її можна конфігурувати через віддалений GUI-інтерфейс. Процес конфігурування цієї платформи досить гнучкий. Єдиною вимогою такої системи є установка на комп'ютері Java Run Time необхідної версії.

Платформа JADE є розподіленою і являє собою набір контейнерів (рисунок 2.14).

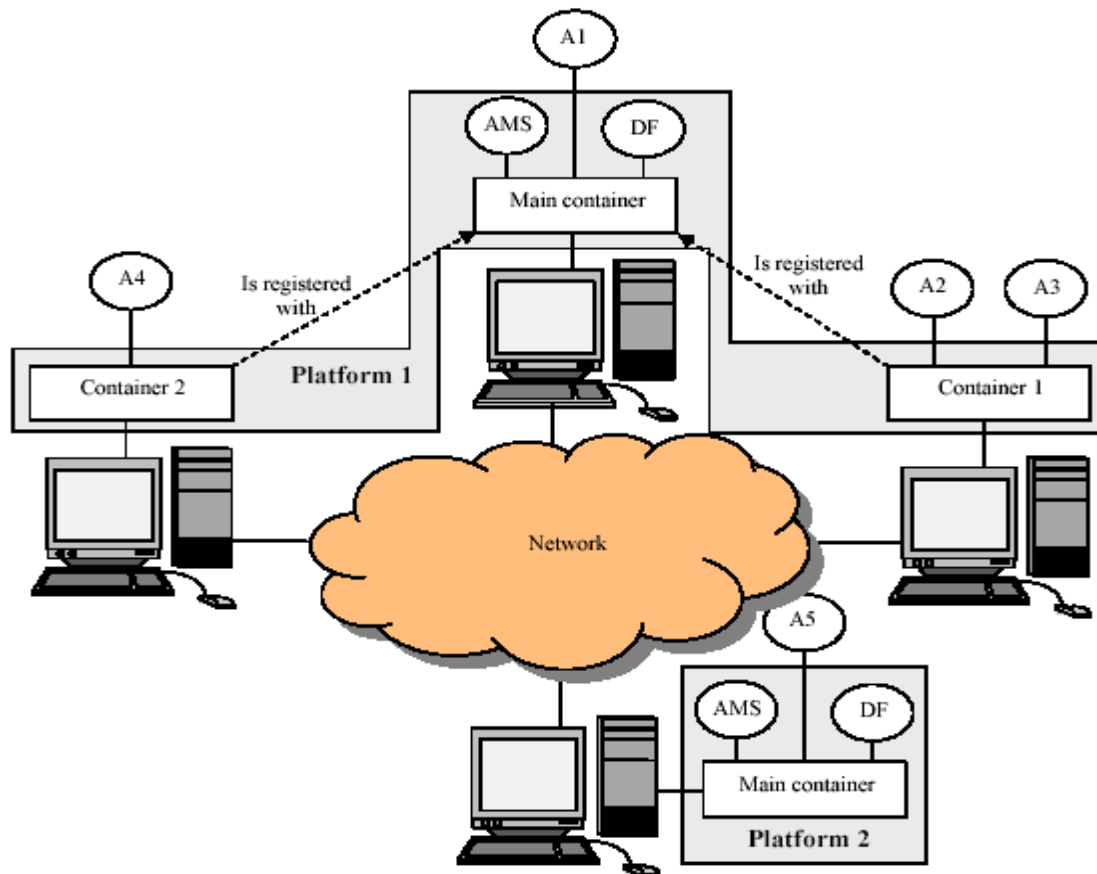


Рисунок 2.14 – Платформа JADE

Контейнером називається динамічна середовище виконання мультиагентних додатків, в якій знаходяться агенти. Кожен контейнер може містити кілька агентів. Набір активних контейнерів називається платформою. Один з контейнерів завжди є головним (Main container), всі інші контейнери зв'язуються з ним і реєструються в момент запуску. Тому першим контейнером при старті платформи повинен бути головний, а всі інші контейнери повинні бути «звичайними» (тобто неголовними) контейнерами і заздалегідь повинні «знати», як знайти головний контейнер, на якому вони будуть реєструватися, тобто повинні мати дані про хост і порт. Інший головний контейнер, запущений де-небудь у мережі, являє собою іншу платформу, на якій можуть зареєструватися нові звичайні контейнери. Рисунок 2.14 ілюструє цю концепцію на основі прикладу, показує дві JADE-платформи, що складаються з трьох і одного контейнера відповідно. JADE-агенти визначаються з унікальними іменами. За умови, що вони знають імена інших агентів, вони можуть

спілкуватися, незалежно від їх фактичного місцезнаходження: загалом контейнері (агенти A2 і A3), в різних контейнерах на одній платформі (агенти A1 і A2), або взагалі на різних платформах (A4 і A5). Користувачеві не обов'язково знати, як працює динамічна середовище JADE, але необхідно запускати її перед початком виконання своїх агентів.

Крім можливості прийому реєстрацій від інших контейнерів, головний контейнер відрізняється від звичайного контейнера тим, що містить два спеціальних агента, які автоматично запускаються одночасно з контейнером:

- AMS (Agent Management System - система управління агентами) забезпечує службу управління агентами, яка дозволяє створювати і видаляти агентів, а також містить у собі простір імен агентів. Ім'я агента є унікальним і має наступний формат: <nickname>@<platform-name>. Знаючи імена один одного, агенти можуть обмінюватися повідомленнями як всередині контейнера і платформи, так і між різними платформами.
- DF (Directory Facilitator - менеджер директорій) являє собою службу «жовтих сторінок» (yellow pages), де агенти можуть публікувати інформацію про наданих ними сервісах. З допомогою DF агент може знаходити агентів, що надають необхідні йому сервіси, і вступати з ними в переговори. Всередині однієї платформи може існувати кілька DF, що надають інформацію про різних групах сервісів або про сервіси різних груп агентів.

Однією з найбільш важливих функціональних можливостей, яку надають JADE агенти, є можливість комунікації між ними. Обмін повідомленнями в JADE є асинхронним і здійснюється з допомогою Message Transport System. Формат повідомлення відповідає стандарту FIPA. При цьому усередині платформи для прискорення роботи повідомлення пересилаються у вигляді Java-об'єктів, а при обміні між платформами (або з іншими MAC) - у вигляді XML-рядка. При відправці повідомлення, воно потрапляє в «поштовий ящик» (черга повідомлень) агента-адресата, про що агент повідомляється. Повідомлення може бути отримано з «поштової скриньки» агента і оброблене будь-яким з його режимів. Для вибору потрібного типу повідомлення з черги програміст може використовувати фільтри.

Повідомлення, якими обмінюються між собою агенти, мають формат мови ACL, визначеного в FIPA як міжнародний стандарт взаємодії агентів. Цей формат включає наступні поля:

- відправник повідомлення;
- список одержувачів;
- комунікативні дії («побажання»), що вказують мету надсилання повідомлення. Типи повідомлень: REQUEST (запит), якщо відправник бажає, щоб одержувач виконав деяку дію, INFORM (інформування), якщо відправник бажає, щоб одержувач був проінформований про деякий факт, QUERY_IF, якщо відправник бажає знати, досягнуто чи ні заданої умови, CFP (повідомлення про пропозицію) пропозиція (PROPOSE), ACCEPT_PROPOSAL (прийняття пропозиції), REJECT_PROPOSAL (відхилення пропозиції) (відхилення пропозиції), якщо відправник і одержувач ведуть переговори, та інші типи комунікативних дій;
- зміст, тобто фактична інформація, що міститься у повідомленні (наприклад, дії, які будуть виконуватися за запитом REQUEST; факт, про який відправник бажає інформувати в повідомленні INFORM тощо);
- мову, на якому представлено зміст (відправник і одержувач повинні мати можливість кодувати/декодувати вираження згідно синтаксису мови);
- онтологія, тобто словник символів, використовуваних в pvscns, та їх значення (відправник і одержувач повинні розуміти сенс символів однаково);
- поля, що використовуються для управління декількома паралельними переговорами, що визначають інтервал часу для отримання відповіді.

Агент Directory Facilitator (DF) - менеджер директорій (згідно специфікації FIPA) представляє службу жовтих сторінок (yellow pages), де агенти можуть публікувати інформацію про наданих ними сервісах. З допомогою DF агент може знаходити агентів, що надають необхідні йому сервіси, і вступати з ними в переговори. Всередині однієї платформи може існувати кілька DF, наприклад, надають інформацію про різних групах сервісів або про сервіси різних груп агентів. Приклад використання DF показаний на рисунку 2.15.

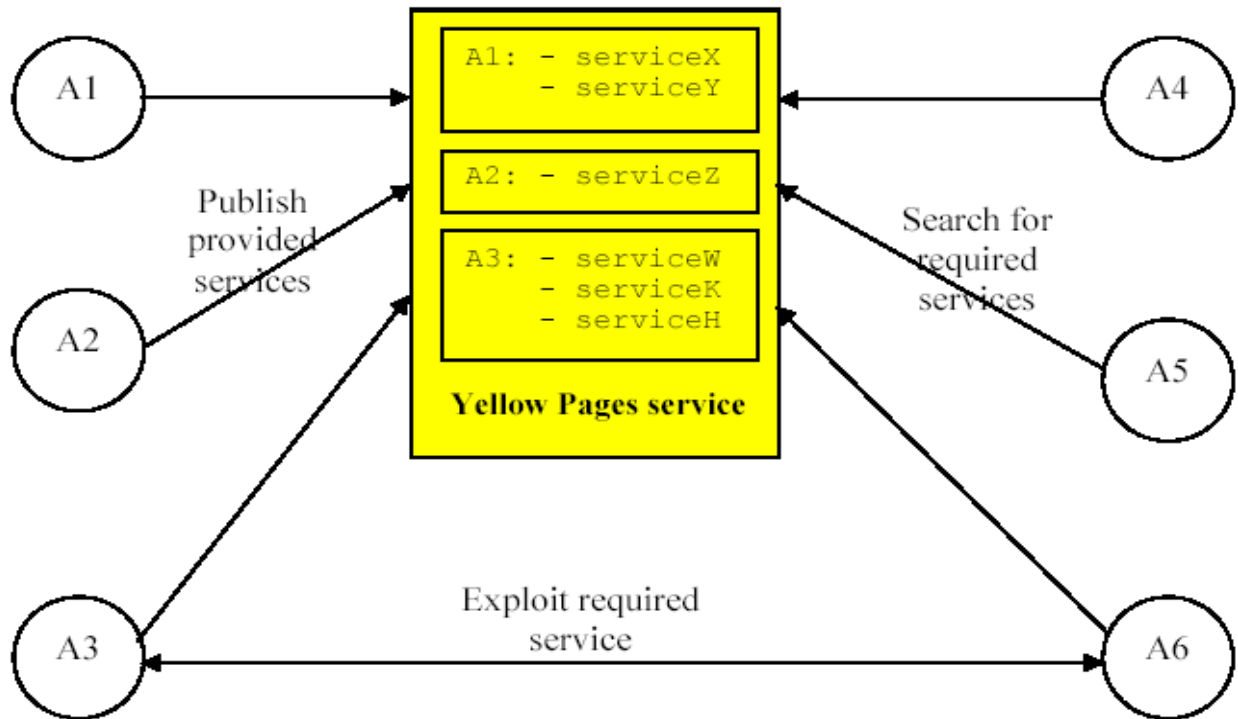


Рисунок 2.15 – Сервіс yellow pages

Кожна платформа JADE за замовчуванням підтримує агента DF (його локальне ім'я «df»). Можуть бути запуснені і об'єднано кілька DF-агентів (включаючи ті, що знаходяться в платформі за замовчуванням), для забезпечення єдиного розподіленого каталогу «жовтих сторінок». Оскільки DF є агентом, є можливість взаємодіяти з ним стандартним чином, обмінюючись ACL-повідомленнями, використовуючи, згідно специфікаціям FIPA, відповідний мова змісту (мова SLO) і відповідну онтологію (онтологію FIPA-agent-management). Щоб спростити ці взаємодії, JADE надає клас `jade.domain.DFService`, з допомогою якого можна публікувати і виконувати сервіси через виклики методів.

2.9.2 Діаграма компонентів системи

Діаграми компонентів - це один з двох видів діаграм, які застосовуються при моделюванні фізичних аспектів об'єктно-орієнтованої системи (другий вид - діаграми розгортання). Вони показують організацію наборів компонентів і залежності між ними. Діаграми компонентів застосовуються для моделювання

статичного виду системи з точки зору реалізації. Сюди відноситься моделювання фізичних сутностей, розгорнутих у вузлі, наприклад виконуваних програм, бібліотек, таблиць, файлів і документів. Діаграми компонентів важливі не тільки для візуалізації, специфікації і документування системи, заснованої на компонентах, але і для створення виконуваних систем шляхом прямого і зворотного проектування.

Діаграма компонентів (Component diagram) показує набір компонентів і відносини між ними. Графічно діаграма компонентів представляється у вигляді графа з ребрами і вершинами. Компоненти існують в матеріальному світі, у вигляді сукупності бітів і тому є важливими будівельними блоками при моделюванні фізичних аспектів системи. Компонент - це фізична замінна частина системи, сумісна з одним набором інтерфейсів і забезпечує реалізацію будь-якого іншого.

Діаграма компонентів системи представлена на рисунку 2.16.

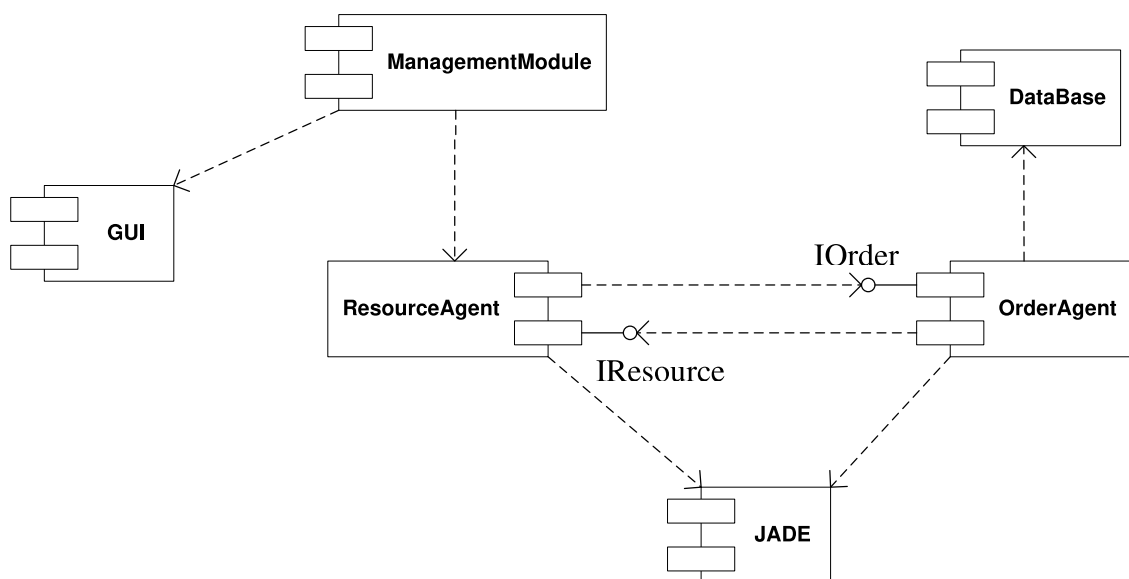


Рисунок 2.16 – Діаграма компонентів

Інтерфейс - це набір операцій, які описують послуги, що надаються класом або компонентом. Істотним є відношення між компонентом і інтерфейсом. Можна побудувати декомпозицію фізичної реалізації шляхом специфікації інтерфейсів, що представляють основні стикувальні вузли системи.

ManagementModule – компонент управління виробничими графіками;

DataBase – загальна база даних підприємства;

ResourceAgent – агент ресурсу;

OrderAgent – агент замовлення;

GUI – графічний інтерфейс користувача для відображення виконавчого графіку;

JADE – бібліотека мультиагентної платформи.

2.9.3 Розгортання системи

Вузол, як і компонент, що існує в матеріальному світі і є важливим будівельним блоком при моделюванні фізичних аспектів системи. Вузол - це фізичний елемент, який існує під час виконання та представляє обчислювальний ресурс, зазвичай володіє як мінімум деяким об'ємом пам'яті, а часто також і процесором. Вузли використовуються для моделювання, топологія апаратних засобів, на яких виконується система. Як правило, вузол - це процесор або пристрій, на якому можуть бути розгорнуті компоненти.

Діаграма розгортання показує конфігурацію вузлів, де проводиться обробка інформації, і те, які компоненти розміщені на кожному вузлі.

Діаграми розгортання використовуються для моделювання статичного виду системи з точки зору розгортання. В основному під цим розуміється моделювання, топологія апаратних засобів, на яких виконується система. По суті, діаграми розгортання - це просто діаграми класів, зосереджені на системних вузлах.

Діаграма розгортання системи представлена на рисунку 2.17

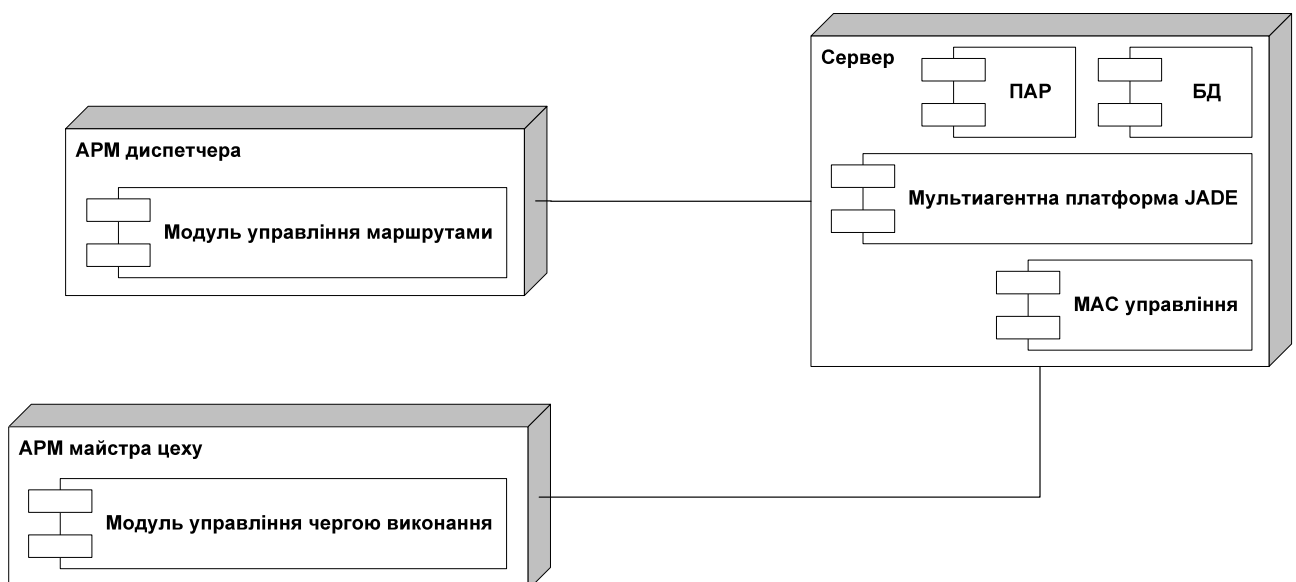


Рисунок 2.17 – Діаграма розгортання системи

На сервері де вже розгорнуто загальну базу даних підприємства та автоматизовану програму розкрою, для функціонування системи необхідно розгорнути мультиагентна платформу JADE та розроблену МАС управління. На робочій станції майстра цеху розгортається модуль управління чергою виконання замовлень на виробничих дільницях. На робочій станції диспетчера розгорнуто модуль управління маршрутами. Всі вузли системи поєднані загальною локальною мережею підприємства.

3 РОЗРОБКА РОБОЧОЇ ДОКУМЕНТАЦІЇ ПО ПРОЕКТУ

Розроблена система являє собою інструмент управління матеріальними потоками підприємства спеціалізованого на розкрій листових матеріалів на етапах виробництва та збуту. Основними функціональними характеристиками системи є:

- Побудова розкладів і маршрутів на основі переговорів програмних агентів;
- Перепланування графіків в реальному часі;
- Узгодження планів з користувачем «на льоту»;
- Реакція на обробку замовлень в «реальному часі».

Розроблений варіант вирішення проблеми управління ресурсами на підприємстві являє собою систему автоматизації побудови і управління виконавчими графіками. Система автоматизує наступні функції:

- Побудова черги виконання нарядів на розкрій для кожної виробничої ділянки;
- Управління та корегування черг виконання;
- Побудова розкладів маршрутів розвозу готових замовлень;
- Включення в отриманий маршрут готових замовлень заявки на перевезення вантажу з оптових баз;
- Управління та корегування маршрутів.

При впровадженні програмного забезпечення розробленої системи основними цілями є:

- зменшити витрати часу керівників всіх рівнів на пошуки потрібної інформації і подання звітів про діяльність підлеглих підрозділів, що дозволить їм більш ефективно виконувати свої функції;
- знизити ризики помилок у плануванні виробництва і прийнятті рішень з його управління;
- підвищити швидкість процесу обробки інформації і знизити витрати, пов'язані з ним;

— підвищити рівень організації і досвіду робітників в використанні інформаційних технологій, що підготує умови для наступного етапу інформатизації підприємства.

Робоча документація на інформаційну систему управління матеріальними потоками підприємства листового розкрою потрібна для розробки і налагодження даної системи в цілому.

Спроектована система відповідає вимогам які викладені в технічному завданні (Додаток А).

При роботі з розробленим програмним забезпеченням системи варто керуватися документом «Технічний опис автоматизованої системи управління матеріальними потоками підприємства» (Додаток Б).

Розроблений документ «Програма та методика випробувань» представлено в додатку В.

Лістинг програмного коду представлено в додатку Г.

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ РОЗРОБЛЕНОЇ СИСТЕМИ

4.1 Вступ

В даний час сучасні обчислювальні засоби широко використовуються в повсякденній діяльності людини. Сьогодні обчислювальна техніка дозволяє вирішувати найрізноманітніші задачі: звільнення людини від одноманітної ручної праці до розробки складних технічних розрахунків та моделювання складних наукових процесів.

Сучасні засоби програмування та апаратні можливості комп'ютерної техніки дозволяє вирішувати в домашніх умовах задачі, що стояли в минулому перед цілими колективами наукових співробітників та вимагали великих обчислювальних комплексів.

Дана дипломна робота присвячена розробці проекту інформаційної управляючої системи управління матеріальними потоками підприємства листового розкрою з метою підвищення ефективності роботи персоналу і як наслідок підвищення прибутку фірми.

Обґрунтувати доцільності розробки системи з економічної точки зору можливо розрахувавши собівартість програмного забезпечення і додатковий прибуток, який буде отриманий в результаті її впровадження. Основним показником, що визначає економічну доцільність витрат на створення програмного забезпечення є річний економічний ефект. Характерним показником економічної ефективності створення і функціонування програмного продукту, що розраховується на стадії створення, впровадження і використання, є ефективність витрат і розрахунок коефіцієнту економічної ефективності капіталовкладень (термін окупності програмного забезпечення).

На першому етапі розраховується собівартість розробки програмного забезпечення. Потім капітальні витрати, необхідні для придбання устаткування. На підставі цього можна одержати річний економічний ефект і строки окупності даного проекту.

4.2 Розрахунок витрат на створення і експлуатацію інформаційної системи управління матеріальними потокам підприємства листового розкрою

Розробка програмного забезпечення виконується двома програмістами відділу, місячний оклад кожного складає 1000 грн. Додаткова заробітна плата складає 20% від основної. Виходячи з цього, основна і додаткова заробітна плата розробників системи 2400 грн., а вартість сучасних робочих станцій складає 3200 грн. (приведена вартість на базі Pentium IV). При вартості кіловат-години електроенергії рівної 0.39 грн., розраховується вартість розробки даної інформаційної системи.

Витрати на допоміжні матеріали приведені в табл. 5.1

Таблиця 4.1 - Витрати на допоміжні матеріали

Пункт витрат	Сума, грн.
Папір	25.00
Заправка картриджа до принтера	45.00
Носії інформації (диски/дискети)	15.00
Необхідна спеціалізована література	60.00
Непередбачені зайві витрати	20.00
Разом матеріали та комплектуючі вироби	Σ 165.00

Вартість розробки системи управління матеріальними потоками розрахуємо по формулі:

$$C_{\text{пр}} = (Z_{\text{зн}} + Z_{\text{сз}} + Z_{\text{зг}} + Z_{\text{е}} + Z_{\text{ін}}) * T + Z_{\text{м}} \quad (5.1)$$

де T - тривалість розробки, мес.

$Z_{\text{зн}}$ - основна і додаткова заробітна плата обслуговування персоналу, грн.

$Z_{\text{сз}}$ - відрахування на соціальні заходи (38% від основної і додаткової заробітної плати), грн.

$Z_{\text{зг}}$ - загальногосподарські витрати (50% від основної заробітної плати), грн.

$Z_{\text{е}}$ - витрати на електроенергію ;

$Z_{\text{м}}$ - витрати на основні і допоміжні матеріали ;

З_е при споживанні потужності 0.25 Вт, тривалості роботи на місяць, рівної $21 \cdot 6 = 126$ годин, вартості кіловат-години електроенергії 0.39 грн.,

$$З_е = 126 \cdot 0.39 \cdot 0.25 = 12.285 \text{ грн.}$$

З_{ін} – Інші витрати (5% від (З_{ст}+З_{сз}+З_{зг}+З_е+З_м))

Витрати на розробку системи показані в табл. 5.2

Таблиця 4.2 - Витрати на розробку системи

Найменування витрат	Одиниця виміру	Кількість
Тривалість розробки	мес.	3
Основна і додаткова заробітна плата	грн.	2400
Відрахування на соціальні заходи	грн.	912
Загальногосподарські витрати	грн.	1200
Витрати на основ, і доп. матеріали	грн.	165
Витрати на електроенергію	грн.	36,855
Інші витрати	грн.	235,692

Відповідно до формули 5.1 вартість розробки системи:

$$С_{пр} = (2400 + 912 + 1200 + 36,855 + 235,692) \cdot 3 + 165 = 14518,64 \text{ грн.}$$

Амортизаційні відрахування на устаткування складають 60% балансової вартості в рік:

$$А_{об} = 3200 \cdot 0.60 = 1920 \text{ грн.}$$

У масштабах підприємства річні витрати на основні і допоміжні матеріали (папір, диски/дискети) визначають в розмірі 5% вартості основного устаткування:

$$В_м = 3200 \cdot 0.05 = 160 \text{ грн.}$$

Річний обсяг робіт ПЄВМ у годинах визначається в такий спосіб:

$$\Phi_м = 253.3 \cdot T_з \quad (5.2)$$

де Т_з - середнє місячне завантаження устаткування (близько 4 годин)

253.3 - середня кількість робочих днів у році. Отже, річний обсяг роботи робочої станції розробника складає:

$$\Phi_м = 253.3 \cdot 4 = 1013 \text{ годин}$$

Витрати на електроенергію З_е при 1013 годинах роботи устаткування в рік складає:

$$З_е = 1013 \cdot 0.25 \cdot 0.39 = 98,76 \text{ грн.}$$

Експлуатаційні витрати для робочих станцій розробників за рік складуть:

$$З_{\text{ер}} = A_{\text{об}} + B_{\text{м}} + З_{\text{е}} + З_{\text{інш}}$$

де $Z_{\text{ер}}$ - експлуатаційні витрати для ПЄОМ за рік ;

$A_{\text{об}}$ - Амортизаційні відрахування на устаткування;

$B_{\text{м}}$ - річні витрати на основні і допоміжні матеріали;

$Z_{\text{е}}$ - витрати на електроенергію ;

$Z_{\text{інш}}$ - Інші витрати (5% від $(A_{\text{об}} + B_{\text{м}} + Z_{\text{е}})$)

$$Z_{\text{інш}} = (1920 + 160 + 98,76) * 0,05 = 108,94$$

$$Z_{\text{ер}} = 1920 + 160 + 98,76 + 108,94 = 2287,7 \text{ грн.}$$

Отже, у перший рік витрати на створення і експлуатацію програми складуть:

$$Z_{\text{се}} = 14518,64 + 2287,7 = 16806,34 \text{ грн.}$$

4.3 Розрахунок економічної ефективності розробки і впровадження системи

Традиційно процес керування інформаційними ресурсами комерційних підприємств ведеться ручним способом. При цьому існує необхідність у введенні чисельних карток, паперових реєстрів і списків.

Визначимо прямий економічний ефект (чистий доход), ґрунтуючись на тому, що впровадження програмного продукту з урахуванням витрат на супровід програми вивільняє 0.8 працівників (за експертною оцінкою фахівців обраного підприємства).

Зарплата 0.8 працівників в рік складає:

$$(2400 * 12) * 0.8 = 23040 \text{ грн}$$

Прямий економічний ефект (чистий доход) впровадження проекту інформаційної управляючої системи приведено в таблиці 5.3

Таблиця 4.3 – Прямий економічний ефект впровадження проекту системи

У перший рік	$23040 - 16806,34 = 6233,66 \text{ грн}$
У наступні	$23040 - 2287,7 = 20752,3 \text{ грн}$

4.4 Розрахунок строку окупності програми

За перший рік чистого прибутку буде на 6233,66 грн. У наступному році залишок витрат буде: $14518,64 - 6233,66 = 8284,98$ грн.

$$(8284,98 / 20752,3) * 12 = 4.8 \text{ місяць}$$

У такий спосіб визначено, що витрати на створення програмного продукту окупляться протягом 1 року і 4,8 місяців

4.5 Висновок

Виходячи з розрахунків, розроблена в даній роботі систем окупить себе через 1 рік і 4,8 місяців. Дана система є більш зручною в користуванні і простій у налаштуванні. Вона дозволяє розширити можливості користувачів системи, спрощує ведення документації та оперативного управління на підприємстві листового розкрою. Усе це вказує на те, що існує необхідність створення даної системи.

5 ОХОРОНА ПРАЦІ КОРИСТУВАЧІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ МАТЕРІАЛЬНИМИ ПОТОКАМИ НА ПІДПРИЄМСТВІ

5.1 Вступ

Охорона праці — це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів та засобів, спрямованих на збереження здоров'я і працездатності людини в процесі праці (стаття 1 Закону «Про охорону праці»).

В статті 4 Закону «Про охорону праці» закріплені такі принципи:

- пріоритет життя і здоров'я працівників відносно результатів виробничої діяльності підприємства, повна відповідальність власника за створення безпечних і нешкідливих умов праці;
- комплексне вирішення завдань охорони праці на засадах національних програм з цих питань, а також з урахуванням інших напрямків економічної і соціальної політики, досягнень в галузі науки, техніки та охорони здоров'я;
- соціальний захист працівників, повне відшкодування шкоди особам, потерпілим від нещасних випадків на виробництві та професійних захворювань;
- установлення єдиних нормативів з охорони праці для всіх підприємств незалежно від форм власності і видів їх діяльності;
- використання економічних методів управління охороною праці, проведення політики пільгового оподаткування, що сприяє створенню безпечних та нешкідливих умов праці, участь держави в фінансуванні заходів з охорони праці;
- здійснення навчання населення, професійної підготовки та підвищення кваліфікації працівників з питань охорони праці;
- забезпечення координації діяльності державних органів, установ, організацій та об'єднань громадян, що вирішують різні проблеми охорони здоров'я, гігієни та безпеки праці, а також співробітництво.

Підвищення технічного рівня виробництва нерозривно зв'язано із широким використанням різних видів електроніки, телебачення, оптичних квантових генераторів, електронно-обчислювальних машин. Необхідно відмітити що сучасні тенденції комп'ютеризація підприємства охоплює всі етапи його діяльності. Введення комп'ютерних технологій принципово змінило характер праці офісних працівників і вимоги до організації і охорони праці.

Недотримання вимог безпеки приводить до того, що через деякий час роботи за комп'ютером співробітник починає відчувати певний дискомфорт: у нього виникають головні болі і різь в очах, з'являються втома і дратівливість. У деяких людей порушується сон, погіршується зір, починають хворіти руки, шия, поперек і тому подібне.

Охорона праці – один з центральних інститутів трудового права. Він має виключно практичне значення. Недодержання вимог охорони праці створює небезпеку для здоров'я і життя працівників. У свою чергу і ті, кого законодавець називає власником або уповноваженим ним органом, несуть сувору, у тому числі й кримінальну, відповідальність за порушення правил охорони праці.

Власник підприємства або уповноважений ним орган повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників.

На власника або уповноважений ним орган покладається систематичне проведення інструктажу (навчання) працівників з питань охорони праці, протипожежної охорони.

Загальний нагляд за додержанням норм охорони праці покладено на прокуратуру, спеціальний — на професійні спілки. Контроль за безпекою праці здійснюють також державні й відомчі спеціалізовані інспекції.

5.2 Аналіз шкідливих та небезпечних факторів

Експлуатація ЕОМ супроводжується виникненням ряду шкідливих і небезпечних виробничих факторів фізичного і психофізіологічного характеру.

Особливістю роботи на ПК є постійне і значне напруження функцій зорового аналізатора, обумовлене необхідністю розрізнення об'єктів (символів, знаків і т.п.) в умовах: рядкової структури екрану, миготіння зображень, недостатньої освітленості поля екрану, недостатньою контрастності об'єктів розрізнення та необхідності постійної переадаптації зорового апарату до різних рівнів освітленості екрану, оригіналу, клавіатури.

Нервово-емоційне напруження при роботі на ПК виникає внаслідок дефіциту часу, великого об'єму і щільності інформації, особливостей діалогового режиму спілкування людини та ПК (збої, оперативне очікування, психологічні особливості роботи оператора, пов'язані з емоційно-вольовою сферою), відповідальності за безпомилковість інформації.

До числа факторів, що погіршують стан здоров'я користувачів, слід віднести електромагнітне і електростатичні поля, акустичний шум, зміна іонного складу повітря та параметрів мікроклімату в приміщенні. На стан користувачів роблять вплив і ергономічні параметри розташування екрану монітора, які ведуть до зміни контрастності зображення в умовах інтенсивної засвіченості, появи дзеркальних відблисків від передньої поверхні екрану монітора і т.д. Важливу роль грає і стан освітленості на робочому місці, параметри меблів і характеристики приміщення, де розташована комп'ютерна техніка.

Всі ці чинники викликані двома основними причинами: неправильною роботою за обладнанням і неправильним вибором обладнання.

Площа приміщення на одне робоче місце користувача повинна становити 6 м², а об'єм не менше ніж 20 м³. Не дозволяється розміщувати кабінети обчислювальної техніки у підвальних та цокольних поверхах. Покриття підлоги повинно бути матовим з коефіцієнтом відбиття 0,3-0,5. Поверхня підлоги має бути рівною, неслизькою, з антистатичними властивостями. Для внутрішнього оздоблення приміщень з ПК слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8; для стін 0,5-0,6.

Забороняється застосовувати для оздоблення інтер'єру приміщень з ПК класів полімерні матеріали (ДС/ДВ плити, шпалери, що миються, рулонні синтетичні матеріали, шаруватий паперовий пластик тощо), які виділяють у повітря шкідливі хімічні речовини, що перевищують гранично допустимі норми.

При розташуванні елементів робочого місця користувача ПК слід враховувати: робочу позу користувача, простір для розміщення користувача, можливість огляду елементів робочого місця, можливість ведення записів, розміщення документації і матеріалів, які використовуються користувачем.

Конструкція робочого місця користувача ПК має забезпечити підтримання оптимальної робочої пози. Робочі місця з ПК слід так розташовувати відносно вікон, щоб природне світло падало збоку переважно зліва (рисунок 5.1).

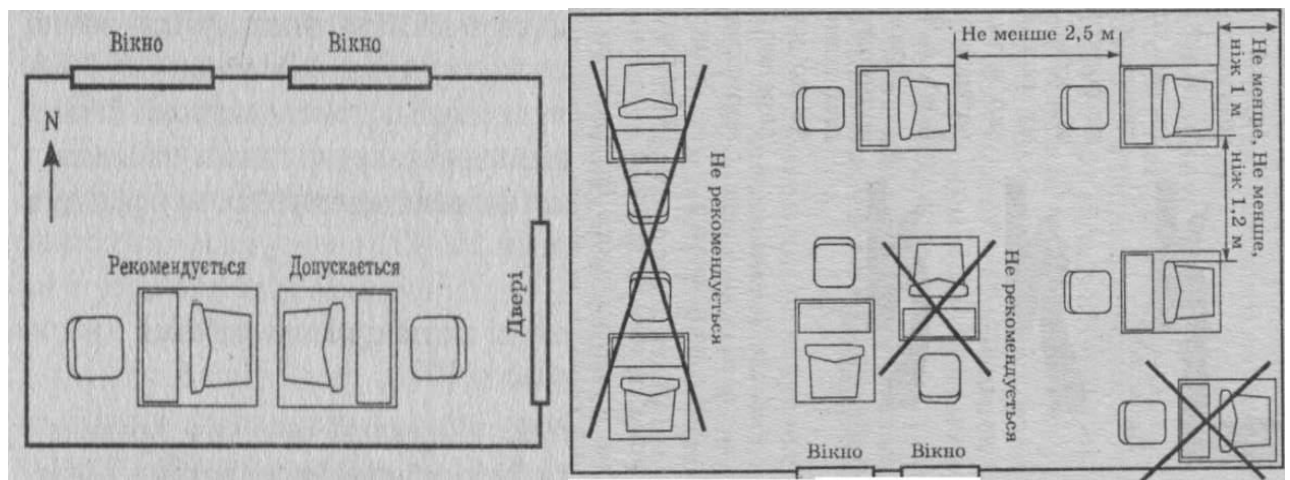


Рисунок 5.1 – Розташування робочих місць

Робочі місця з ПК повинні бути розташовані від стіни з вікнами на відстані не менш ніж 1,5 м, від інших стін - на відстані не менше ніж 1 м. При розміщенні робочого місця поряд з вікном кут між екраном монітора і площиною вікна повинен складати не менше 90° (для виключення відблисків), частину вікна, що прилягає, бажано зашторити. Недопустиме розташування ПК, при якому працюючий повернений обличчям або спиною до вікон кімнати або до задньої частини ПК, в яку монтуються вентилятори. При розміщенні робочих столів з ПК слід дотримуватись таких відстаней: між бічними поверхнями ПК – 1,2 м, від тильної поверхні одного ПК до екрана іншого ПК – 2,5 м.

Монітор повинен бути встановлений таким чином, щоб верхній край екрана знаходився на рівні очей. Екран монітора ПК має розташовуватися на оптимальній відстані від очей користувача, що становить 600-700 мм, але не ближче ніж 600 мм з урахуванням розміру літерно-цифрових знаків і символів (рисунок 5.2).

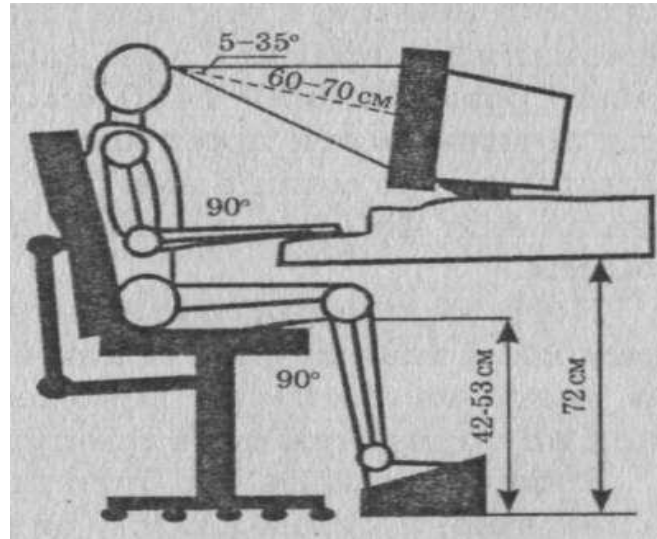


Рисунок 5.2 – Робоче положення

Для забезпечення точного та швидкого зчитування інформації в зоні найкращого бачення площина екрана монітора повинна бути перпендикулярною нормальній лінії зору. Розташування екрана монітора ПК має забезпечувати зручність зорового спостереження у вертикальній площині під кутом 30° до нормальної лінії погляду користувача.

Клавіатура повинна бути розташована так, щоб на ній було зручно працювати двома руками. Клавіатуру слід розміщати на поверхні столу на відстані 100-300 мм від краю. Кут нахилу клавіатури до столу повинен бути в межах від 5° до 15°, зап'ястя та долоні рук мають розташовуватися горизонтально до площини столу.

Висота робочої поверхні столу з ПК має регулюватися в межах 680-800 мм, а ширина і глибина - забезпечувати можливість виконання операцій у зоні досяжності моторного поля (рекомендовані розміри: 600-1400 мм, глибина - 800-1000 мм). Робочий стіл повинен мати простір для ніг висотою не менше ніж 600 мм, шириною - не менше ніж 500 мм, глибиною (на рівні колін) - не менше ніж 450 мм, на рівні простягнутої ноги - не менше ніж 650 мм.

Робочий стілець має бути підйомно-поворотним, регульованим за висотою, з кутом нахилу сидіння та спинки, поверхня сидіння має бути плоскою, передній край - заокругленим. Регулювання за кожним із параметрів має здійснюватися незалежно, легко і надійно фіксуватися. Висота поверхні сидіння має регулюватися в межах 400-500 мм, а ширина і глибина становити не менше ніж 400 мм. Кут нахилу сидіння - до 15° вперед і до 5° - назад. Висота спинки стільця має становити 300 ± 20 мм, ширина - не менше ніж 380 мм. Кут нахилу спинки має регулюватися в межах $1-30^\circ$ від вертикального положення. Відстань від спинки до переднього краю сидіння має регулюватися в межах 260-400 мм.

Для зниження статичного напруження м'язів верхніх кінцівок слід використовувати стаціонарні або змінні підлокітники завдовжки не менше ніж 250 мм, завширшки - 50-70 мм, що регулюються за висотою над сидінням у межах 230-260 мм і відстанню між підлокітниками у межах 350-500 мм.

Поверхня сидіння і спинки стільця має бути напівм'якою з нековзним повітронепроникним покриттям, що легко чиститься і не електризується.

Робоче місце має бути обладнане підставкою для ніг шириною не менше ніж 300 мм, глибиною - не менше ніж 400 мм, що регулюється за висотою в межах до 150 мм і за кутом нахилу опорної поверхні підставки - до 20° . Підставка повинна мати рифлену поверхню і бортик по передньому краю висотою 10 мм.

5.3 Розрахунок системи штучного освітлення на підприємстві

Основною проблемою виробництва, зв'язаною з освітленням – є брак через суб'єктивне читання технічної документації (карт розкрою, креслень). Тому розрахунок штучного освітлення буде проводитись в приміщенні конструкторсько-технологічного відділу.

Параметри приміщення: довжина $A=45$ м, ширина $B=16,5$ м, висота $H=4,5$ м. Висота робочої поверхні $h_p=1$ м. Для освітлення приймаємо світильник типу «НСП-07». Мінімальна освітленість лампи розжарювання з нормами $E_{\min}=100$ лк. Коефіцієнт відображення стелі $S_n=50\%$, стін $S_c=30\%$, робочої поверхні $S_p=10\%$. Напруга у мережі 220 В.

Розрахунок:

1 Відстань від стелі до робочої поверхні:

$$H_0 = H - h_p = 4,5 - 1 = 3,5 \text{ м}$$

2 Відстань від стелі до світильника:

$$h_c = 0,2 * H_0 = 0,2 * 3,5 = 0,7 \text{ м}$$

3 Висота підвісу світильника над поверхнею, що освітлюється:

$$h = H_0 - h_c = 3,5 - 0,7 = 2,8 \text{ м}$$

4 Висота підвісу світильника над підлогою:

$$H_{\Pi} = h + h_p = 2,8 + 1 = 3,8 \text{ м}$$

5 Для досягнення рівномірності освітлення приймаємо відношення $L/h=1,5$. Тоді відстань між центром світильників:

$$L = 1,5 * h = 1,5 * 2,8 = 4,2 \text{ м}$$

6 Необхідна кількість світильників (рисунок 1):

$$N = S/L^2 = 45 * 16,5 / 4,2^2 = 42 \text{ світ.}$$

Приймаємо 44 світильників для повного заповнення кожного ряду. (по 11 світильників в чотирьох рядах).

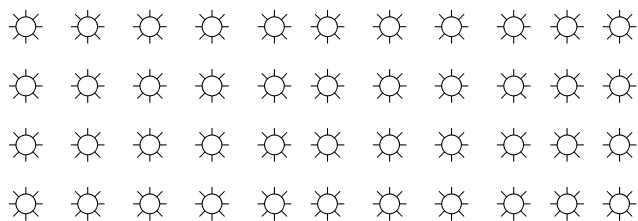


Рисунок 5.3 – Розташування світильників у виробничому приміщенні

7 Індекс приміщення:

$$I = (A * B) / (h * (A + B)) = 45 * 16,5 / (2,8 * 61,5) = 4,3$$

8 За таблицею коефіцієнтів використання світлового потоку при $i=4,3$, $S_n=50\%$, $S_c=30\%$, $S_p=10\%$ для світильника типу «НСП-07» коефіцієнт використання світлового потоку:

$$\eta = 0,73$$

9 Світловий потік однієї лампи:

$$\Phi_p = (E_{\min} * S * K_3 * Z) / (N * \eta),$$

де $S=A*B$ – площа приміщення, K_3 – коефіцієнт запасу, Z – тип лампи.

$$\Phi_p = (100 * 45 * 16,5 * 1,3 * 1,15) / (44 * 0,73) = 3455,9 \text{ лм}$$

10 За знайденим світловим потоком з таблиці «Параметри лампи розжарювання загального призначення з розрахунковими напругами 130 та 220В» обираємо лампу потужністю 220 Вт, що має світловий потік 4610 лм найбільш близький до розрахункового.

11 Фактична освітленість E_ϕ буде дорівнювати:

$$E_\phi = E_{\min} * (\Phi_l / \Phi_p) = 133,3 \text{ лк}$$

12 Загальна потужність:

$$P_{\text{заг}} = P_l * N = 13200 \text{ Вт} = 13,2 \text{ кВт.}$$

Отже для оптимальної роботи в приміщенні необхідно встановити 4 ряди по 11 світильників. Загальна потужність усіх світильників складає 13,2 кВт.

5.4 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з персональним комп'ютером

Для зниження впливу електромагнітного випромінювання треба застосовувати ЖК монітори. Монітори, які відповідають останнім стандартам ТСО'99 та MPR II є найбільш безпечними для здоров'я людини. Вони визначають максимально припустимі величини випромінювання магнітного й електричного полів, а також методи їхнього виміру. MPRII базується на концепції про те, що люди живуть і працюють у місцях, де вже є магнітні й електричні поля, тому пристрою, які ми використаємо, такі, як монітор для комп'ютера, не повинні створювати електричні й магнітні поля, більші ніж ті, які вже існують. Помітимо, що стандарти ТСО вимагають зниження випромінювань електричних і магнітних полів від пристроїв настільки, наскільки це технічно можливо, поза залежністю від електричних і магнітних полів, що вже існують навколо нас.

Заходи щодо усунення небезпеки поразки електричним струмом зводяться до правильного розміщення устаткування та електричних кабелів. Інші заходи щодо забезпечення електробезпеки, збігаються з загальними заходами пожежо- та електробезпеки.

Для зниження величини виникаючих зарядів статичної електрики покриття технологічних підлог варто виконувати з одношарового полівінілхлоридного антистатичного лінолеуму. Іншим методом захисту є нейтралізація заряду статичної електрики іонізованим газом. У промисловості широко застосовуються радіоактивні нейтралізатори. До загальних мір захисту від статичної електрики можна віднести загальне і місцеве зволоження повітря.

В якості профілактичних заходів для забезпечення пожежної безпеки слід використовувати скриту електромережу, надійні розетки з пожежобезпечних матеріалів, силові мережі живлення устаткування виконувати кабелями, розрахованими на підключення в 3-5 разів більшого навантаження, включати й виключати живлення обладнання за допомогою штатних вимикачів. Треба регулярно робити очистку внутрішніх частин комп'ютерів, іншого устаткування від пилу, розташовувати комп'ютери на окремих не спалюваних столах. Для запобігання іскріння необхідно рідше встромляти і виймати штепсельні вилки з розеток.

При напруженій роботі за комп'ютером щогодини необхідно робити перерву на 15 хвилин через кожну годину і треба займатися іншою справою. Декілька разів на годину бажано виконувати серію легких вправ для розслаблення. Для поліпшення ергономічності робочого місця необхідно здійснити такі заходи:

- Додайте монітор так, щоб його верхня точка знаходилася прямо перед вашими очима або вище, що дозволить тримати голову прямо, і виключить розвиток шийного остеохондрозу. Відстань від монітора до очей має бути не менше 45 см;
- Стілець повинен мати спинку і підлокітники, а так само таку висоту, при якій ноги можуть міцно стояти на підлозі. Ідеальним буде придбання крісла з висотою, в такому випадку спинка дозволить тримати спину прямо, підлокітники дадуть можливість відпочити рук, правильне положення ніг не буде заважати кровообігу в них;
- Розташування часто використовуваних речей не повинно призводити до довгого знаходження в якій або викривленої позі;

- Освітлення робочого місця не повинно викликати відблиски на екрані монітора. Не можна ставити монітор поруч з вікном, так щоб ви одночасно бачили і екран і те, що знаходиться за вікном.
- При роботі з клавіатурою, кут згину руки в лікті повинен бути прямим (90 градусів);
- При роботі з мишкою кисть повинна бути прямою, і лежати на столі як можна далі від краю.

При роботі за персональним комп'ютером в цілях безпеки користувачів категорично забороняється:

- Включати і вимикати комп'ютер без необхідності (це може призвести до його несправності);
- Чіпати роз'єми з'єднувальних кабелів, проводів, вилки і розетки;
- Торкатися до екрану і до тильної сторони блоків комп'ютера;
- Працювати на ПЕОМ мокрими руками;
- Працювати на ПЕОМ, що мають порушення цілісності корпусу, порушення ізоляції проводів, несправну індикацію включення живлення, з ознаками електричної напруги на корпусі.
- Класти на ПЕОМ сторонні предмети (гуртки з рідинами, жирні предмети, книги і предмети, що випромінюють електромагнітні поля).

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ

Охорона навколишнього середовища на підприємстві характеризується комплексом ужитих заходів, які спрямовані на попередження негативного впливу людської діяльності підприємства на навколишню природу, що забезпечує сприятливі та безпечні умови людської життєдіяльності.

Основними умовами для поліпшення екології в є: раціональне використання, охорона і витрачання запасів природного резерву, забезпечення безпеки екології та протирадіаційні заходи, підвищення та формування екологічного мислення у населення, а також контроль над екологією в промисловості. Охорона навколишнього середовища на підприємстві визначила ряд заходів для зниження рівня забруднень, що виробляється підприємствами:

- Виявлення, оцінка, постійний контроль і обмеження викиду шкідливих елементів в атмосферу, а також створення технологій і техніки, охороняють та зберігають природу і її ресурси.
- Розробка правових законів, спрямованих на охоронні заходи навколишнього середовища і матеріальне стимулювання виконаних вимог і профілактики комплексу природоохоронних заходів.
- Профілактика екологічної обстановки шляхом виділення спеціально відведених територій (зон).

Закон України «Про охорону навколишнього природного середовища» був прийнятий 26.06.91 р. і протягом останніх років систематично доповнювався й удосконалювався. Крім загальних норм і правил використання природного середовища, даний закон передбачає економічні заходи, які спрямовані на організацію раціонального природокористування і захист навколишнього природного середовища. У Законі також обговорені питання екологічної безпеки визначені природні об'єкти, що підлягають особливій охороні.

В існуючому законодавстві багато уваги приділяється питанням охорони навколишнього середовища. Жорсткість вимог до виробництва і матеріалів, а

також розробка нових виробничих і утилізаційних технологій дозволять зменшити антропогенне навантаження на навколишнє середовище. При масовому виробництві і використанні ПК не можна не враховувати їхній вплив на навколишнє середовище на всіх стадіях їхнього циклу – виробництво, експлуатація і після закінчення терміну їхньої служби.

На сьогоднішній день діють екологічні стандарти, що визначають вимоги до виробництва і матеріалів, що використовуються в конструкції приладу. Матеріали не повинні містити фріонів, що руйнують озоновий шар, хлоридів і бромідів. У стандартах ТСО-99 наводяться вимоги по замісту кадмію у світлочутливому шарі екрана дисплея і замістові ртуті в елементах харчування. Апарати, тара, документи повинні підлягати нетоксичної переробки після використання. Міжнародні стандарти, починаючи з ТСО-92, включають вимоги зниженого енергоспоживання й обмеження припустимих рівнів потужності в режимі споживання.

Підрозділи, що регулюють природоохоронну діяльність, представлені в органах законодавчої, виконавчої та судової властей.

Екологічний паспорт підприємства - це комплексний документ, що містить характеристику взаємин підприємства з навколишнім середовищем. Екологічний паспорт підприємства складається з двох частин. Перша частина містить загальні відомості про підприємство, використовуваний сировині, опис технологічних схем вироблення основних видів продукції, схем очищення стічних вод і викидів в атмосферу, їхньої характеристики після очищення, дані про тверді та інші відходи, а також відомості про наявність у світі технологій, що забезпечують досягнення найкращих питомих показників по охороні природи. Друга частина паспорта містить перелік планованих заходів, спрямованих на зниження навантаження на навколишнє середовище, із зазначенням строків, об'ємів витрат, питомих і загальних об'ємів викидів шкідливих речовин до і після здійснення кожного заходу.

6.2 Аналіз забруднення навколишнього середовища підприємством листового розкрою

Обране в роботі підприємство спеціалізується на виготовленні корпусних меблів. Основа його діяльності – розкрій листових матеріалів, які в основному представленні деревостружковими плитами – ДСП, ДВП, фанера, тощо.

Чотири основних принципу екологічної безпеки матеріалів для навколишнього середовища.

- Хімічна безпека: матеріали не повинні виділяти в повітря шкідливі леткі речовини, а концентрація яких-небудь летких речовин у повітрі не повинна перевищувати середньодобову концентрацію речовини в атмосферному повітрі - ГДК.
- Фізична безпека: матеріали повинні забезпечувати тепловий комфорт за величиною допустимих значень коефіцієнта теплопередачі і коефіцієнта теплопровідності; матеріали не повинні електризуватися і накопичувати на поверхні заряди статичної електрики; матеріали не повинні екранувати геомагнітне поле землі і випромінювання з космосу; при вітрових навантаженнях матеріали не повинні бути джерелом звукових коливань на частотах, шкідливих для здоров'я людини; матеріали для стін, перегородок і перекриттів повинні володіти ефективним звукопоглинанням.
- Пожежна безпека: всі матеріали, що застосовуються, повинні бути по категорії горючості не гірше Г2, з часом збереження конструкційної міцності при пожежі Р60.
- Біологічна безпека: всі матеріали, що застосовуються, повинні бути антисептировані не токсичними для людини антисептиками, не виділяють у повітря приміщень ніяких шкідливих речовин.

Проаналізуємо забруднення навколишнього середовища обраним в роботі підприємством на підставі шкідливості матеріалів, які використовуються в виробництві.

Фанера. Основний обсяг фанери представлений багатошаровою фанерою з лущеного шпону, одержуваної шляхом гарячого пресування пакету березового шпону, попередньо обробленого клеями на основі карбамидоформальдегідних і фенолформальдегідних смол. Всі фанери з карбамидформальдегідним сполучною виділяють у повітря формальдегід і метанол. Формальдегід є

канцерогенною речовиною і в цій якості внесений в список канцерогенів Всесвітньої організації охорони здоров'я. Метанол також відноситься до високотоксичним речовин. Досить широко відомо, що за європейськими стандартами ДПМ з формальдегідів сполучними за змістом формальдегіду в мг/100 г матеріалу діляться на три категорії: E0 - 6 і менше мг/100 г; E1 - від 9 до 7 мг/100 г; E2 - від 10 до 20 мг/100 г. Кращі види фанери, що випускаються відповідають лише класу E2. У країнах Східної та Західної Європи випускаються фанери класу E1, одержувані за рахунок застосування КФ-смола з різко зниженим вмістом формальдегіду та метанолу. Проте для широкого застосування в будівництві не підходить ні фанера E1, ні тим більше E2. Причини тут дві: 1) затверділі КФ-смоли, незалежно від будь-яких обставин, постійно відщеплюють формальдегід, і 2) рівень його виділення в повітря приміщень підвищується при підвищенні температури і вологості. Офіційно встановлено значення ГДК для формальдегіду, що дорівнює 0,003 мг/м³ повітря. В силу цього, будь фанера, використана для чорнової підлоги або обробки стін, стель, буде створювати в приміщенні концентрацію формальдегіду, в 10 і більше разів перевищує ГДК.

Для того щоб повернути фанеру як оздоблювальний матеріал у будівництво житла, необхідний цілий ряд серйозних заходів щодо поліпшення якості смоли та введенню до складу клеїв компонентів, активно поглинають (необоротно) формальдегід протягом всього терміну експлуатації фанери, тобто здійснити комплекс технологічних та композиційних нововведень, що зводяться до радикальної детоксикації фанери. Наявність у повітрі одночасно формальдегіду, метанолу і фенолу перетворює приміщення в справжню газову камеру. Переважна маса виробленої фанери відноситься до категорії горючості Г4, тобто до повністю горючих матеріалів. У той же час введення до складу клеїв для фанери ефективних антипіренів (одночасно є детоксикантами), а також обробка шпону після вирівнюючих вальців водними розчинами високоефективних і нетоксичних антипіренів дозволяють з невеликими змінами технології отримувати фанеру класу горючості Г2 і, при бажанні, Г1, з одночасним багаторазовим зниженням рівня виділення формальдегіду.

Деревостружкові плити - ДСП, ДВП. Жодна з різновидів ДСП, що виготовляються, не відповідає вимогам хімічної безпеки, перш за все із-за

постійного, протягом всього часу експлуатації виділення в повітря приміщень формальдегіду. Навіть при використанні ДСП класу Е1 перевищення концентрації формальдегіду в повітрі приміщень буде багаторазовим у порівнянні з ГДК. Ніяких претензій до ДСП з точки зору фізичної безпеки не має. У сухих приміщеннях ДСП відповідають критерію біологічної безпеки. У приміщеннях з підвищеною вологістю можуть з'являтися грибкові утворення, Переважна частина ДСП належить до класу горючості Г4 і без спеціальної протипожежної обробки не може бути використана в житловому будівництві. Таким чином, за двома найважливішими критеріями екологічної безпеки - хімічної і пожежної ДСП не можуть бути використані в житловому будівництві. Для серйозної реабілітації всіх різновидів ДСП необхідно: вводити до складу ДСП детоксиканти, необоротно поглинають формальдегід протягом всього терміну експлуатації зі швидкістю, що перевищує швидкість відщеплення формальдегіду від отвердження карбамідоформальдегідної смоли; вводити до складу ДСП ефективні антипірени постійної дії; при будівництві житлових приміщень обробляти внутрішню поверхню ДСП детоксикаційної та вогнезахисної ґрунтовкою, знижує як мінімум в 10 разів рівень виділення в повітря формальдегіду і повністю поглинає фенол. Оброблена ДСП переходить з класу горючості Г4 в клас Г1.

6.3 Розробка заходів щодо зменшення забруднення

Природоохоронна діяльність включають в себе комплекс технічних, технологічних, економічних, юридичних та організаційних заходів, що забезпечують захист навколишнього середовища від впливу виробничих процесів і спрямованих на раціональну взаємодію між діяльністю людини і навколишнім середовищем.

Технічні заходи передбачають впровадження у виробництво обладнання, що передбачає нейтралізацію, регенерацію і повторне використання застосовуваних у технологічному процесі матеріалів, а також

природоохоронного обладнання (фільтри, пиловловлюючі установки, установки локальної очистки та ін), що запобігає забрудненню атмосфери, ґрунту та води.

Технологічні заходи передбачають впровадження у виробництво щадних, маловідходних, безвідходних технологічних процесів при широкому використанні вторинних ресурсів, ефективної нейтралізації, регенерації (вода, повітря, розчини, речовини), удосконалення застарілих технологічних процесів, спрямованих на вирішення екологічних проблем.

Економічні заходи передбачають:

- розробку системи показників природоохоронної діяльності підрозділів;
- розроблення нормативної бази для планування, складання заходів і фінансового плану забезпечення проведення природоохоронних заходів;
- систему санкцій до підрозділів, що порушують норми і правила охорони навколишнього середовища;
- систему заохочення;
- за впровадження технології та обладнання, спрямованих на скорочення відходів виробництва, що роблять шкідливий вплив на охорону навколишнього середовища;
- за вирішення питань регенерації, рециркуляції, нейтралізації токсичних відходів;
- за збільшення використання вторинної сировини, відходів.

Юридичні заходи засновані на сукупності установлених державою правових норм, спрямованих на збереження та оздоровлення навколишнього середовища, раціональне використання природних ресурсів, дотримання природоохоронного законодавства, дотримання галузевих нормативних документів та інших наказів і інструкцій.

Організаційні заходи передбачають удосконалення взаємозв'язку між бюро охорони навколишнього середовища та іншими підрозділами підприємства та філіями, створення планів-заходів по забезпеченню охорони навколишнього середовища на підприємстві.

ВИСНОВКИ

В роботі було вирішено ряд наступних задач:

- Досліджено концепції та підходи до організації управління матеріальним потоком та визначені проблемні області;
- Проаналізовані автоматизовані системи вирішення задач логісти та виявлені їх характерні недоліки;
- Досліджено технологію МАС та визначено можливості, переваги та шляхи її застосування для підвищення ефективності процесу управління матеріальним потоком;
- Розроблено проект МАС управління ресурсами для підприємств листового розкрою на етапі виробництва та збуту.

Отже, на основі проведеного в роботі дослідження можна стверджувати, що використання мультиагентного підходу в управлінні матеріальними потоками підприємства має ряд своїх переваг:

- збалансований підхід до процесу планування та побудови розкладів;
- врахування індивідуальних вимог на рівні кожного окремо взятого замовлення та ресурсу;
- використання в процесі планування стратегій, інтуїтивно використовуваних людиною на базі досвіду;
- можливість ефективної і швидкої реакції на непередбачені події.

Застосування мультиагентних технологій в області управління матеріальними потоками приводить до кардинальних змін в самій організації логістики виробництва. Централізація, жорстке управління та глобальна оптимізація класичних систем замінюється децентралізованою організацією системи, переговорами між агентами та пошуком локальних оптимумів. Такий підхід забезпечує високу ефективність, гнучкість, адаптивність та надійність систем підприємства.

Мультиагентні системи будують не оптимальні плани, а плани приближені до оптимальних з задоволення всіх об'єктів системи з урахуванням їх індивідуальних інтересів задля досягнення внутрішньої гармонії. Але практика показує, що у виконанні будь-якого плану завжди відбуваються збої, і тому

навіть план, складений оптимальним чином, але який не передбачає вірогідних змін, гірше і менш ефективний, ніж план, не настільки оптимальний, але який дозволяє внести без великого збитку деякі зміни, викликані непередбаченими обставинами.

Однак, незважаючи на усі переваги мультиагентних систем, нові можливості мультиагентного підходу приводять до ряду нових проблем:

- В умовах постійних змін важко оцінити наскільки поточне рішення системи відрізняється від «оптимального»;
- Рішення системи залежить від історії виникнення подій;
- «Ефект метелика»: невеликі зміни на вході можуть призвести до, неочікуваних для спостерігача, великих змін на виході;
- Реакція системи може непередбачено сповільнюватися для спостерігача у разі виникнення довгого ланцюга змін та переговорів (потрібен контроль активностями в умовах обмеженого часу на відповідь);
- При повторному запуску системи, при тих самих вхідних даних рішення системи може бути іншим;
- Неможливо «відкотити» рішення назад, так як ситуація кардинально змінюється;
- Рішення системи іноді буває складно пояснити користувачу, в силу складних зв'язків та залежностей між агентами.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Аксьонов К.А., Неволіна А.Л., Аксьонова О.П., Смолій Є.Ф. Мультіагентне моделювання та планування логістики // Сучасні проблеми науки та освіти. - 2013. - № 4
2. Бажин І.І. Логістика: компакт-підручник [Текст]. - Харків: Консум, 2003. - 181 с.
3. Гранічина Н. О. Мультіагентна система для розподілу замовлень // УБС. 2010. №30-1. С.549-566.
4. Іващенко А. В. Адаптивне управління планом дрібносерійного виробництва на промисловому підприємстві / А. В. Іващенко, М. В. Андрєєв // Системи управління та інформаційні технології. - 2008. - № 3 (33). - С. 62-66.
5. Іващенко А. В. Мультіагентна система для управління виробництвом у реальному часі.
6. Джордж Ржевський. Мільтон Кейнс. Мультіагентні системи у логістиці
7. Люггер Д.О. Штучний інтелект: стратегії та методи вирішення складних проблем: [Текст]/Д. Люггер. - Видавничий дім "Вільямс", 2003. - 863 с.
8. Рассел С. Штучний інтелект: сучасний підхід, 2-ге вид [Текст].: Пер. з англ. п. ред. Птіцина А. К. / Рассел С., Норвіг П. // Вільямс, 2006, -1408 с.: іл.
9. Буч Г., Рамбо Д, Джекобсон А. UML посібник користувача / Г. Буч, Д. Рамбо, А. Джекобсон; Пров. з англ. - М.: ДМК Прес, 2001. - 432 с.
10. Прокопенко, Т. Крезуб, В. Моделювання інформаційних потоків в системах управління проектно-орієнтованими підприємствами / Вісник Черкаського державного технологічного університету. Серія: Технічні науки. 1., 2018. – с. 53-59. 10.24025/2306-4412.1.2018.153274.

1 Загальні положення

Назва інформаційної системи: «Інформаційна система управління матеріальними потоками підприємства листового розкрою».

Створюваний програмний продукт являє собою автоматизовану систему (АС) обліку та управління матеріально-технічними цінностями (ТМЦ) на підприємстві спеціалізованому на розкрої листових матеріалів на етапі виробництва і збуту. Дана автоматизована система забезпечує ведення, управління та побудову плану виробництва та маршрутів перевезень готових замовлень.

2 Призначення та мета створення системи

2.1 Призначення

Дана система призначена для автоматизації побудови, управління та ведення графіків виконання нарядів на розкрій та маршрутів розвозу готового продукту до замовників.

2.2 Мета створення

Метою створення даної системи є покращення ефективності роботи працівників підприємства, усунення можливих втрат інформації, економія часу та координація потоку матеріалів. Експлуатаційне призначення: скорочення термінів надання необхідної інформації для прийняття управлінських рішень, реалізація нових задач керування підприємством.

3 Характеристика об'єкту автоматизації

Об'єктом автоматизації даного проекту є облік потоку матеріалів на підприємстві спеціалізованому на розкрій листових матеріалів. Під автоматизацію підпадає такі етапи логістичного ланцюга як обробка та збут. Дані процеси описані в розділі 2 пункті 2.2 даної роботи.

4 Цілі та завдання розробки системи

Ціллю даної ІС є:

- збільшення кількості користувачів, що працюють із системою;
- зведення до мінімуму помилок обліку товару;
- підвищення достовірності даних;
- розмежування доступу до сховища даних по правах користувачів;
- створення єдиної інформаційної системи;
- підвищення надійності функціонування.

Створювана ІС повинна виконувати наступні функції:

- Побудова черги виконання нарядів на розкрій для кожної виробничої ділянки;
- Управління та корегування черг виконання;
- Побудова розкладів маршрутів розвозу готових замовлень;
- Включення в отриманий маршрут готових замовлень заявки на перевезення вантажу з оптових баз;
- Управління та корегування маршрутів.

5 Вимоги до системи

5.1 Вимоги до функціональних характеристик

Для майстру цеху система автоматизує:

- Алгоритм побудови виробничого плану;

- Перегляд та узгодження виробничого плану;
- Керування виробничим планом шляхом зміни пріоритетів замовлень та переносом замовлень з однієї ділянки на іншу.

Для диспетчера перевозок система автоматизує:

- Алгоритм побудови маршрутів перевезень;
- Перегляд та узгодження отриманого розкладу маршрутів;
- Керування розкладом перевезень.

5.2 Вимоги до надійності

Для коректного функціонування програмного додатку необхідно забезпечувати:

- Безперебійне живлення технічних засобів;
- Використання ліцензійного програмного забезпечення;
- Випробування програмних засобів на наявність комп'ютерних вірусів.

Необхідно слідкувати за коректністю введення даних. Якщо обов'язкові поля не заповнені програма повинна видати про це повідомлення і не продовжувати подальшу роботу то того часу як дані поля не будуть заповнені. Також якщо введена інформація не відповідає типу поля програма повинна видати про це попередження. При введенні помилкової інформації програма повинна забезпечувати її редагування.

5.3 Вимоги до експлуатації

Промисловий панельний комп'ютер повинний бути спеціально призначений для експлуатації в жорстких цехових умовах. Ця спецтехніка має найвищу ступінь захисту від пилу і відповідає рівню IP65 по стандарту ISO9001, що забезпечує пило-ізоляцію та безпроблемний контакт з водою.

Від користувача вимагається знання інструкції користувача, та загальні вміння користуватися ПК.

5.4 Вимоги до складу та параметрів технічних засобів

- серверна платформа і серверне програмне забезпечення;
- клієнтські платформи клієнтське програмне забезпечення (MS Windows 10);
- мережне середовище (100 Mbit);

Апаратне забезпечення повинне максимально повно охоплювати існуючий парк комп'ютерної техніки на підприємстві.

5.5 Вимоги до інформаційної та програмної сумісності

- Операційна система: Windows 10;
- Наявність програмного забезпечення Java Agent Developed Framework.

6 Вимоги до програмної документації

Програмна документація повинна містити:

- Технічне завдання;
- Технічний опис;
- Програму та методики випробувань;
- Інструкцію користувача;

7 Стадії та етапи розробки

Роботи по створенню системи включають в себе наступні етапи та стадії розробки:

- Перед проектне дослідження:
 - Аналіз предметної галузі;
 - Оформлення вимог користувача до системи;
- Розробка концепції системи:
 - Розробка варіантів концепцій системи та їх функціональний аналіз;

- Вибір варіанту концепції системи.
- Ескізне проектування:
 - Прийняття загальносистемних рішень;
 - Розробка попередніх рішень по вибору варіантів інформаційного, технічного та програмного забезпечення системи.
- Технічне проектування:
 - Прийняття проектних рішень по системі та її частинам;
 - Розробка технічної документації по проекту.
- Ввід в дію:
 - Виготовлення та налагодження компонентів системи;
 - Розробка акту передачі.

8 Порядок прийому та контролю

Порядок контролю і здачі даного програмного продукту виконується представником замовника у присутності представника розробника згідно програмі і методиці випробування.

Для контролю і здачі даного програмного продукту повинна бути представлена розроблена АС, методика випробувань і інструкція користувача.

За результатами здачі складається акт, який підписується представником замовника і представником розробника, і затверджується керівниками організації-замовника і організації-розробника.

ДОДАТОК Б – Технічний опис

1 Загальні відомості

Назва: «Автоматизована інформаційна система управління матеріальними потоками на підприємстві».

Для функціонування системи необхідні наступні програмні засоби:

- Операційна система Microsoft Windows 10 встановлена на робочих станціях та сервері;
- Наявність на сервері бібліотеки мультиагентної платформи Java Agent Developed Framework.
- Установлене на клієнтських місцях і на сервері ПЗ інформаційної системи підприємства.

Основним засобом розробки програмного забезпечення системи є програмне середовище NetBeans IDE. Програмне забезпечення виконане мовою програмування JAVA. Мультиагентна платформа – JADE (Java Agent Developed Framework).

2 Функціональне призначення

Система являє собою інструмент управління матеріальними потоками на етапі виробництва та збуту.

Створювана система реалізує наступні функції:

Для майстру цеху система автоматизує:

- Алгоритм побудови виробничого плану;
- Перегляд та узгодження виробничого плану;
- Керування виробничим планом шляхом зміни пріоритетів замовлень та переносом замовлень з однієї ділянки на іншу.

Для диспетчера перевозок система автоматизує:

- Алгоритм побудови маршрутів перевезень;
- Перегляд та узгодження отриманого розкладу маршрутів;

— Керування розкладом перевезень.

З урахуванням даних особливостей використання, основними функціональними характеристиками системи є:

- ведення загальної інформаційної бази з розподіленим доступом;
- авторизований доступ до даних на основі прав доступу;
- негайне відображення змін, внесених з будь-якого робочого місця;
- можливість введення і редагування планів робіт і оперативного контролю ходу їхнього виконання;
- формування необхідних вихідних документів.

3 Опис логічної структури

При розробці програмного забезпечення системи використані методи об'єктно-орієнтованого проектування і розробки. Основною структурною одиницею ПЗ є клас, діаграма класів системи приведена в проектній документації. Основними частинами програмного забезпечення є клієнтські додатки та серверна частина. На клієнтські додатки покладена функція передачі даних на сервер, а безпосередню обробку інформації в базі даних забезпечує серверна частина. Клієнтська частина реалізує також графічний інтерфейс користувача з використанням стандартних віконних засобів операційної системи.

4 Використовувані технічні засоби

Для функціонування програмного забезпечення системи використовуються:

- Комп'ютери середньої комплектації на ступного типу: CPU Pentium E2200 3000 MHz; RAM 2048MB; HDD 320GB; VIDEO GeForce 7300 GT 256 DDR2; MONITOR 21”;

- Мережеве обладнання типу: LAN Ethernet Card Surecom, 10/100Mbps, RJ-45; Ethernet Switch 16-port 10/100Mbps Base TX;
- Сервер з характеристиками типу: CPU: 16 (32 core); RAM: 128 Gb; HDD: 500 Gb; Network Card: 2 (2 Gbit); Fiber Channel: 4
- Інше обладнання: принтери;
- Ефективного функціонування комп'ютерної мережі виробничих приміщень для зв'язку з сервером необхідне кабель – 100Base-FX. В середині комп'ютерних мереж офісу та приміщень цеху використовується тип кабелю 100Base-TX.

ДОДАТОК В – Інструкція з використання

1 Призначення програми

Даний програмний продукт є системою обліку та управління матеріально-технічними цінностями (ТМЦ) на підприємстві.

Дана система призначена для автоматизації побудови, управління та ведення графіків виконання нарядів на розкрій та маршрутів розвозу готового продукту до замовників..

Система автоматизує наступні функції:

Для майстру цеху система автоматизує:

- Алгоритм побудови виробничого плану;
- Перегляд та узгодження виробничого плану;
- Керування виробничим планом шляхом зміни пріоритетів замовлень та переносом замовлень з однієї ділянки на іншу.

Для диспетчера перевозок система автоматизує:

- Алгоритм побудови маршрутів перевезень;
- Перегляд та узгодження отриманого розкладу маршрутів;
- Керування розкладом перевезень.

Програмне забезпечення функціонує в мережі підприємства, обмін даними будується через єдину інформаційну базу.

2 Умови виконання програми

Для виконання програми необхідна наявність на підприємстві наступних апаратних і програмних засобів:

Апаратні засоби:

- Комп'ютери середньої комплектації на ступного типу: CPU Pentium E2200 3000 MHz; RAM 2048MB; HDD 320GB; VIDEO GeForce 7300 GT 256 DDR2; MONITOR 21”;

- Мережеве обладнання типу: LAN Ethernet Card Surecom, 10/100Mbps, RJ-45; Ethernet Switch 16-port 10/100Mbps Base TX;
- Сервер з характеристиками типу: CPU: 16 (32 core); RAM: 128 Gb; HDD: 500 Gb; Network Card: 2 (2 Gbit); Fiber Channel: 4
- Інше обладнання: принтери;
- Ефективного функціонування комп'ютерної мережі виробничих приміщень для зв'язку з сервером необхідне кабель – 100Base-FX. В середині комп'ютерних мереж офісу та приміщень цеху використовується тип кабелю 100Base-TX.

Програмні засоби:

- Оптимальним варіантом операційної системи для використання та реалізації системи є ОС Microsoft Windows 7;
- Наявність на сервері мультиагентної платформи Java Agent Developed Framework.
- Установлене на клієнтських місцях і на сервері ПЗ інформаційної системи підприємства.

3 Опис функціональності

Система забезпечує управління ресурсами на підприємстві на етапі виробництва та збуту являє собою автоматизовану підсистему побудови та управління виконавчими графіками.

Мультиагентний підхід вирішення задачі побудови виконавчих графіків передбачає діяльність агентів-замовників та агентів-виконавців, які ставляться у відповідність реальним об'єктам предметної галузі представляючи їх інтереси. Черода переговорів даних типів агентів буде шляхом побудови виконавчого графіку. Агент-замовник починає планування шляхом пошуку необхідних йому ресурсів в моделі поточної ситуації, а саме, який агент-виконавець який розклад виконує. Модель ситуації представлена взаємозв'язками між агентами. Агенти кожного замовлення та ресурсу будують свої власні, але пов'язані в загальну мережу розклади.

Для побудови графіку виробництва створено агентів замовлення для представництва в черзі виконання інтересів кожного замовлення та агентів виробничих ділянок для формування в інтересах своєї ділянки графіку виконання. Також розроблено АРМ майстра цеху, для корегування отриманого графіку та фіксації факту виконання замовлень.

В системі вантажі на етапах закупки та збуту об'єднуються в один маршрут. Даний підхід полягає в тому, що завдяки застосуванню мультиагентної технології стає можливе динамічне перепланування маршрутів вантажівок підприємства в «реальному часі». Суть ідеї в тому, що вантажівки які розвозять готові замовлення будуть «підхоплювати» замовлення на доставку матеріалів з місцевих оптових баз на підприємство. Такий підхід дозволить відмовитись від великих за об'ємом планових поставок, які можуть не виправдатись, і при цьому не створити великих зайвих транспортних розходів на малі закупки під замовлення.

4 Виконання програми

Робота з програмним додатком комірника починається з головного меню яке пропонує два варіанти використання: «Обробити дані про надходження матеріалу» та «Обробити дані про стан складу». При виборі першого варіанту використання користувачу пропонується роботу з наступними формами: «Обробити дані про накладну», «Обробити дані про постачальника» та «Обробити дані про поставку». Форми обробки даних містять кнопки керування «ДОДАТИ», «ВИДАЛИТИ» та «КОРЕГУВАТИ». Форма вводу даних про накладну містить наступні поля: номер, дата, постачальник. Форма вводу даних про постачальника містить наступні поля: номер, Ф.И.О./назва фірми, контактні дані. Форма вводу даних про поставку містить наступні поля: номер накладної, код листа, кількість ціна. При виборі варіанту використання «Обробити дані про стан складу» користувачу пропонується обробити дані про лист, матеріал та отримати список зарезервованих листів при натиску відповідної кнопки. Форма

даних про листи має такі поля: код, матеріал, довжина, ширина, товщина, кількість, ціна. Форма матеріалу: код, текстура, тип, виробник.

Для розкрою замовлення в програмі автоматичного розкрою необхідно зарезервувати листи під замовлення. Для цього користувач натискає кнопку «СТВОРИТИ ЗАМОВЛЕННЯ» та заповнює такі поля: номер замовлення, дата. Після чого при натиску відповідної кнопки користувач отримує список наявних на складі листів. Обравши потрібний лист при натиску кнопки «ДОДАТИ» користувач додає лист до замовлення, тим самим блокує його використання для інших користувачів.

При введенні некоректних даних у полях введення виводиться діалогове вікно з описом типу і місця виникнення помилки.

При неправильному введенні чи пароля імені користувача виводиться повідомлення «У доступі відмовлено, невірне чи ім'я пароль».

При неузгодженості даних, що знову вводиться, із уже введеними раніше, користувачу виводиться діалогове вікно з попередженням про це, наприклад, «У замовленій відомості відсутня потрібна кількість МТЦ» і т.д. Прийняття рішень по цих повідомленнях (якщо вони не викликані помилкою користувача) вимагає внесистемних дій.

ДОДАТОК Г – Програма та методика випробувань

Об'єкт випробувань.

Об'єктом випробувань є автоматизована система підприємства листового розкрою для побудови та управління матеріальними потоками на етапі виробництва та збуту логістичного ланцюга.

Мета випробувань.

Будь-який програмний продукт може мати в собі невиявлені помилки, оскільки дуже важко протестувати або передбачити реакцію чи поведінку його на будь-яких можливих комбінаціях вхідних даних, котрі вводить користувач. В загальному випадку під помилкою розуміють неправильність, похибку або навмисне спотворення об'єкта чи процесу. При цьому ми розуміємо чи правильний стан об'єкта чи процесу відомий. Попередньо, перед створенням завантажувального файлу, компілятор виправляє деякі помилки, але передбачити можливі значення різних змінних він не в змозі.

Цілі випробувань: перевірка придатності системи до використання по призначенню, її відповідність заданим вимогам і документації; перевірка працездатності і правильності роботи системи.

Вимоги до системи.

Вимоги до функціональних характеристик розробленої системи поставлені у технічному завданні (додаток А). Відповідно до цих вимог система забезпечує наступні функції:

Для майстру цеху система автоматизує:

- Алгоритм побудови виробничого плану;
- Перегляд та узгодження виробничого плану;
- Керування виробничим планом шляхом зміни пріоритетів замовлень та переносом замовлень з однієї ділянки на іншу.

Для диспетчера перевозок система автоматизує:

- Алгоритм побудови маршрутів перевезень;
- Перегляд та узгодження отриманого розкладу маршрутів;

— Керування розкладом перевезень.

Вимоги до програмної документації.

У комплект програмної документації повинні входити наступні документи:

- Загальний опис системи;
- Керівництво користувача;
- Технічне завдання;
- Програма і методика випробувань;
- Текст програми.

Засоби та порядок випробувань.

Вимоги до складу і параметрів технічних і програмних засобів вказані в технічному завданні (додаток А). Для правильної установки і настройки програмного забезпечення системи необхідно скористатися технічним описом системи (додаток Б) та інструкцією користувача (додаток В). Сервера повинні бути об'єднані в комп'ютерну мережу з декількома клієнтами. Як мережне середовище можна використовувати кабелі типу «вита пара» категорії 3, 5 і вище. Мережне обладнання повинно підтримувати швидкість не менше 10 Мбіт/сек.

Випробування системи будуть проводитися протягом двох тижнів. Замовник і/або його представники самостійно проводять тестування усіх функцій системи в присутності розробника. При наявності відповідної можливості, у процесі тестування можуть додатково брати участь програмісти, тестувальники і системотехніки, які не брали участь у розробці системи.

Перед початком (не пізніше 3 днів) випробувань уточнюється кінцевий склад групи, що буде проводити випробування. Кожний член групи отримує необхідну документацію до системи (технічне завдання, загальний опис системи, керівництво користувача, інструкція з формування і ведення бази даних, керівництво адміністратора з установки програмного забезпечення на серверах, програма і методика випробувань).

Порядок проведення випробувань:

- Виконуються контрольні тести в довільному порядку. Результати заносяться у відповідний журнал.
- Робиться аналіз отриманих результатів і встановлюється відповідність програми заданим вимогам і системним документам.

При виявленні помилок у роботі системи складається їхній перелік і обговорюється термін їхнього виправлення розробником. Після цього замовник проводить повторне тестування в повному обсязі (можливо використання нових чи додаткових тестів).

Методика випробувань.

Процес тестування являє собою експлуатацію системи в контрольованих умовах і вивчення отриманих результатів. При цьому перевіряється робота системи з нормальними і помилковими даними і подіями. Тестування розподілених систем сильно ускладнено великою кількістю взаємозв'язків. Кожний елемент системи повинний тестуватися не тільки на коректність роботи, але і на сумісність один з одним у різних конфігураціях.

Найкраще для тестування розподілених систем застосовувати метод «знизу-нагору». Спочатку розробник повинен перевірити кожен компонент (елемент системи) окремо; потім – протестувати взаємодію компонента з іншими компонентами системи на автономному комп'ютері; і тільки потім – перевірити роботу всієї системи в розподіленому середовищі. Усі ці види тестування виконуються розробником системи без участі замовника. Розробник не зобов'язаний надавати тести і протоколи результатів тестування кому-небудь.

При прийманні системи проводяться випробування системи в розподіленому середовищі. Замовник і/або його представники самостійно проводять тестування усіх функцій системи методом «чорного ящика» у присутності розробника.

Програма випробувань

Тести складаються замовником і/або його представниками. При цьому перевіряється функціонування системи з нормальними і помилковими даними і подіями.

Наприклад:

- Авторизація користувача з правильними даними;
- Авторизація користувача з неправильними даними.

Під неправильними чи помилковими даними в контексті даної системи потрібно розуміти:

- Логін і/або пароль користувача, яких немає у файлах авторизації (для перших двох тестів).
- Неіснуючий порядковий номер в черзі (для всіх інших тестів).

Програмне забезпечення розроблено таким чином, що користувач не може ввести які-небудь інші неправильні дані в систему через наданий йому інтерфейс. Таким чином, будь-які дані, що не є неправильними, потрібно вважати правильними.

Для визначення переліку вхідних даних для кожного тесту можна скористатися формами для введення відповідних даних. У формах зазначені необхідні поля.

Вихідні дані чи результат при правильних даних повинні відповідати виконанню функції, для тестування якої створений тест; при неправильних даних — відмовлення в доступі до системи (тест 2) чи повідомлення про помилку.

ДОДАТОК Д – Текст програми

Лістинг 1 – клас OrderGUI.java

```
package GUI;

import OrderAgent.OrderAgent;

public class OrderGUI extends javax.swing.JFrame {

    OrderAgent orderAgent;

    public OrderGUI(OrderAgent a) {
        orderAgent = a;
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel2 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jLabel1 = new javax.swing.JLabel();
        jTextField2 = new javax.swing.JTextField();
        jLabel3 = new javax.swing.JLabel();
        jTextField3 = new javax.swing.JTextField();
        jLabel4 = new javax.swing.JLabel();
        jTextField4 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)), " Наряд на розкрій "));

        jLabel2.setText("Номер замовлення:");

        jLabel1.setText("Загальна довжина пізів:");

        jLabel3.setText("Кількість пізів:");

        jLabel4.setText("Номер карти розкрою:");

        jButton1.setText("Додати");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        jButton2.setText("<");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel1Layout = new
        javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
```



```

jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanell1Layout.createSequentialGroup()
        .addContainerGap()

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanell1Layout.createSequentialGroup()
        .addComponent(jButton2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton1,
javax.swing.GroupLayout.DEFAULT_SIZE, 116, Short.MAX_VALUE))
    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanell1Layout.createSequentialGroup()
        .addComponent(jLabel2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jTextField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 62, Short.MAX_VALUE))
    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanell1Layout.createSequentialGroup()
        .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jTextField2,
javax.swing.GroupLayout.DEFAULT_SIZE, 36,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanell1Layout.createSequentialGroup()
        .addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jTextField3,
javax.swing.GroupLayout.DEFAULT_SIZE, 85, Short.MAX_VALUE))
    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanell1Layout.createSequentialGroup()
        .addComponent(jLabel4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jTextField4,
javax.swing.GroupLayout.DEFAULT_SIZE, 44, Short.MAX_VALUE))
    .addContainerGap()
);
jPanell1Layout.setVerticalGroup(

jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanell1Layout.createSequentialGroup()
        .addContainerGap()

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel2)
    .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1)
    .addComponent(jTextField2,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(jTextField3,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel4)
        .addComponent(jTextField4,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jButton1)
        .addComponent(jButton2))
        .addContainerGap()
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGroup(layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
            );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGroup(layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
            );

    pack();
} // </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String name = jTextField1.getText();
    orderAgent.getAID().setName(name);
    double length = Double.parseDouble(jTextField2.getText());
    orderAgent.setLength(length);
    int number = Integer.parseInt(jTextField3.getText());
    orderAgent.setNumber(number);
    int mapNumber = Integer.parseInt(jTextField4.getText());
    orderAgent.setMapNumber(mapNumber);
    System.out.println("Agent by order #" + orderAgent.getAID().getName() +
" is ready");
    this.setVisible(false);
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    this.setVisible(false);
}

```

```

    }
    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JTextField jTextField1;
    private javax.swing.JTextField jTextField2;
    private javax.swing.JTextField jTextField3;
    private javax.swing.JTextField jTextField4;
    // End of variables declaration
}

```

Лістинг 2 – клас OrderAgent.java

```

package OrderAgent;

import GUI.OrderGUI;
import jade.core.Agent;

public class OrderAgent extends Agent {

    double cutsLength;
    int cutsNumber;
    int cutsMapNumber;
    String instruction;
    String note;

    @Override
    public void setup() {
        OrderGUI orderGUI = new OrderGUI(this);
        orderGUI.setVisible(true);
    }

    public void setLength(double length) {
        cutsLength = length;
    }

    public void setNumber(int number) {
        cutsNumber = number;
    }

    public void setMapNumber(int mapNumber) {
        cutsMapNumber = mapNumber;
    }

    public double getLength() {
        return cutsLength;
    }

    public int getNumber() {
        return cutsNumber;
    }

    public int getMapNumber() {
        return cutsMapNumber;
    }

    @Override
    protected void takeDown() {
        System.out.println("OrderAgent #" + getAID().getName() + "
terminating");
    }
}

```

Лістинг 3 – клас WorkSchedule.java

```
package ResourceAgent;

import OrderAgent.OrderAgent;
import java.util.ArrayList;

public class WorkSchedule {

    ArrayList<OrderAgent> schedule = new ArrayList<OrderAgent>();

    public void addOrder (OrderAgent order) {
        schedule.add(order);
    }

    public void addOrder (int i, OrderAgent order) {
        schedule.add(i, order);
    }

    public void setOrder (int i, OrderAgent order) {
        schedule.set(i, order);
    }

    public void removeOrder (int i) {
        schedule.remove(i);
    }

    OrderAgent getOrder (int i) {
        return schedule.get(i);
    }
}
```

Лістинг 4 – клас MenagementGUI.java

```
package GUI;

public class MenagementGUI extends javax.swing.JFrame {

    public MenagementGUI() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();
        jLabel1 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
    }
}
```

```

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(javax.swing.Border
rFactory.createLineBorder(new java.awt.Color(0, 0, 0)), " Графік виконання "));

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "№", "Номер замовлення", "Номер карти розкред", "Інструкція",
"Примітка"
    }
));
jScrollPane1.setViewportViewView(jTable1);

jLabel1.setText("Зараз виконується:");

jButton1.setText("Відмітити завершення");

jButton2.setText("<");

jButton3.setText("Змінити пріоритет");

jButton4.setText("Перенести замовлення на іншу ділянку");

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(10, 10, 10)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 415, Short.MAX_VALUE)
        .addGap(10, 10, 10)
    )
);

```

```

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
        .addComponent(jTextField1,
javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel1,
javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jButton1,
javax.swing.GroupLayout.DEFAULT_SIZE, 309, Short.MAX_VALUE))
        .addGroup(jPanell1Layout.createSequentialGroup())
        .addComponent(jButton2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jButton3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jButton4)))
        .addContainerGap())
);
jPanell1Layout.setVerticalGroup(

jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanell1Layout.createSequentialGroup()
        .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jButton1))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 91,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(jPanellLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BA
SELINE)

        .addComponent(jButton2)
        .addComponent(jButton3)
        .addComponent(jButton4))
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
            .addContainerGap()
            .addComponent(jPanell, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addContainerGap())
        );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
            .addContainerGap()
            .addComponent(jPanell, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );

    pack();
} // </editor-fold>

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {
            new MenagementGUI().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;

```

```

private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;

// End of variables declaration
}

```

Лістинг 5 – клас ResourceAgent.java

```

package ResourceAgent;

import OrderAgent.OrderAgent;
import jade.core.Agent;
import jade.core.behaviours.CyclicBehaviour;
import jade.core.behaviours.OneShotBehaviour;
import jade.domain.DFService;
import jade.domain.FIPAAgentManagement.DFAgentDescription;
import jade.domain.FIPAAgentManagement.ServiceDescription;
import jade.domain.FIPAException;
import jade.domain.introspection.ACLMessage;
import jade.lang.acl.MessageTemplate;

public class ResourceAgent extends Agent {

    WorkSchedule schedule;

    @Override
    protected void setup() {
        getAID().setName("ResourceAgent");
        System.out.println(getAID().getName() + " is ready");
        schedule = new WorkSchedule();
        DFAgentDescription dfd = new DFAgentDescription();
        dfd.setName(getAID());
        ServiceDescription sd = new ServiceDescription();
        sd.setType("CutOut");
        sd.setName("JADE-order-cutting");
        dfd.addServices(sd);
        try {
            DFService.register(this, dfd);
        } catch (FIPAException fe) {
            fe.printStackTrace();
        }
        addBehaviour(new OfferRequestsServer());
        addBehaviour(new PurchaseOrdersServer());
    }

    @Override
    protected void takeDown() {
        try {
            DFService.deregister(this);
        } catch (FIPAException fe) {
            fe.printStackTrace();
        }
        System.out.println(getAID().getName() + " terminating");
    }

    public void updateSchedule(final OrderAgent order) {
        addBehaviour(new OneShotBehaviour() {

```



```

        public void action() {
            schedule.addOrder(order);
            System.out.println("Order " + order.getName() + " inserted into
Work Schedule");
        }
    });
}

private class OfferRequestsServer extends CyclicBehaviour {

    public void action() {
        MessageTemplate mt =
MessageTemplate.MatchPerformative(ACLMessage.CFP);
        jade.lang.acl.ACLMessage msg = myAgent.receive(mt);
        if (msg != null) {
            String title = msg.getContent();
            jade.lang.acl.ACLMessage reply = msg.createReply();
            Integer price = (Integer) schedule.getOrder(0).getNumber();
            if (price != null) {
                reply.setPerformative(ACLMessage.PROPOSE);
                reply.setContent(String.valueOf(price.intValue()));
            } else {
                reply.setPerformative(ACLMessage.REFUSE);
                reply.setContent("not-available");
            }
            myAgent.send(reply);
        } else {
            block();
        }
    }
}

private class PurchaseOrdersServer extends CyclicBehaviour {

    public void action() {
        MessageTemplate mt =
MessageTemplate.MatchPerformative(ACLMessage.ACCEPT_PROPOSAL);
        jade.lang.acl.ACLMessage msg = myAgent.receive(mt);
        if (msg != null) {
            String title = msg.getContent();
            jade.lang.acl.ACLMessage reply = msg.createReply();
            Integer price = (Integer) schedule.remove(order);
            if (price != null) {
                reply.setPerformative(ACLMessage.INFORM);
                System.out.println(title + " sold to agent " +
msg.getSender().getName());
            } else {
                reply.setPerformative(ACLMessage.FAILURE);
                reply.setContent("not-available");
            }
            myAgent.send(reply);
        } else {
            block();
        }
    }
}
}

```