

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

до виконання лабораторних робіт
з дисципліни «Проектування інформаційних систем»
для здобувачів освітнього ступеня «бакалавр»
зі спеціальності 126 Інформаційні системи та технології
(освітньої програми «Web-технології, Web-дизайн»)
усіх форм навчання

Черкаси
2021

УДК 004.9
М54

*Затверджено вченою радою ФІТІС
протокол № 5 від 8.12.2020 р.
згідно з рішенням кафедри інформаційних
технологій проектування,
протокол № 4 від 13.11.2020 р.*

Упорядник: Оксамитна Л. П., *канд. техн. наук*

Рецензент: Мірошкіна І. В., *канд. техн. наук, доцент*

М54 Методичні рекомендації до виконання лабораторних робіт з дисципліни «Проектування інформаційних систем» для здобувачів освітнього ступеня «бакалавр» зі спеціальності 126 «Інформаційні системи та технології» (освітньої програми «Web-технології, Web-дизайн») усіх форм навчання [Електронний ресурс] / [упоряд. Оксамитна Л.П.]; М-во освіти і науки України, Черкас. держ. технол. ун-т. Черкаси: ЧДТУ, 2021. 99 с.

Методичні рекомендації спрямовані на формування у здобувачів освітнього ступеня «бакалавр» за спеціальністю 126 «Інформаційні системи та технології» (освітньої програми «Web-технології, Web-дизайн») вмінь аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов та застосовувати CASE-технології створення й супроводу інформаційних систем.

УДК 004.9

Виробничо-практичне
електронне видання
комбінованого використання

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
до виконання лабораторних робіт
з дисципліни «Проектування інформаційних систем»
для здобувачів освітнього ступеня «бакалавр»
зі спеціальності 126 Інформаційні системи та технології
(освітньої програми «Web-технології, Web-дизайн»)
усіх форм навчання

Упорядник: **Оксамитна Любов Павлівна**
В авторській редакції.

ВСТУП

Проектування інформаційних систем (ІС) важливий елемент у формуванні високоєфективного підприємства.

Проектування ІС завжди пов'язане із метою проекту. У реальних умовах проектування – це пошук способу, який задовольняє вимогам функціональності системи засобами наявних технологій з урахуванням заданих обмежень. Воно необхідне для вибудовування процесів підприємства відповідно до вимог ринку; включає стадії проектування, а також аналізу і оптимізації процесів як частина безперервного циклу управління процесами. За таких умов важливого значення набуває підготовка висококваліфікованих фахівців, які б могли не тільки кваліфіковано вирішувати питання вибору ІС, а й організувати процеси проектування ІС під вимоги конкретних бізнес-процесів.

Метою викладання навчальної дисципліни «Проектування інформаційних систем» є забезпечення базової профілюючої підготовки за фахом, формування теоретичних знань та практичних навичок у галузі проектування інформаційних систем (ІС) з використанням сучасних методів та засобів створення інформаційних систем під час їх розробки, налагодження та експлуатації.

Основними завданнями вивчення дисципліни «Проектування інформаційних систем» є отримання студентами теоретичних знань і практичних навичок вивчення побудови і функціонування інформаційних технологій та інформаційних систем на підприємствах, нормативної бази, сучасних підходів до їх проектування і впровадження, а також оволодіння практичними навичками роботи в середовищі готових програмних продуктів та технологіями їх застосування до вирішення проблем та практичних задач сьогодення.

Дані лабораторні роботи спрямовані на вивчення CASE-засобу структурно-функціонального моделювання AllFusion Process Modeler, а саме: на вивчення та отримання практичних навичок студентами за стандартами IDEF0, IDEF3, DFD, які будуть використовуватися ними при виконанні завдань проектування інформаційних систем та їх програмного забезпечення.

Метою лабораторних робіт є набуття здобувачами вищої освіти практичних навичок щодо проектування комплексу типових завдань з застосуванням методології структурно-функціонального моделювання SADT та стандартів IDEF для окремих модулів автоматизованої інформаційної системи.

ЛАБОРАТОРНА РОБОТА №1

Вивчення програми AllFusion Process Modeler, інструментів моделювання та основних фігур

Мета роботи: Набути практичних навичок застосування інструментів засобу AllFusion Process Modeler для побудови контекстної діаграми бізнес-процесу.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Загальна характеристика CASE-засобу AllFusion Process Modeler

CASE-засіб – будь-який програмний засіб, що автоматизує сукупність процесів життєвого циклу програмного забезпечення і володіє наступними особливостями:

- наявністю потужних графічних засобів для опису і документування інформаційних систем, що забезпечує зручний інтерфейс;
- можливістю інтеграції окремих компонентів, забезпечує керованість процесом розробки інформаційних систем;
- можливістю використання спеціальним чином організованого сховища проектних метаданих (сховища).

Приклади CASE-засобів: BPwin (AllFusion Process Modeler), ERwin (AllFusion ERwin Data Modeler), Rational Rose, Object Team та ін.

AllFusion Process Modeler (раніше BPwin) – інструмент, призначений для моделювання, аналізу, документування та оптимізації бізнес-процесів. AllFusion Process Modeler можна використовувати для графічного представлення бізнес-процесів. Графічно представлена схема виконання робіт, обміну інформацією, документообігу візуалізує модель бізнес-процесу. Графічне виклад цієї інформації дозволяє перевести завдання управління організацією з області складного ремесла в сферу інженерних технологій.

AllFusion Process Modeler допомагає чітко документувати важливі аспекти будь-яких бізнес-процесів: дії, які необхідно виконати, способи їх здійснення і контролю, потрібні для цього ресурси, а також візуалізувати одержувані від цих дій результати. AllFusion Process Modeler підвищує бізнес-ефективність IT-рішень, дозволяючи аналітикам і проектувальникам моделей співвідносити корпоративні ініціативи та завдання з бізнес-вимогами і процесами

інформаційної архітектури та проектування додатків. Таким чином, формується цілісна картина діяльності підприємства: від потоків робіт в невеликих підрозділах до складних організаційних функцій.

AllFusion Process Modeler ефективний в проектах, пов'язаних з описом діючих баз даних підприємств, реорганізацією бізнес-процесів, впровадженням корпоративної інформаційної системи. Інструмент дозволяє оптимізувати діяльність підприємства і перевірити її на відповідність стандартам якості ISO 9000, спроектувати оргструктуру, знизити витрати, виключити непотрібні операції і підвищити ефективність. В основу продукту закладені загальновизнані методології моделювання, наприклад, методологія IDEF0. Простота і наочність моделей *Process Modeler* спрощує взаєморозуміння між усіма учасниками процесів.

AllFusion Process Modeler (раніше *BPwin*) – це CASE-засіб для моделювання бізнес-процесів, що дозволяє створювати діаграми в нотації IDEF0, IDEF3, DFD. В процесі моделювання. Засіб дозволяє переключитися з нотації IDEF0 на будь-які гілки моделі на нотацію IDEF3 або DFD і створити змішану модель і підтримує функціонально-вартісний аналіз (ABC).

Поширеність самого *AllFusion Process Modeler* дозволяє вести узгодження функціональних моделей з партнерами в електронному вигляді. Продукт *AllFusion Process Modeler* створений компанією *Computer Associates*. *AllFusion Process Modeler* поряд з *ERwin Data Modeler* (раніше: *ERwin*), *Data Model Validator* (раніше: *ERwin Examiner*), *Model Manager* (раніше: *ModelMart*) входить до складу пакету програмних засобів *AllFusion Modeling Suite*, комплексне використання якого забезпечує всі аспекти функціональної методики моделювання інформаційних систем.

Основні можливості AllFusion Process Modeler 7:

- Підтримка різних технологій моделювання.
- Аналіз показників витрат і продуктивності.
- Інтеграція процесів / даних.
- Підтримка стандартних нотацій.
- Експорт об'єктів і властивостей в інші моделі.
- Документування інформації в межах всієї моделі.
- Масштабованість звітності без втрати якості графіків.

Переваги AllFusion Process Modeler 7:

- підтримує відразу три стандартні нотації – IDEF0 (функціональне моделювання), DFD (моделювання потоків даних) і IDEF3 (моделювання потоків робіт). Ці три основні ракурси дозволяють описувати предметну область більш комплексно;
- дозволяє підвищити ефективність бізнесу, оптимізувати будь-які процедури в компанії;

- повністю підтримує методи розрахунку собівартості за обсягом господарської діяльності (функціонально-вартісний аналіз ФСА / ABC).
- недорогий, поширений, по ньому багато інформації і компетентних фахівців;
- легкий в освоєнні і застосуванні, є курси російською мовою;
- дозволяє полегшити сертифікацію на відповідність стандартам якості ISO9000;
- є стандартом де-факто, інтегрований з ERwin Data Modeler (для моделювання баз даних)
- завдяки вищезгаданій інтеграції та підтримки спільної, командної роботи над одними і тими ж моделями (за допомогою ModelManager), не має аналогів для великих проектів.
- інтегрований із засобом імітаційного моделювання Arena.
- містить власний генератор звітів.
- дозволяє ефективно маніпулювати моделями – зливати і розщеплювати їх.
- має широкий набір засобів документування моделей, проектів.

В даний час популярністю користується інтегральний пакет інструментальних засобів AllFusion Modeling Suite, що підтримує всі етапи розробки інформаційних систем і складається з наступних програмних продуктів:

1. AllFusion Process Modeler – нова назва програми BPwin, використовуваної для моделювання процесів.
2. AllFusion ERwin Data Modeler – нова назва програми ERwin, використовуваної для моделювання даних.
3. AllFusion Data Model Validator – нова назва програми ERwin Examiner, використовуваної для пошуку і виправлення помилок моделі даних.
4. AllFusion Model Manager – нова назва програми ModelMart, використовуваної як сховище моделей BPwin і ERwin для організації колективної роботи.
5. AllFusion Component Modeler – нова назва програми Paradigm Plus, яка використовується як інструмент створення об'єктних моделей.

AllFusion Process Modeler 7 має достатньо простий та інтуїтивно зрозумілий інтерфейс користувача, що дає можливість аналітикові створювати складні моделі при мінімальних зусиллях.

2. Принципи побудови моделі IDEF0

Методологія IDEF0 базується на підході, який був запропонований Дугласом Т. Россом ще на початку 70-х років минулого століття і знаний у всьому світі як метод структурного аналізу і проектування, англomовна назва – SADT (Structured Analysis & Design Technique). Процес моделювання системи за методологією IDEF0 починається з визначення контексту, тобто найбільш абстрактного рівня опису системи в цілому. У контекст входить визначення суб'єкта моделювання, цілі та точки зору на модель.

Під *суб'єктом* розуміємо саму систему, при цьому необхідно точно встановити, що входить у систему, а що лежить за її межами. На визначення суб'єкта системи істотно впливатиме точка зору, з якою розглядається система, і мета моделювання – питання, на які побудована модель повинна дати відповідь.

Точка зору – вказівка на посадову особу або підрозділ організації, з позиції якого розробляється модель.

Мета моделювання (Purpose) має бути визначена передусім і полягає у формуванні відповідей на такі запитання:

- Чому цей процес повинен бути змодельований?
- Що повинна показувати модель?
- Що може отримати користувач?

3. Синтаксис графічної мови IDEF0

Модель IDEF0 – графічний опис системи, розроблений з певною метою і з вибраної точки зору. Опис системи за допомогою графічної мови IDEF0 називається *функціональною моделлю*.

Основу методології IDEF0 складає графічна мова опису бізнес-процесів. Модель в нотації IDEF0 є сукупністю ієрархічно впорядкованих і взаємозв'язаних діаграм. Кожна діаграма є одиницею опису системи і розташовується на окремому листі. Модель може містити чотири типи діаграм:

- контекстну діаграму (у кожній моделі може бути тільки одна контекстна діаграма);
- діаграми декомпозиції;
- діаграми дерева вузлів;
- діаграми тільки для експозиції (FEO).

Контекстна діаграма є вершиною деревовидної структури діаграм і є найзагальнішим описом системи і її взаємодії із зовнішнім середовищем. Після опису системи в цілому проводиться розбиття її на крупні фрагменти. Цей процес називається функціональною декомпозицією, а діаграми, які описують кожен фрагмент і взаємодію фрагментів, називаються діаграмами декомпозиції.

Методологія IDEF0 передбачає побудову ієрархічної системи діаграм – одиничних описів фрагментів системи. Спочатку проводиться опис системи в цілому і її взаємодії з навколишнім світом (контекстна діаграма), після чого проводиться функціональна декомпозиція – система розбивається на підсистеми і кожна підсистема описується окремо (діаграми декомпозиції). Потім кожна підсистема розбивається на дрібніші і так далі до досягнення потрібного ступеня деталізації.

Кожна IDEF0-діаграма містить блоки і дуги. Блоки зображують роботи модельованої системи. Дуги зв'язують роботи разом і відображають взаємодії та взаємозв'язки між ними.

Функціональні блоки або Роботи (Activity) на діаграмах позначаються прямокутниками, що означають поійменовані процеси, функції або завдання, які відбуваються протягом певного часу і мають певні результати. Блок описує роботу. У середині кожного блоку міститься його ім'я і номер. Ім'я повинне бути

активним дієсловом або дієслівним оборотом, що відображає зміст роботи. Номер блоку розміщується у правому нижньому куті. Номери блоків використовуються для їх ідентифікації на діаграмі і у відповідному тексті.

Контекстна діаграма – це діаграма, що має вузловий номер A-n ($n \geq 0$) і являє собою контекст моделі. Діаграма A-0, що складається з одного блоку, є необхідною (обов'язковою) контекстною діаграмою; діаграми з вузловими номерами A-1, A-2... – додаткові контекстні діаграми.

Взаємодія робіт із зовнішнім світом і між собою описується за допомогою стрілок, що зображуються одинарними лініями із стрілками на кінцях. Стрілки іменуються іменниками.

Номер блоку – це число (0–8), яке розміщене в правому нижньому куті блоку й однозначно ідентифікує блок на діаграмі.

Після присвоєння імені роботі до відповідних його сторін приєднуються вхідні, вихідні й керуючі стрілки, а також стрілки механізмів, що й визначає наочність і виразність зображення блоку IDEF0.

Кожна сторона функціонального блоку має стандартне значення з погляду зв'язку блок/стрілка. У свою чергу, сторона блоку, до якої приєднана стрілка, однозначно визначає її роль і тип.

У IDEF0 розрізняють п'ять типів стрілок (рис. 1.1).

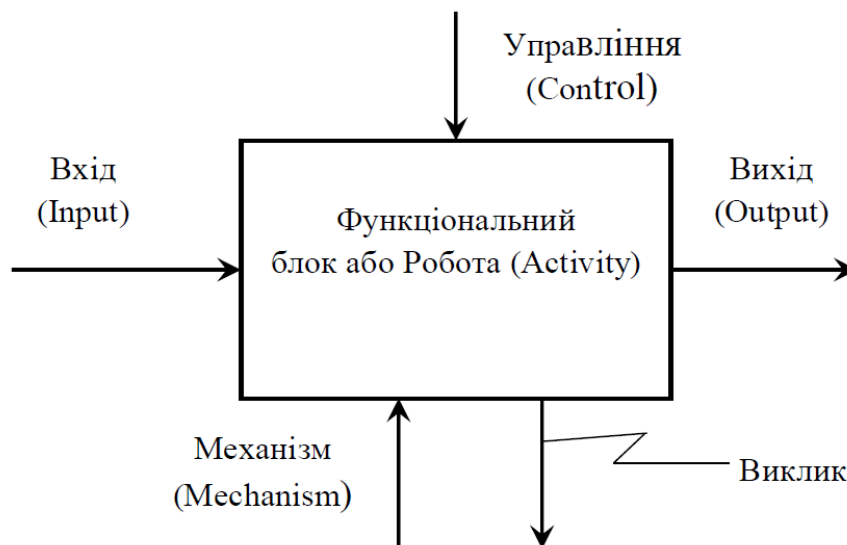


Рисунок 1.1 – Схема взаємодії роботи і дуг

Вхід (Input) – об'єкти, що використовуються і перетворюються роботою для отримання результату (виходу). Допускається, що робота може не мати жодної стрілки входу. Стрілка входу малюється як така, що входить у ліву грань роботи.

Управління (Control) – інформація, що керує діями роботи. Стрілки, що керують, несуть інформацію, яка вказує, що і як повинна виконувати робота. Кожна робота повинна мати хоча б одну стрілку управління, яка зображується як така, що входить у верхню грань роботи.

Вихід (Output) – об'єкти, в які перетворюються входи. Кожна робота повинна мати хоча б одну стрілку виходу, яка малюється як витікаюча з правої грані роботи.

Механізми (Mechanism) – ресурси, що виконують роботу. Стрілка механізму входить у нижню грань роботи. Аналітик іноді може прийняти рішення стрілки механізму не відобразити у моделі.

Виклик (Call) – спеціальна стрілка, що вказує на іншу модель роботи. Стрілка виклику малюється як витікаюча з нижньої частини роботи і використовується для вказівки того, що деяка частина роботи виконується за межами модельованої системи.

Код ICOM – аббревіатура (Input – Вхід, Control – Управління, Output – Вихід, Mechanism – Механізм), код, що забезпечує відповідність граничних стрілок дочірньої діаграми із стрілками батьківського блоку; використовується для посилянь.

Стрілки на контекстній діаграмі служать для опису взаємодії системи з навколишнім світом. Вони можуть починатися біля межі діаграми і закінчуватися у роботи, або навпаки. Такі стрілки називаються *граничними*.

Гранична стрілка – стрілка, один з кінців якої пов'язаний з джерелом або споживачем, а інший не приєднаний до жодного блоку на діаграмі. Відображає зв'язок діаграми з іншими блоками системи і відрізняється від внутрішньої стрілки.

Для внесення граничної стрілки входу потрібно:

- 1) клацнути по кнопці з символом стрілки в палітрі інструментів: перенести курсор до лівої сторони екрану, поки не з'явиться початкова штрихова смужка;
- 2) клацнути один раз по смужці (звідки виходить стрілка) і ще раз в лівій частині роботи з боку входу (де закінчується стрілка);
- 3) повернутися в палітру інструментів і вибрати опцію редагування стрілки;
- 4) клацнути правою кнопкою миші на лінії стрілки, в спливаючому меню вибрати Name Editor і додати ім'я стрілки в закладці Name діалогу IDEF0 Arrow Properties.

Стрілки управління, виходу, механізму і виходу зображуються аналогічно.

4. Середовище AllFusion Process Modeler

AllFusion Process Modeler (AFPM) підтримує три методології – IDEF0, IDEF3 і DFD, кожна з яких вирішує свої специфічні завдання. У AFPM можлива побудова змішаних моделей, тобто модель може містити одночасно як діаграми IDEF0, так і IDEF3 і DFD. Склад панелі інструментів змінюється автоматично, коли відбувається перемикання з однієї нотації на іншу.

Робоче вікно програми AllFusion Process Modeler містить такі області:

- панель інструментів;
- головне меню;
- браузер моделі (*Model Explorer*) має три вкладки – *Activities* (), *Diagrams* і *Objects* ();
- бланк діаграми;
- рядок стану.

На рис. 1.2 представлено робоче вікно середовища AllFusion Process Modeler.

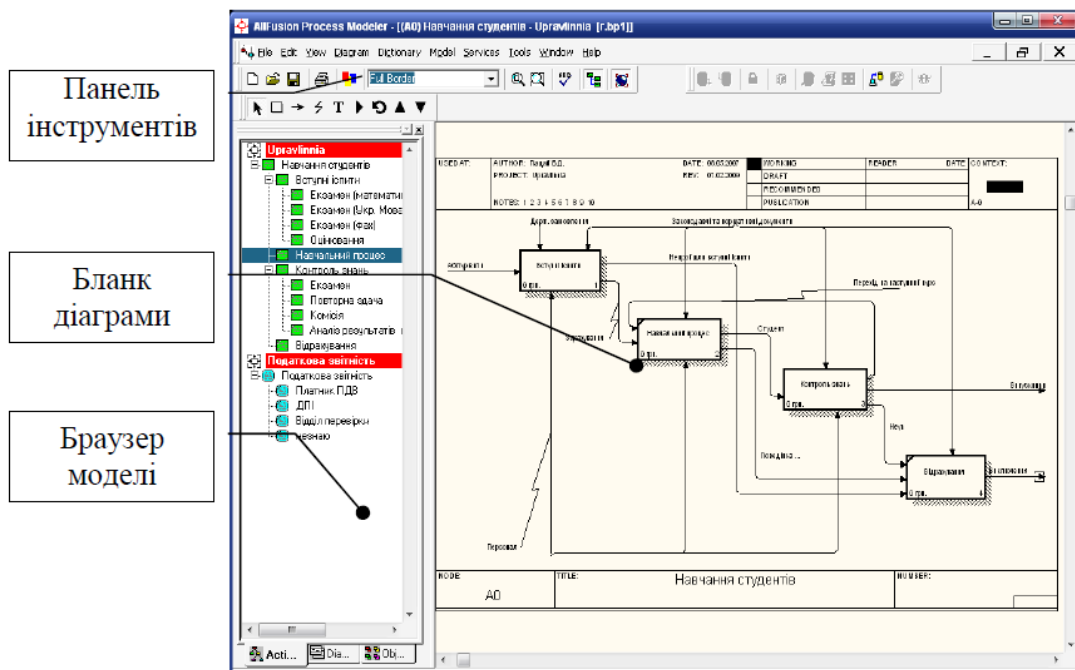


Рисунок 1.2 – Середовище AllFusion Process Modeler

5. Панель інструментів

При запуску AllFusion Process Modeler за умовчанням з'являється основна панель інструментів, палітра інструментів (вид якої залежить від вибраної нотації). На стандартній панелі інструментів розташовані елементи управління, знайомі з інших Windows-програм (рис. 1.3):



Рисунок 1.3 – Панель інструментів стандартна

5.1 Опис елементів управління основної панелі інструментів

1. Створити нову модель – *File/New*;
2. Відкрити модель – *File/Open*;
3. Зберегти модель – *File/Save*;
4. Надрукувати модель – *File/Print*;
5. Майстер створення звітів – *File*;
6. Вибір масштабу – *View/Zoom*;
7. Масштабування – *View/Zoom*;
8. Збільшення ділянки – *View/Zoom*;
9. Перевірка правопису – *Tools/Spelling*;
10. Включення і виключення навігатора моделі – *View/Model Explorer*;
11. Включення і виключення додаткової панелі інструментів роботи з *ModelManage* – *ModelManage*.

Панель інструментів редактора AFPM (рис.1.4) містить:



Рисунок 1.4 – Панель інструментів AFPM

1. *Pointer Tool* – використовується для вибору і визначення позиції об'єктів доданих в діаграму;
2. *Activity Box Tool* – використовується для додавання робіт у діаграму;
3. *Arrow Tool* – використовується, щоб додавати стрілки (дуги) на діаграмі;
4. *Squiggle Tool* – використовується для створення тильди (squiggle), яка сполучає стрілку з її назвою;
5. *Text Block Tool* – використовується для створення текстових блоків;
6. *Diagram Dictionary Editor* – відкриває діалогове вікно Diagram Dictionary Editor, де можна перейти на яку-небудь діаграму або створити нову діаграму;
7. *Go to Sibling Diagram* – використовується для відображення наступної діаграми того ж рівня;
8. *Go to Parent Diagram* – перехід на батьківську діаграму;
9. *Go to Child Diagram* – використовується, щоб відобразити дочірню діаграму або декомпозувати дану роботу.

5.2 Опис полів бланку діаграми

Будь-яка діаграма складається з сукупності таких об'єктів:

- робіт;
- дуг (стрілок);
- текстових блоків.

Для роботи з будь-яким з цих об'єктів можна використовувати головне меню або контекстне. Кожна діаграма розташовується усередині бланку, що має декілька інформаційних полів.

5.3 Поля верхньої частини рамки

1. *Used At (Використовується в)* – використовується для вказівки на батьківський блок у випадку, якщо на поточну діаграму посилалися за допомогою стрілки виклику.
2. *Author (Автор)* – ім'я автора діаграми.
3. *Date (Дата)* – дата створення проекту.
4. *Project (Проект)* – ім'я проекту.
5. *Rev (Переглянуто)* – дата останнього редагування діаграми;
6. *Notes 12345678910 (Зауваження)* – використовується при проведенні сеансу експертизи. Експерт повинен (на паперовій копії діаграми) вказати

число зауважень, викреслюючи цифру із списку кожного разу при внесенні нового зауваження.

7. *Status (Статус)* – статус відображає стадію створення діаграми, всі етапи публікації:
 - *Working (Робоча версія)* – нова діаграма, кардинально оновлена діаграма або новий автор діаграми;
 - *Draft (Ескіз)* – діаграма пройшла первинну експертизу і готова до подальшого обговорення;
 - *Recommended (Рекомендовано)* – діаграма і всі її супроводжуючі документи пройшли експертизу, нових змін не очікується;
 - *Publication (Публікація)* – діаграма готова до остаточного друку і публікації;
 - *Reader (Читач)* – ім'я читача (експерта);
 - *Date (Дата)* – дата прочитання (експертизи).
8. *Context (Контекст)* – схема розташування робіт у діаграмі верхнього рівня. Робота, що є батьківською, зображується темним прямокутником, інші роботи – світлим прямокутником. На контекстній діаграмі (A-0) показується напис TOP. У лівому нижньому куті – номер по вузлу батьківської діаграми.

5.4 Поля нижньої частини рамки

1. *Node (Вузол)* – номер вузла діаграми (номер батьківського блоку).
2. *Title (Назва)* – ім'я діаграми; за замовчуванням – ім'я батьківського блоку.
3. *Number (Номер)* – С-номер, унікальний номер версії діаграми.
4. *Page (Сторінка)* – номер сторінки, може використовуватися як номер сторінки при формуванні теки.
5. У стартовому вікні (рис. 1.5) користувачу пропонується вибір нотації майбутньої моделі.

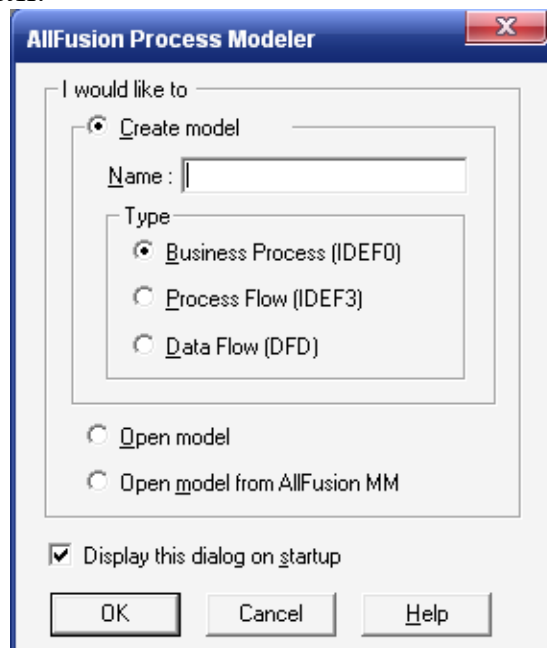


Рисунок 1.5 – Стартове вікно створення моделі

Модель у AFPM розглядається як сукупність робіт, кожна з яких оперує з деяким набором даних. Робота зображується у вигляді прямокутників, дані – у вигляді стрілок. Якщо клацнути по будь-якому об'єкту моделі лівою кнопкою миші, з'являється спливаюче контекстне меню даного об'єкта.

6. Встановлення кольору і шрифту об'єктів

Пункти контекстного меню *Font Editor* і *Color Editor* викликають відповідні діалогові вікна для установки шрифту (зокрема його розміру і стилю) та кольору об'єкта. Крім того, AFPM дозволяє встановити шрифт за умовчанням для об'єктів певного типу на діаграмах і у звітах.

Для цього слід вибрати меню *Tools/Default Fonts*, після чого з'являється каскадне меню, кожен пункт якого служить для встановлення шрифтів для певного типу об'єктів:

- Context Arrow* – стрілки на контекстній діаграмі;
- Decomposition Activity* – роботи на діаграмі декомпозиції;
- Decomposition Arrow* – стрілки на діаграмі декомпозиції;
- NodeTree Text* – текст на діаграмі дерева вузлів;
- Frame User Text* – текст, що вноситься користувачем у каркасі діаграм;
- Frame System Text* – системний текст у каркасі діаграм;
- Text Blocks* – текстові блоки;
- Parent Diagram Text* – текст батьківської діаграми;
- Parent Diagram Title Text* – текст заголовка батьківської діаграми;
- Report Text* – текст звітів.

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler з використанням стандарту функціонального моделювання IDEF0.


Завдання 1. Розглянути приклад 1.

Завдання 2. Побудувати контекстну діаграму бізнес-процесу згідно варіантів завдань (табл. 1.1).

Таблиця 1.1 – Предметна область для виконання завдання

Номер варіанта	Назва предметної області
1	Брокерський продаж квартир.
2	Обмін валют у фінансовій установі.
3	Нарахування стипендії студентам.
4	Проходження практики в ІТ-фірмах.
5	Діяльність ювелірної крамниці.
6	Поселення студентів у гуртожиток.
7	Вступ до магістратури.
8	Діяльність туристичної фірми (або готелю).
9	Резервування і доставка замовлень у ресторані.
10	Купівля квартири.

Приклад 1. Побудова контекстної діаграми бізнес-процесу на прикладі діяльності ТОВ «Буєрак» по виготовленню тютюнових виробів.

1. Запустимо програму AllFusion Process Modeler (Кнопка Start / AllFusion Process Modeler r7). Натиснемо кнопку . З'являється діалогове вікно *I would like to* (рис. 1.6). Введемо у текстове поле *Name* ім'я моделі «Виготовлення тютюнових виробів» та за допомогою альтернативного перемикача виберемо *Type – Business Process (IDEF0)*. Відкриється діалогове вікно *Properties for New Models* (рис. 1.7). У текстове поле *Author* (Автор) необхідно ввести своє П.І.П. – автора моделі і в текстове поле *Author initials* назву групи. Автоматично створюється незаповнена контекстна діаграма.

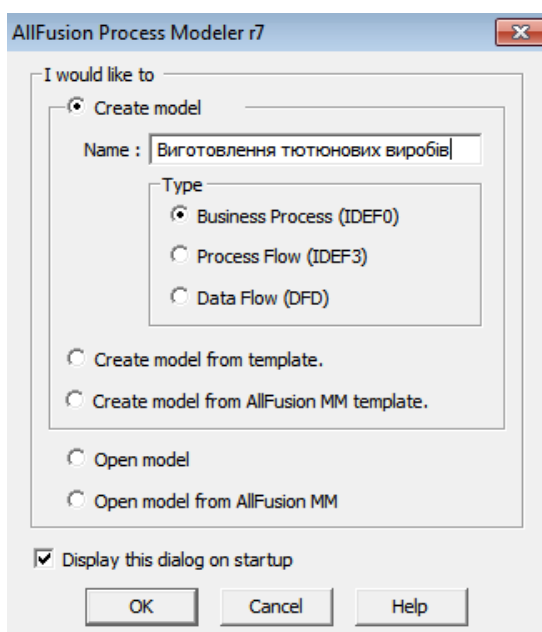


Рисунок 1.6 – Надання імені моделі та вибір нотації

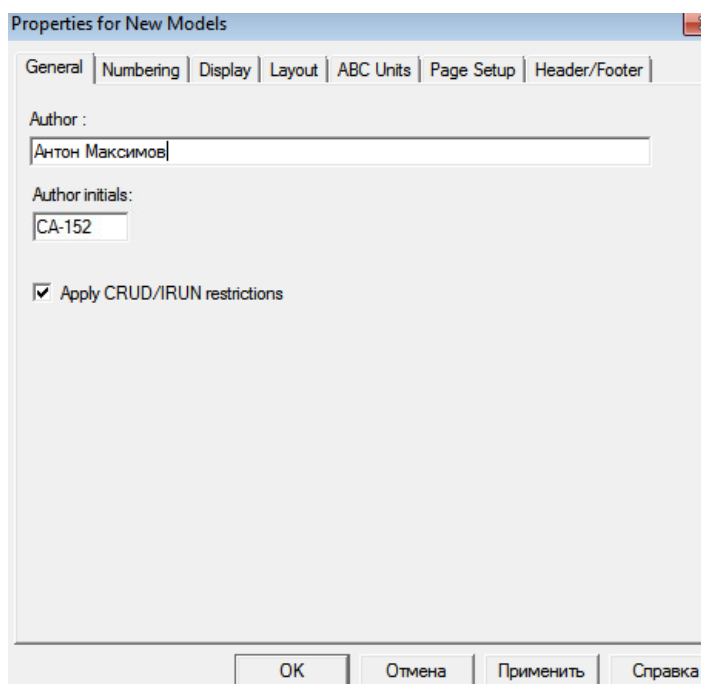


Рисунок 1.7 – Введення імені автора моделі та його ініціалів

2. На вкладці *Activities* браузера моделі вибираємо опції редагування властивостей моделі (рис. 1.8).

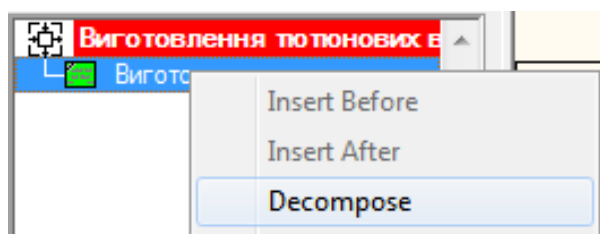


Рисунок 1.8 – Зміна властивостей моделі

3. Переходимо у меню *Model/Model Properties*. На вкладці *General* діалогового вікна *Model Properties* у текстове поле *Model name* вводимо ім'я моделі «Виготовлення тютюнових виробів», а в текстове поле *Project* ім'я

проекту «Модель роботи тютюнового заводу» та у текстове поле *Time Frame* (Часове охоплення) – AS-IS (Як є) (рисунок 1.9).

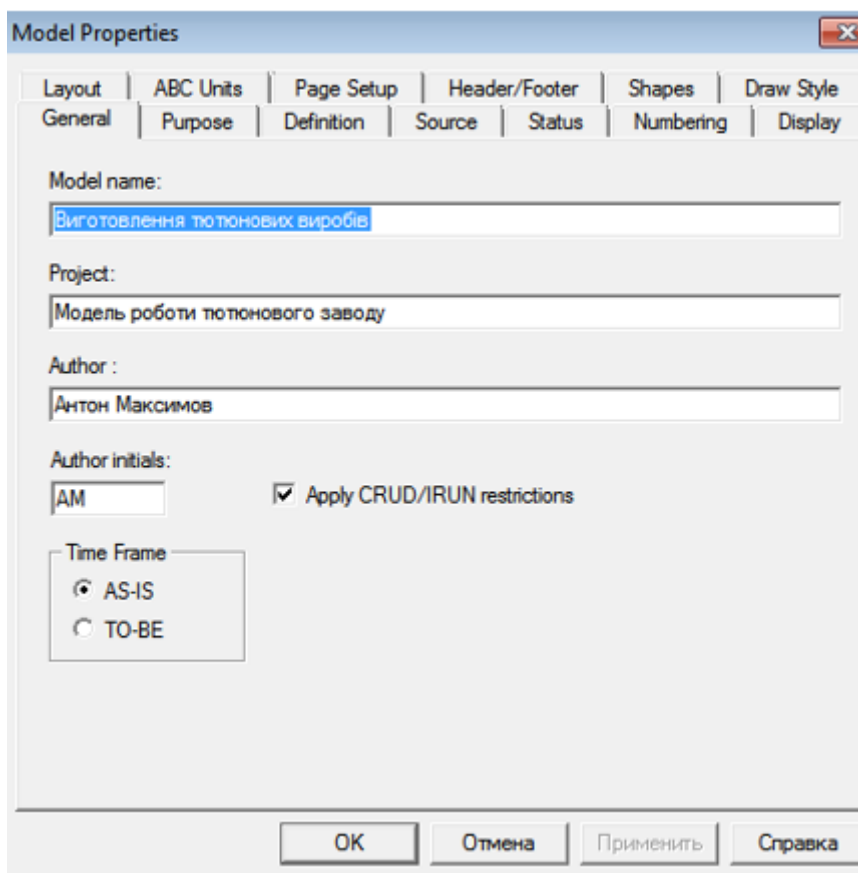


Рисунок 1.9 – Вікно присвоєння властивостей моделі

4. На вкладці *Purpose* діалогового вікна *Model Properties* у текстове поле *Purpose* (мета) вносимо дані про мету розробки моделі – «Змодельовати поточні бізнес-процеси ТОВ «Буєрак», а в текстове поле *Viewpoint* (точка зору) – «Керівник» (рис. 1.10).

5. На вкладці *Definition* діалогового вікна *Model Properties* у текстове поле *Definition* (Визначення) вносимо «Це навчальна модель, що описує діяльність виготовлення тютюнових виробів на ТОВ «Буєрак»» і в текстове поле *Scope* (охоплення) – «Загальне керівництво бізнесом: дослідження ринку, закупка сировини, формування партії тютюнових виробів» (рисунок 1.11).

6. Переходимо на контекстну діаграму і правою кнопкою миші клацнемо на прямокутнику, що являє собою в нотації IDEF0 умовне графічне позначення роботи. У контекстному меню виберемо опцію *Name* (рисунок 1.12). На вкладці *Name* вносимо ім'я «Виготовлення тютюнових виробів» (рисунок 1.13).

7. На вкладці *Definition* діалогового вікна *Activity Properties* у текстове поле *Definition* (Визначення) вносимо «Існуючі бізнес-процеси компанії» (рис. 1.14). Текстове поле *Note* (Примітки) залишимо незаповненим.

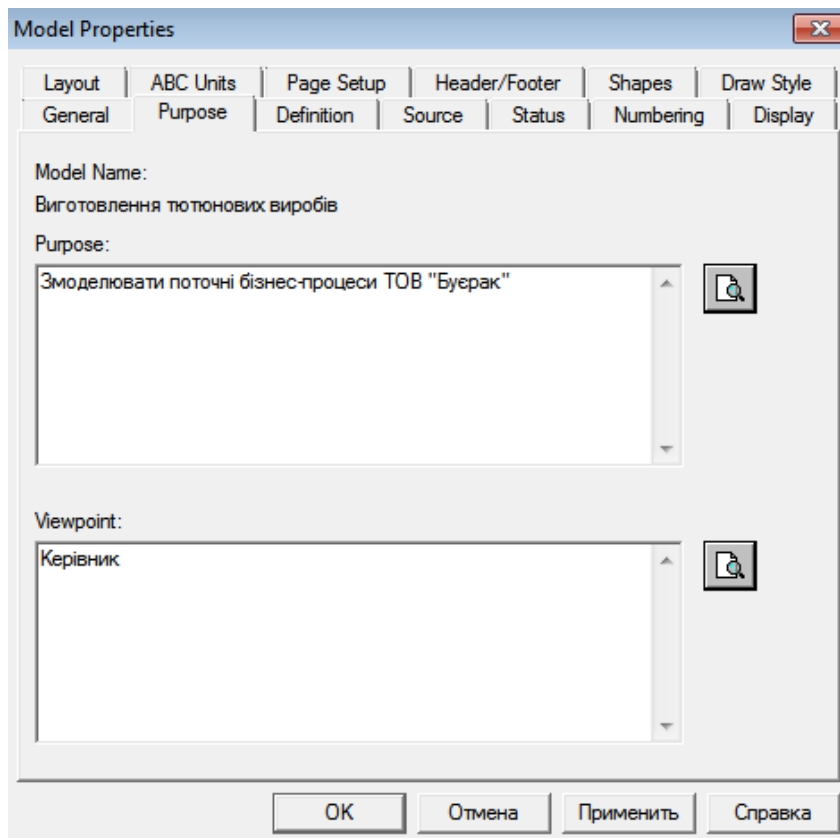


Рисунок 1.10 – Внесення даних про мету моделювання і точку зору на модель

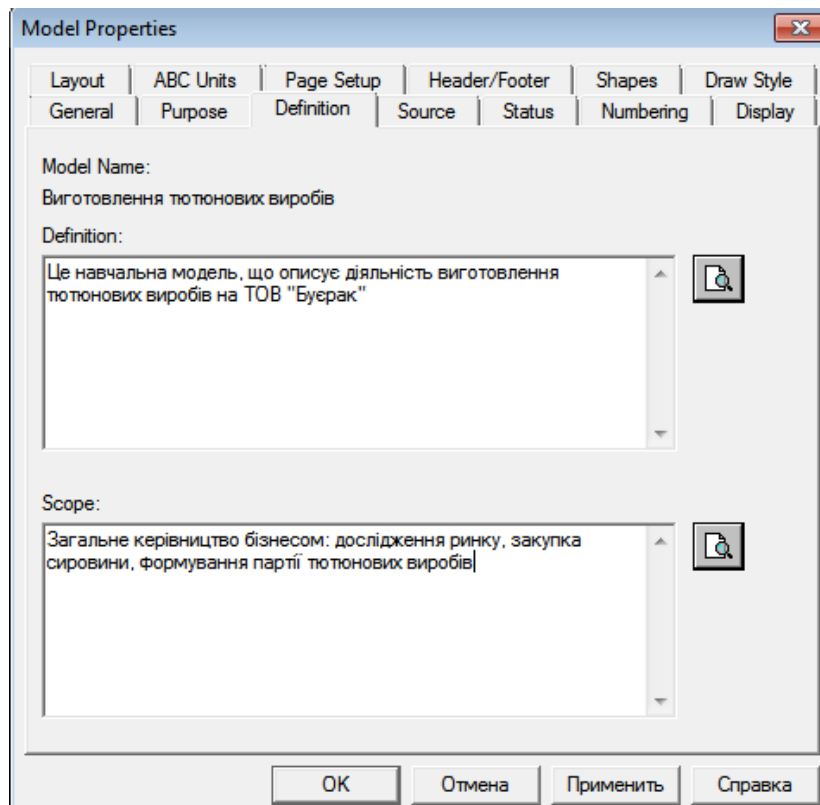


Рисунок 1.11 – Внесення додаткових даних тих, що визначають модель

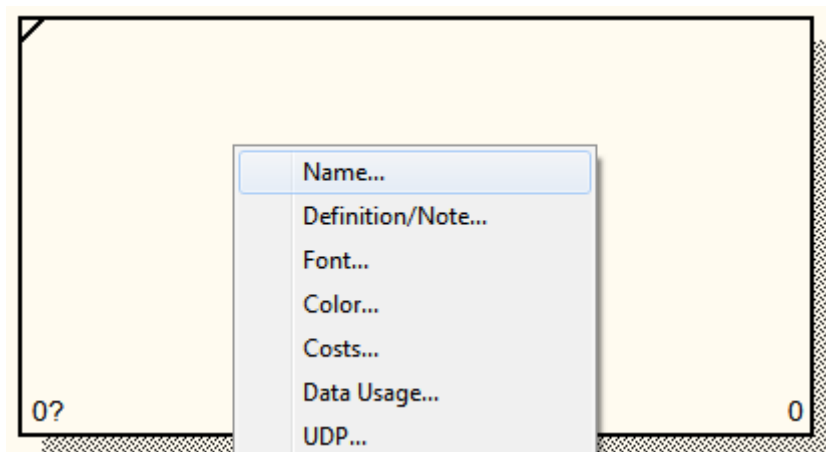


Рисунок 1.12 – Контекстне меню для роботи з вибраною опцією *Name*

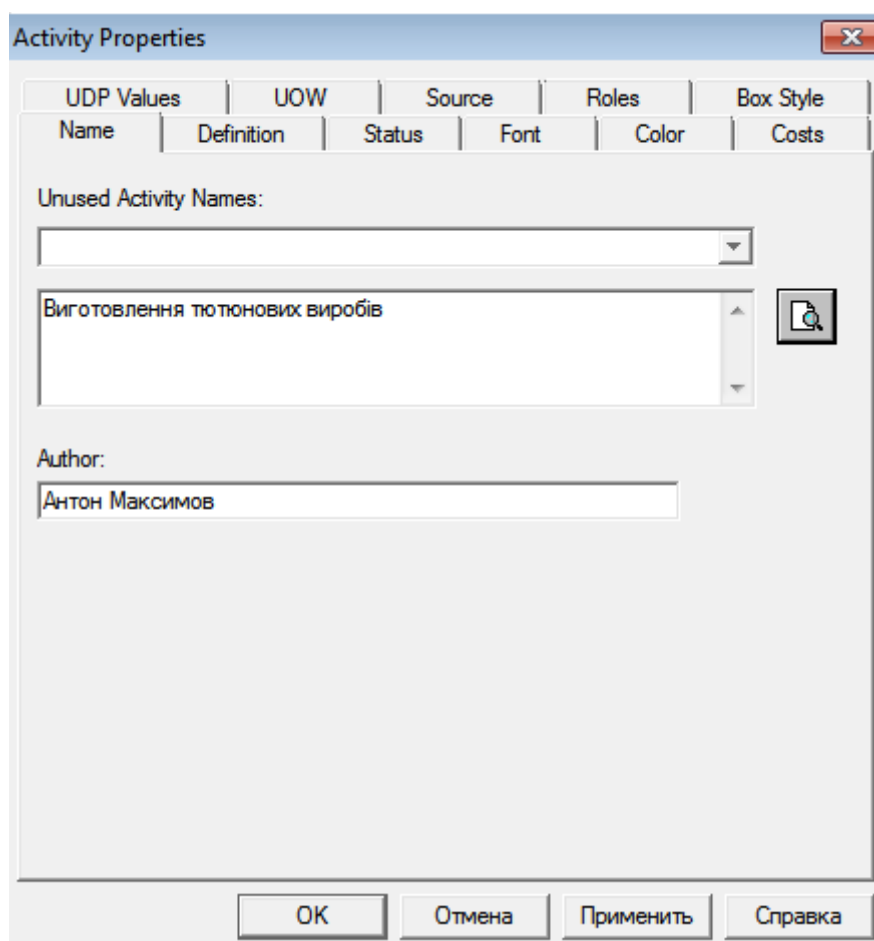


Рисунок 1.13 – Привласнення роботі назви

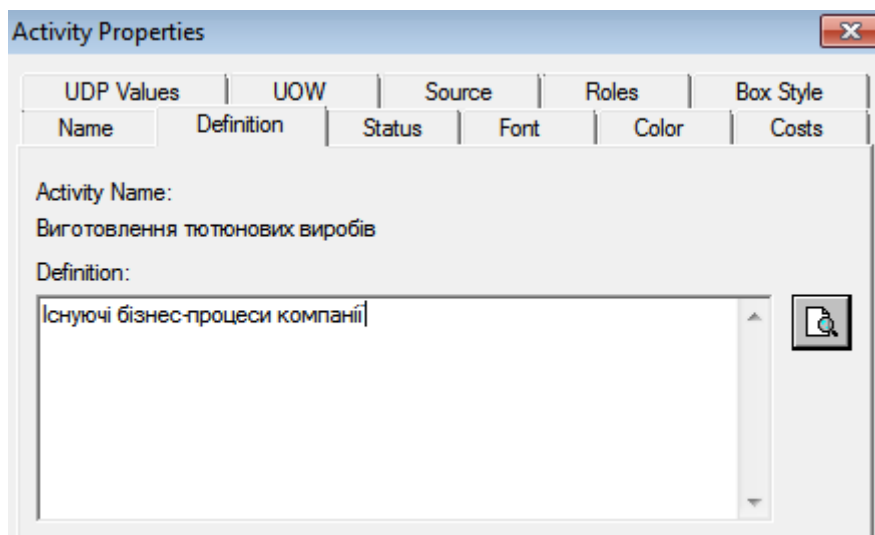


Рисунок 1.14 – Внесення додаткових даних про роботу

8. Створюємо ICOM-стрілки на контекстній діаграмі (таблиця 1.2).

Таблиця 1.2 – Стрілки контекстної діаграми

Назва стрілки (Arrow Name)	Визначення стрілки (Arrow Definition)	Тип стрілки (Arrow Type)
Цигарковий папір	Запити інформації у будь-якій формі, замовлення, технічна підтримка, тощо.	Input
Тютюнова суміш	Головний компонент для виготовлення тютюнових виробів.	Input
Добавки	Допоміжний компонент для виготовлення тютюнових виробів.	Input
Цигарковий фільтр	Оформлення рахунків, оплата рахунків, робота із замовленнями.	Input
Целофан	Матеріал для захисту готової пачки цигарок.	Input
Картон	Матеріал для виготовлення корпусу пачки цигарок.	Input
Клей	Матеріал для склеювання складових частин пачки цигарок.	Input
Етикетки	Матеріал для зазначенні бренду на пачці та створення акцизних марок.	Input
ДЕСТ 30038-93	Правила по роботі з сировиною для виготовлення тютюнових виробів.	Control
ДЕСТ 30422-96	Правила по роботі з формування та пакування тютюнових виробів.	Control
Купажист	Працівник компанії, який формує партії сировини.	Mechanism

Працівники компанії	Персонал компанії, який забезпечує правильну роботу автоматизованих процесів.	Mechanism
Конвеєр для переробки сировини	Машина для обробки тютюну, жилки та їх змішування.	Mechanism
Цигаркова машина для формування тютюнового виробу	Машина для формування тютюнового джгута, виготовлення та нарізання цигаркового штрангу, для приєднання штрангу до фільтропалочки та обклеювання ободковим папером.	Mechanism
Конвеєр для пакування тютюнових виробів	Машина для створення картонної заготовки для пачки, наклеювання етикетки, створення розривної стрічки, формування «комірця», наклеювання акцизних марок, пакування цигарок в пачки та блоки.	Mechanism
Тютюновий виріб	Готовий продукт для імпорту / експорту.	Output

9. За допомогою кнопки **T** вносимо в поле діаграми текст з використанням редактора *Text Block Editor* – точки зору і мети (рисунок 1.15).

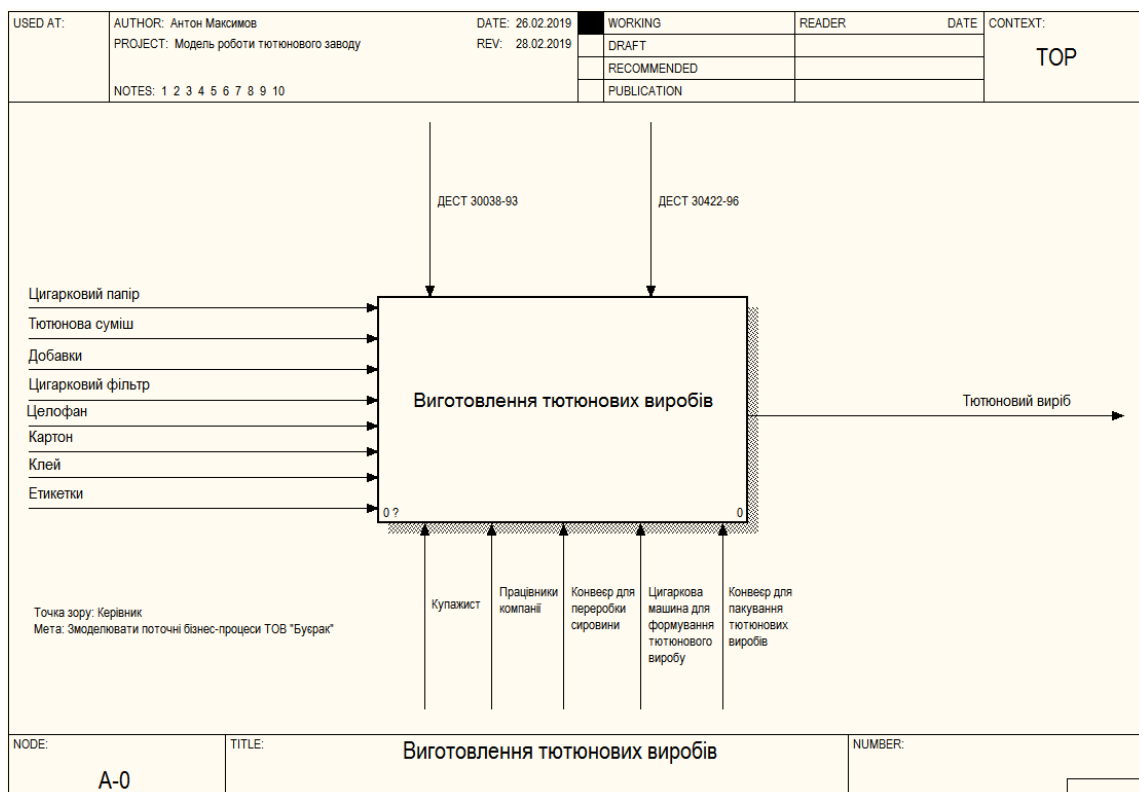


Рисунок 1.15 – Контекстна діаграма бізнес-процесу ТОВ «Буєрак» по виготовленню тютюнових виробів

10. Створимо звіт за моделлю. У меню *Tools/Reports/Model Report* (рисунок 1.16) задамо опції генерування звіту (встановлюємо прапорці) і натиснемо кнопку *Preview* (попередній перегляд) (рисунок 1.17).

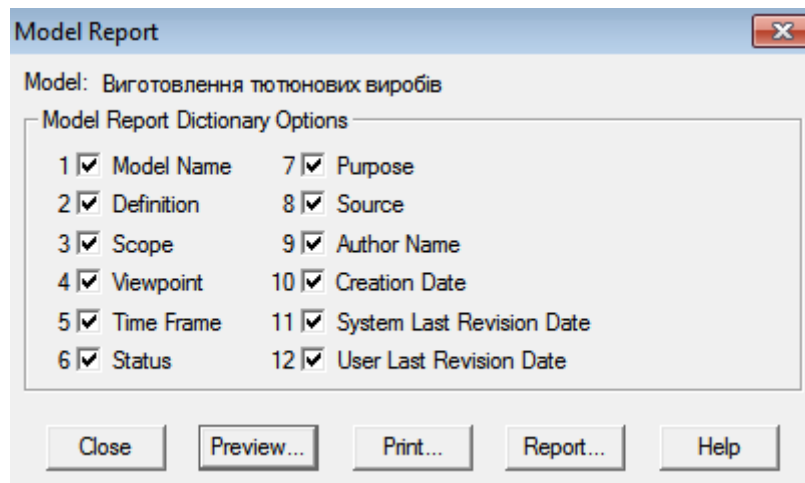


Рисунок 1.16 – Задання опцій генерування звіту *Model Report*

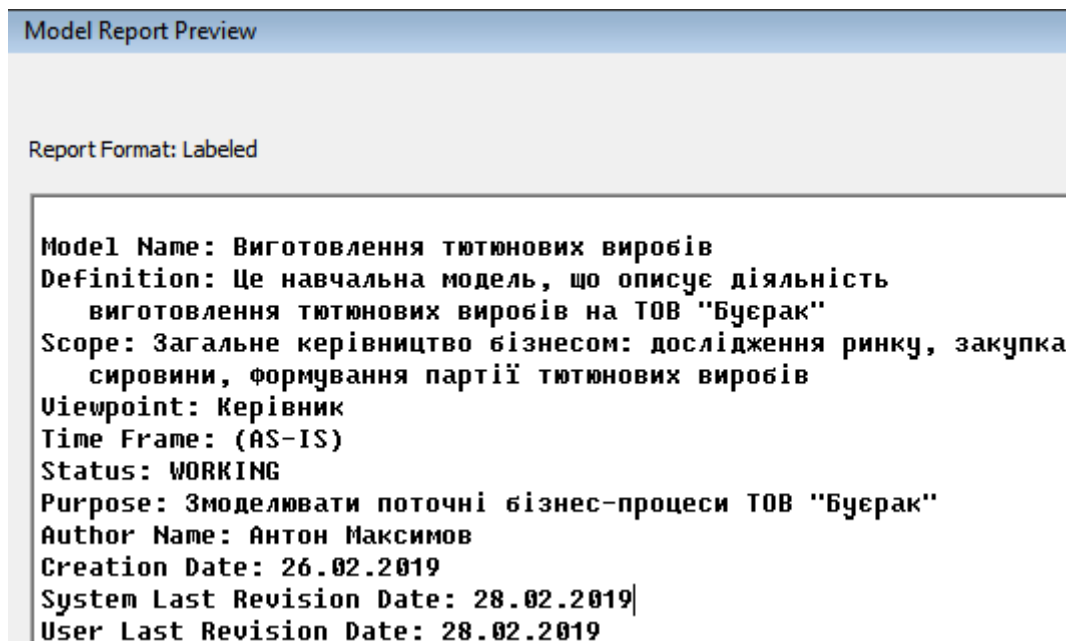


Рисунок 1.17 – Попередній перегляд звіту *Model Report*

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №2.
5. Висновки.

Контрольні питання

1. Назвіть призначення засобу AllFusion Process Modeler та його основні функції.
2. У чому полягають основні переваги AllFusion Process Modeler?
3. Які типи об'єктів існують у AllFusion Process Modeler?
4. Які моделі діаграм існують у AllFusion Process Modeler?
5. Назвіть основні елементи середовища AllFusion Process Modeler.
6. Що таке модель IDEF0?
7. Розкрийте сутність понять: «контекстна діаграма», «суб'єкт», «функціональна діаграма».
8. Назвіть області, які містить робоче вікно програми AllFusion Process Modeler.
9. Скільки типів стрілок існує в IDEF0? Охарактеризуйте кожен тип стрілок.
10. Для чого використовують Model Report?
11. Назвіть призначення стрілок типу Mechanism.
12. Назвіть призначення стрілок типу Output.
13. Назвіть призначення стрілок типу Control.
14. Назвіть призначення стрілок типу Input.
15. Назвіть призначення усіх об'єктів функціональної діаграми.
16. Назвіть призначення сторін прямокутників робіт на діаграмах.

ЛАБОРАТОРНА РОБОТА №2

Розробка моделі бізнес-процесу в нотації IDEF0. Створення діаграми декомпозиції

Мета роботи: Набути практичних навичок застосування інструментів засобу AllFusion Process Modeler для побудови діаграми декомпозиції бізнес-процесу в нотації IDEF0.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Загальна характеристика CASE-засобу AllFusion Process Modeler

Діаграми декомпозиції містять споріднені роботи, тобто дочірні роботи, що мають загальну батьківську роботу. Для створення діаграми декомпозиції слід клацнути на відповідній кнопці панелі інструментів.

Декомпозиція – це розділення модельованої роботи на роботи-компоненти.

У діалоговому вікні Activity Box Count (рис. 2.1) слід вказати нотацію нової діаграми і кількість робіт на ній (діапазон числа робіт 2–8).

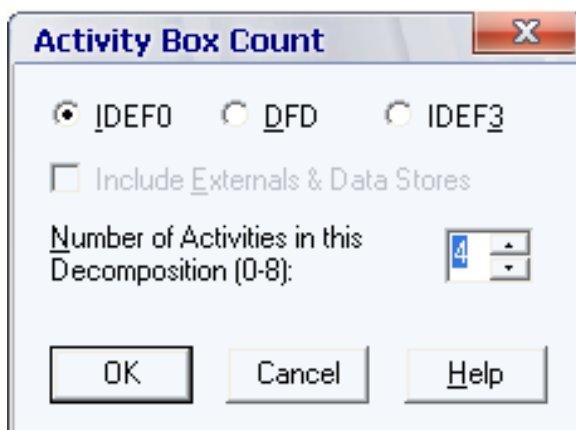


Рисунок 2.1 – Діалог Activity Box Count

Декомпозувати роботи на одну роботу не має сенсу. Діаграми з кількістю робіт більше восьми виходять перенасиченими і погано читаються. Для забезпечення наочності та кращого розуміння модельованих процесів рекомендується використовувати від трьох до шести блоків на одній діаграмі.

Якщо виявляється, що кількість робіт недостатня, то роботу можна додати в діаграму, клацнувши спочатку по відповідній кнопці на панелі інструментів, а потім по вільному місцю на діаграмі.

Роботи на діаграмах декомпозиції зазвичай розташовують по діагоналі від лівого верхнього кута до правого нижнього. Такий порядок називається *порядком домінування*. Згідно з цим принципом у лівому верхньому куті розташовується найважливіша робота або робота, що виконується за часом першою. Далі вправо вниз розташовуються менш важливі або роботи, які виконуються пізніше. Таке розташування полегшує читання діаграм, крім того, на ньому ґрунтується поняття взаємозв'язків робіт.

Отже, робота, що представлена на контекстній діаграмі верхнього рівня, може бути розкладена на основні підроботи за допомогою створення дочірньої діаграми. У свою чергу, кожна з цих підробіт може бути розкладена на складові частини за допомогою створення дочірньої діаграми наступного, нижчого рівня, на якій деякі або всі функції також можуть бути розкладені на складові. Кожна дочірня діаграма містить дочірні роботи і стрілки, що забезпечують додаткову деталізацію батьківської роботи.

Діаграма декомпозиції – це частина моделі, що описує декомпозицію блоку.

Дочірня діаграма, що створюється при декомпозиції, охоплює ту область, що і батьківський блок, але описує її детальніше. Таким чином, дочірня діаграма як би вкладена у свій батьківський блок (рис. 2.2).

Дочірній блок (робота): блок (робота) на дочірній діаграмі.

Батьківська діаграма – діаграма, яка містить один і більше батьківських блоків.

Будь-яка неконтекстна діаграма є також дочірньою діаграмою, оскільки за визначенням вона детально описує деякий батьківський блок. Таким чином, будь-яка діаграма може бути як батьківською діаграмою (містити батьківські блоки), так і дочірньою (детально описувати власний батьківський блок). Аналогічно, блок може бути як батьківським (детально описуватись дочірньою діаграмою), так і дочірнім (що з'являється на дочірній діаграмі). Основне ієрархічне відношення існує між батьківським блоком і дочірньою діаграмою (рис. 2.2).

Дочірня діаграма – це діаграма, що деталізує батьківський блок.

Батьківська робота – це робота, яка описується дочірньою діаграмою.

Те, що блок є дочірнім і розкриває зміст батьківського блоку на діаграмі попереднього рівня, вказується спеціальним кодом, що розміщується у нижньому правому куті блоку (*Model/Model Properties/Numbering*). Цей код може формуватися декількома способами, з яких найпростіший полягає у такому: код, який починається з букви А, містить цифри, що визначаються номерами батьківських блоків.

Номер блоку – число (0–8), що розміщується у правому нижньому куті блоку й однозначно ідентифікує блок на діаграмі.

У методології IDEF0 існує шість типів відношень між блоками в межах однієї діаграми:

- домінування;

- управління;
- вихід – вхід;
- зворотний зв'язок по управлінню;
- зворотний зв'язок по входу;
- вихід – механізм.

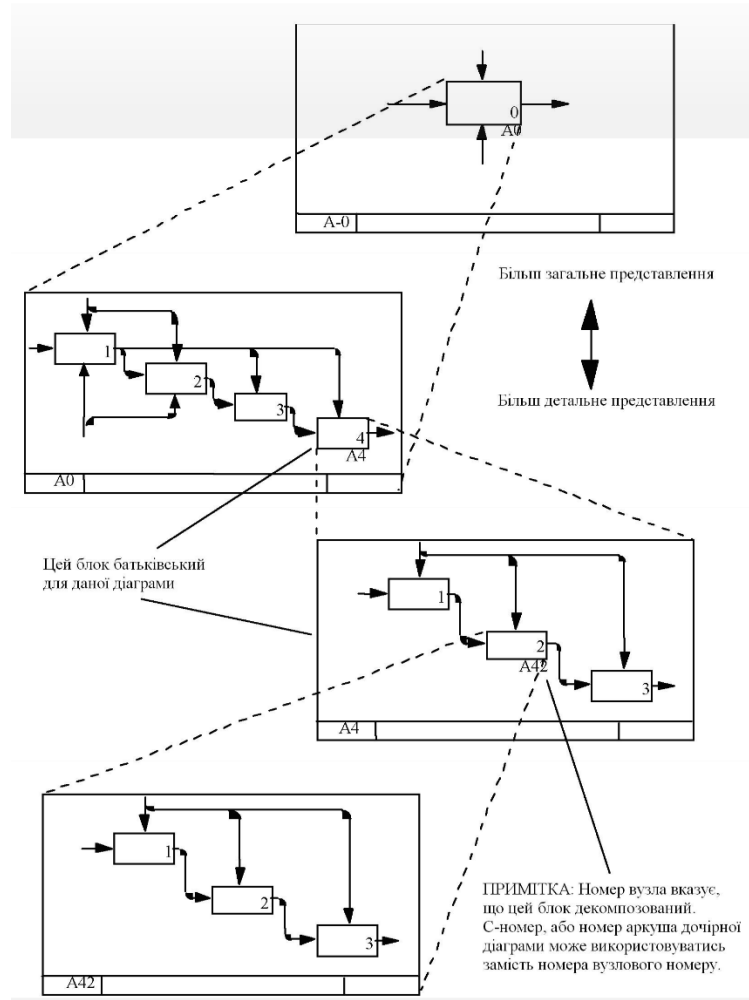


Рисунок 2.2 – Ілюстрація структури декомпозиції

Перше з перерахованих відношень визначається взаємним розташуванням блоків на діаграмі. Передбачається, що блоки, розташовані на діаграмі вище і лівіше, «домінують» над блоками, розташованими нижче і правіше.

Під «домінуванням» розуміють вплив, який один блок має на інші блоки діаграми.

Останні п'ять відношень описують зв'язки між блоками і зображуються відповідними стрілками.

Відношення управління і вихід-вхід є простими, оскільки відображають прямі взаємодії, які зрозумілі й очевидні.

Відношення управління (рис. 2.3) виникає тоді, коли вихід одного блоку служить дією, що керує блоком з меншим домінуванням.

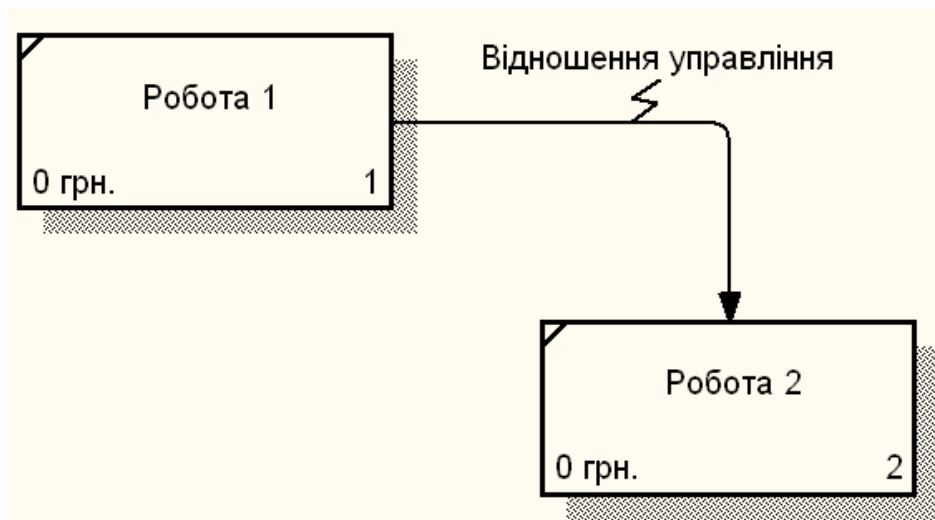


Рисунок 2.3 – Відношення управління

Відношення вихід-вхід (рис. 2.4) виникає при з'єднанні виходу одного блоку з входом іншого блоку з меншим домінуванням. Зворотний зв'язок по управлінню і зворотний зв'язок по входу є складнішими типами відношень, оскільки вони являють собою ітерацію (вихід функції впливає на майбутнє виконання інших функцій з великим домінуванням, що згодом впливає на початкову функцію).

Зворотний зв'язок по управлінню (рис. 2.5) виникає тоді, коли вихід деякого блоку створює дію, що впливає, на блок з більшим домінуванням.

Відношення «Зворотного зв'язку по входу» (рис. 2.6) має місце тоді, коли вихід блоку може стати входом іншого блоку з більшим домінуванням.

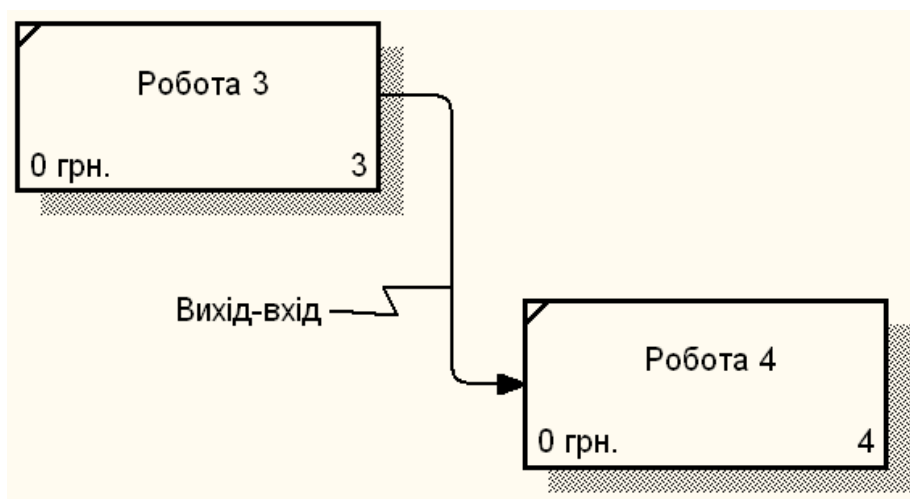


Рисунок 2.4 – Відношення *Вихід-вхід*

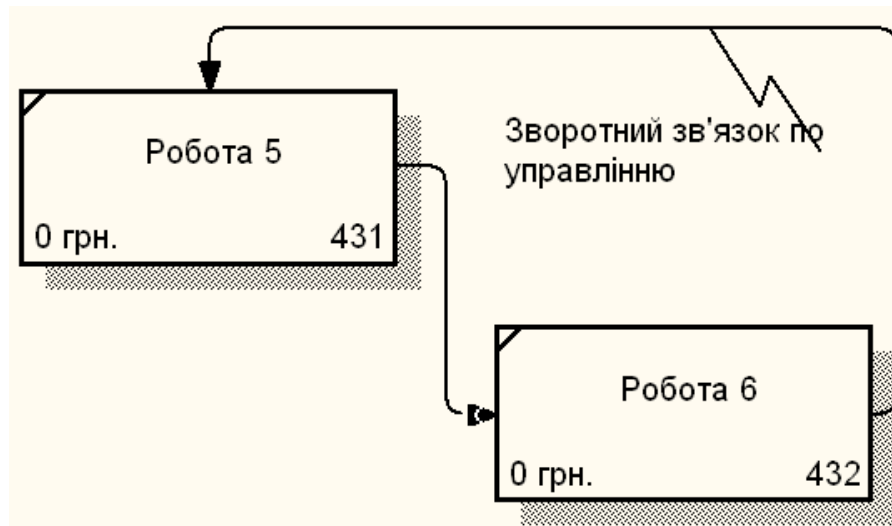


Рисунок 2.5 – Зворотний зв'язок по управлінню

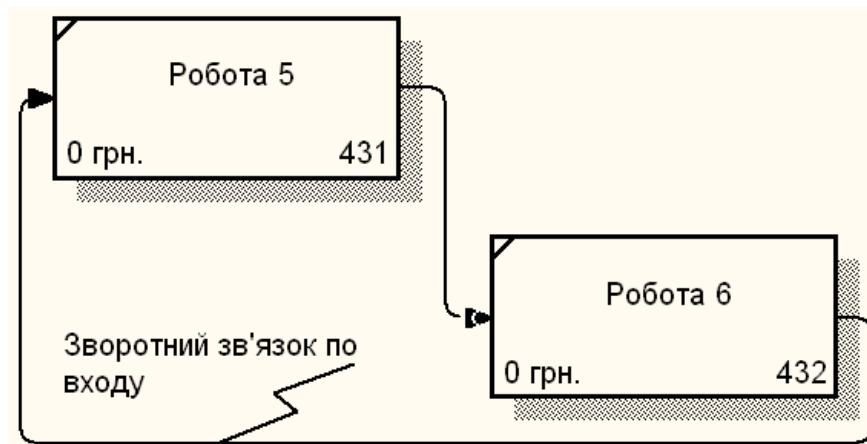


Рисунок 2.6 – Зворотний зв'язок по входу

Відношення «Вихід – механізм» (рис. 2.7) відображають ситуацію, при якій вихід однієї функції може ставати засобом досягнення мети для іншої.

Зв'язки «Вихід – механізм» виникають під час відображення у моделі процедур поповнення і розподілу ресурсів, створення або підготовки засобів для виконання функцій системи (наприклад, придбання або виготовлення необхідних інструментів і устаткування, навчання персоналу, організація фізичного простору, фінансування, закупівля матеріалів тощо).

Тунель – дужки на початку і/або закінченні стрілки. Тунельні стрілки означають, що дані, виражені цими стрілками, не розглядаються на батьківській діаграмі і/або на дочірній діаграмі. Внесені граничні стрілки на діаграмі декомпозиції нижнього рівня зображуються у квадратних дужках і автоматично не з'являються на діаграмі верхнього рівня (рис. 2.8, а).

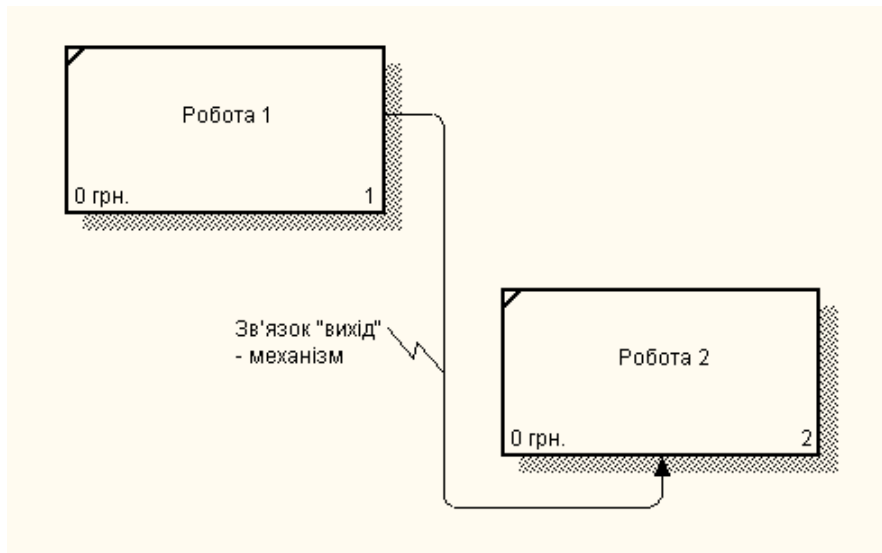


Рисунок 2.7 – Відношення-зв'язок «вихід-механізм». Стрілки, поміщені в «тунель»

Відображення стрілки із квадратними дужками у кінцевому варіанті діаграми вважається помилкою. Необхідно, щоб дана стрілка відображалась на батьківській/дочірній діаграмі або взята у круглі дужки (поміщена в тунель) (рис. 2.8, б). Стрілка, поміщена в тунель, там, де вона приєднується до блоку, означає, що дані, виражені цією стрілкою, не обов'язкові та не відображаються на наступному рівні декомпозиції.

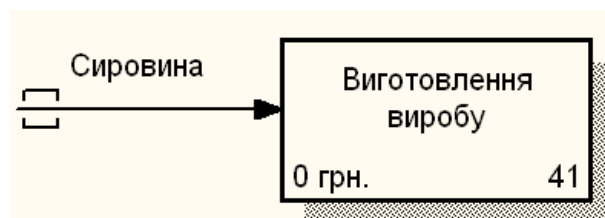


Рисунок 2.8, а – Відображення граничних стрілок на діаграмі декомпозиції

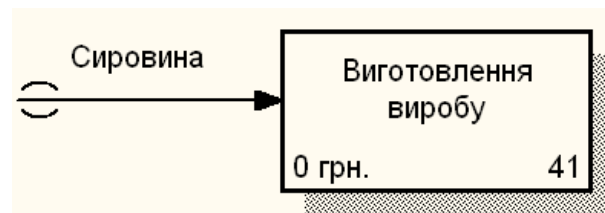


Рисунок 2.8, б – Відображення тунельних стрілок

Для того, щоб дана стрілка відображалась на батьківській діаграмі або була взята у круглі дужки (поміщена в тунель), необхідно вибрати у контекстному меню (рис. 2.9) закінчення/початок стрілки пункт *Arrow tunnel....*

У діалоговому вікні *Border Arrow Editor* вибір варіанта *Resolve it to border arrow* призведе до відображення стрілки на батьківській чи дочірній діаграмах, а *Change it to resolved rounded tunnel* – до тунелювання.

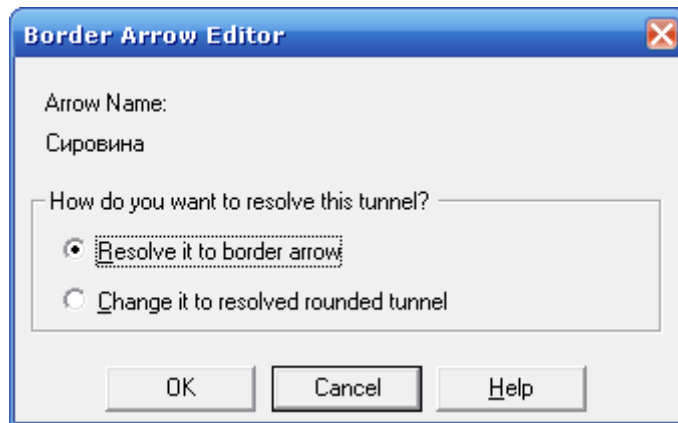


Рисунок 2.9 – Діалогове вікно редагування стрілки

2. Стрілки як обмеження

Стрілки на діаграмі IDEF0, являючи собою дані або матеріальні об’єкти, одночасно задають свого роду обмеження (умови). Вхідні та керуючі стрілки роботи, що сполучають його з іншими роботами або із зовнішнім середовищем, по суті, описують умови, які повинні бути виконані для того, щоб реалізувалася функція, записана як ім’я роботи.

2.1 Паралельне функціонування

Різні роботи в моделі можуть бути виконані паралельно, якщо задовольняються необхідні обмеження (умови).

Один блок може створити дані або матеріальні об’єкти, необхідні для паралельної роботи декількох блоків (рис. 2. 10).

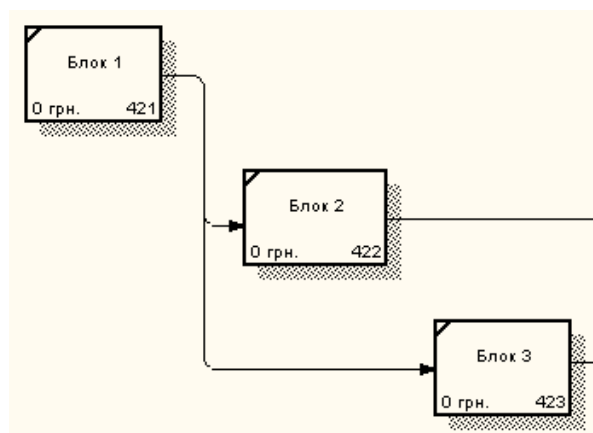


Рисунок 2.10 – Паралельна робота блоків

2.2 Розгалуження і злиття сегментів стрілок

Розгалуження і злиття стрілок покликане зменшити завантаженість діаграм графічними елементами (лініями). Мітки зв'язуються з сегментами за допомогою тильд. Мітку можна присвоїти окремій гілці стрілки.

3. Основні правила побудови діаграм декомпозиції в нотації IDEF0

- Блоки завжди повинні містити хоча б одну управляючу й одну вихідну стрілку, але можуть не мати вхідних стрілок.
- Роботи розміщуються по діагоналі.
- Якщо одні і ті ж дані служать і для управління і для входу, будується тільки стрілка управління. Цим підкреслюється керуючий характер даних і зменшується складність діаграми.
- Максимальна відстань між вхідними і вихідними стрілками.
- Максимальна відстань між поворотами і перетином стрілок.
- Якщо дві стрілки паралельні, їх варто об'єднати.
- Зворотній зв'язок по входу будується за допомогою нижньої петлі.
- Зворотній зв'язок по управлінню будується за допомогою верхньої петлі.
- Мінімум перетинів.

Розгалуження – це розділення стрілки на два і більше сегментів. Може означати «розв'язування пучка».

Сегмент стрілки – це сегмент лінії, який починається або закінчується на стороні блоку, в точці розгалуження або злиття, або на границі (незв'язаний кінець стрілки).

Злиття – це об'єднання двох або більшого числа сегментів стрілок в один сегмент (може означати «зв'язування пучка»).

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler з використанням стандарту функціонального моделювання IDEF0 та згідно номеру варіанту (табл. 1.1).

Завдання 1. Розглянути приклад 1.


Завдання 2. Сформувати контекстну діаграму системи відповідно до методології IDEF0, задавши входи, виходи, механізми й керування.

Завдання 3. Виконати декомпозицію контекстної діаграми з проведенням зв'язків по входу, виходу та управлінню.

Завдання 4. Побудувати діаграму декомпозиції верхнього рівня.

Завдання 5. Побудувати діаграму декомпозиції роботи A2.

Приклад 1. Побудова діаграми декомпозиції верхнього рівня на прикладі діяльності комп'ютерної компанії.

1. Виберемо кнопку  переходу на нижній рівень в діалоговому вікні *Activity Box Count* (рис. 2.11) та встановимо число робіт на діаграмі нижнього рівня – 3. Після цього автоматично буде створена діаграма декомпозиції.

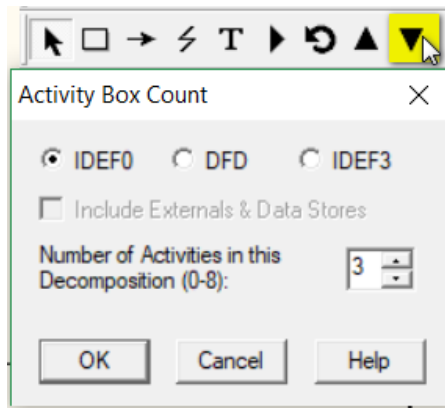


Рисунок 2.11 – Діалогове вікно *Activity Box Count*

2. Вносимо по чергову імена та визначення робіт (табл. 2.2).

Таблиця 2.2 – Роботи діаграми декомпозиції A0

Назва роботи (Activity Name)	Визначення роботи (Activity Definition)
Продажі і маркетинг	Телемаркетинг і презентації, виставки.
Збірка та тестування комп'ютерів	Збірка і тестування настільних та портативних комп'ютерів.
Відвантаження і одержання	Відвантаження замовлень клієнтам та одержання компонентів від постачальників.

3. Правою кнопкою миші потрібно клацнути по роботі, вибрати опцію *Name* і ввести ім'я роботи. Потрібно повторити операцію для всіх трьох робіт. Потім вносимо визначення, статус і джерело для кожної роботи згідно з табл. 2.2. Діаграма декомпозиції набуде вигляду, який представлено на рис. 2.12.

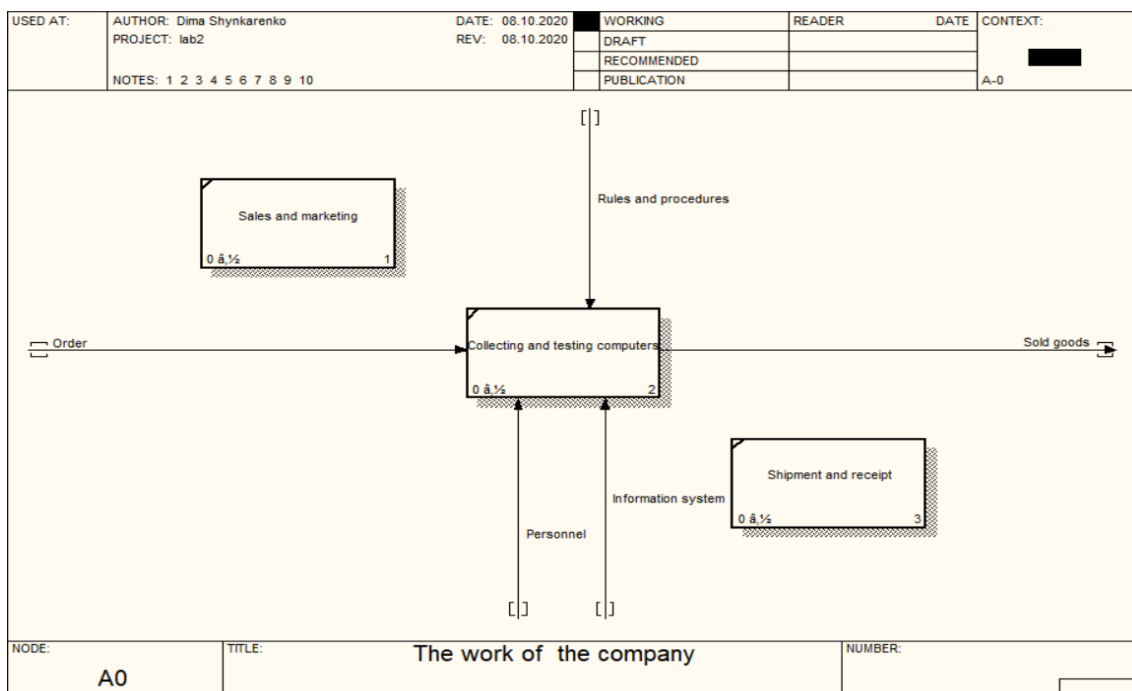


Рисунок 2.12 – Діаграма декомпозиції після присвоєння імен роботам

4. Для зміни властивостей робіт після їх внесення до діаграми можна скористатися словником робіт (рис. 2.13). Виклик словника проводиться за допомогою пункту головного меню *Dictionary /Activity*.

Name	Definition	Author	Status
Collecting and testing computers	Collect and test desktops and laptops.	Dima Shynkarenko	WORKING
Sales and marketing	Telemarketing and presentations, exhibitions.	Dima Shynkarenko	WORKING
Shipment and receipt	Shipment of orders to customers and receipt of components from suppliers.	Dima Shynkarenko	WORKING
The work of the company	Existing business processes	Dima Shynkarenko	WORKING

Рисунок 2.13 – Словник *Activity Dictionary*

5. Переходимо в режим побудови стрілок і пов'яжемо граничні стрілки, скориставшись кнопкою на панелі інструментів (рис. 2.14).

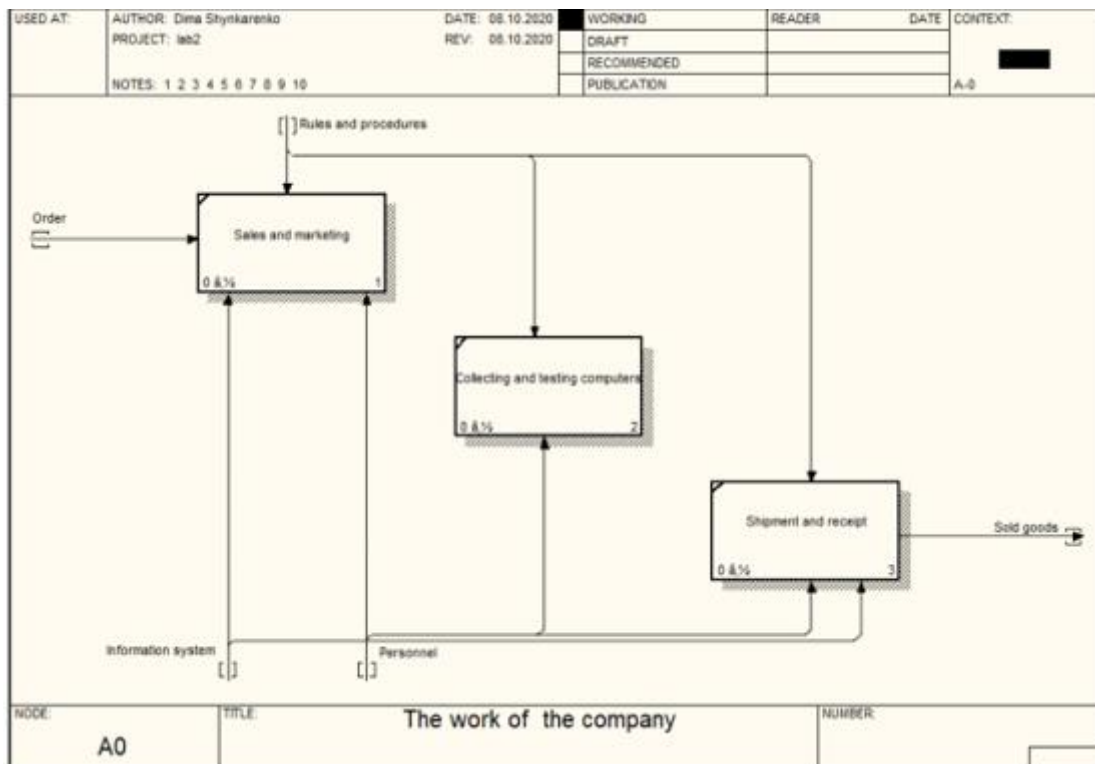


Рисунок 2.14 – Пов'язані граничні стрілки на діаграмі A0

6. Правою кнопкою миші натиснемо по сегменту стрілки управління роботою «Збірка і тестування комп'ютерів» і перейменуємо її в «Правила збору і тестування» (рис. 2.15). Вносимо визначення для нового сегмента стрілки: «Інструкції по збору, процедури тестування, критерії продуктивності тощо». Правою кнопкою миші натиснемо по гілці стрілки механізму роботи «Продажі і маркетинг» і перейменуємо її в «Система оформлення замовлень».

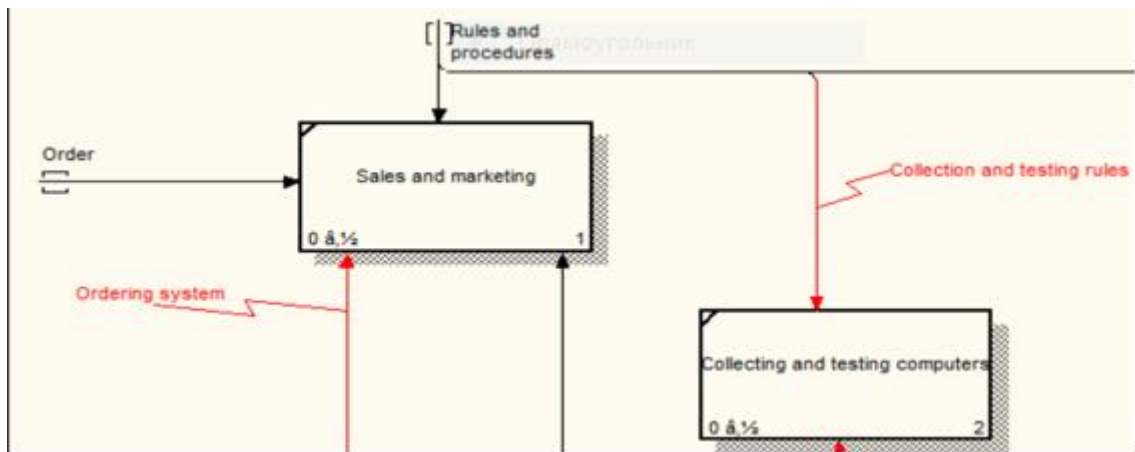


Рисунок 2.15 – Стрілка «Правила збору і тестування»

7. Аналогічно перейменуємо гілки стрілки «Персонал» відповідно «Персонал відділу маркетингу», «Персонал виробничого відділу» та «Складський персонал» (рис. 2.16).

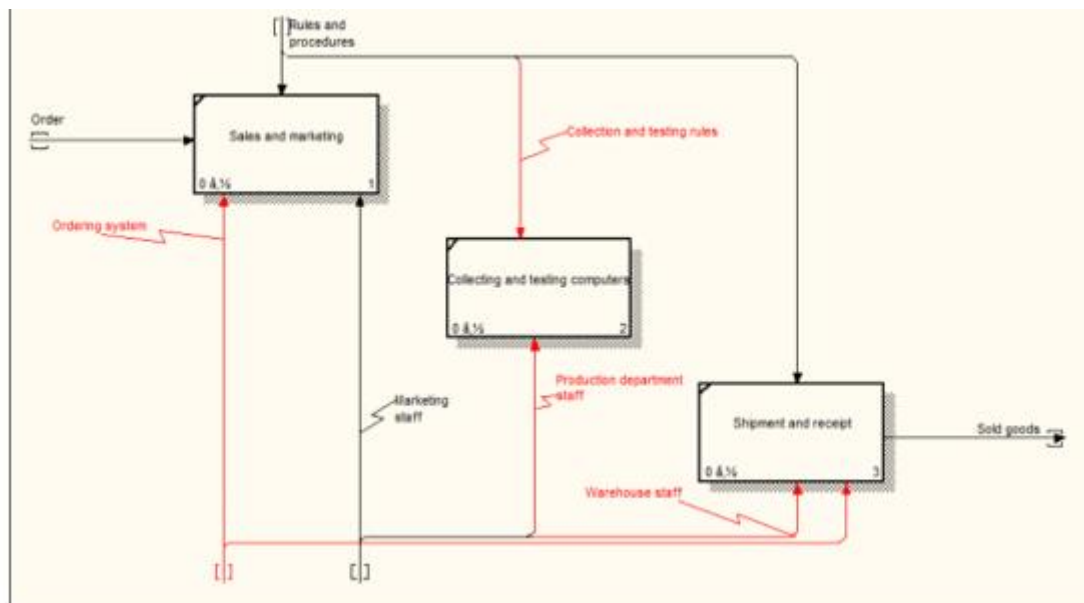


Рисунок 2.16 – Результат перейменування гілок

8. Альтернативний метод внесення імен і властивостей стрілок – використання словника стрілок (виклик словника – меню *Dictionary/Arrow*). Якщо вносити ім'я і властивості стрілки до словника (рис. 2.17), її можна буде вносити до діаграми пізніше.

9. Створимо нові внутрішні стрілки так, як вказано на рис. 2.18.

10. Створимо стрілку зворотного зв'язку (по управлінню) «Результати збору та тестування», яка проходить від роботи «Збірка і тестування комп'ютерів» до роботи «Продажі і маркетинг». Змінимо стиль стрілки (товщина ліній) і встановимо опцію *Extra Arrowhead* (додатковий наконечник стріли). Методом *drag&drop* перенесемо імена стрілок (мітки) так, щоб їх було зручніше читати та встановимо параметр *Squiggle* (Тильда).

Name	Definition	Author	Status
Order	Requests for information in any form	Dima Shynkarenko	WORKING
Customer orders		Dima Shynkarenko	WORKING
Assembled computers		Dima Shynkarenko	WORKING
Information system	Registration of accounts, payment	Dima Shynkarenko	WORKING
The final product	Desktops	Dima Shynkarenko	WORKING
Marketing materials		Dima Shynkarenko	WORKING
Personnel		Dima Shynkarenko	WORKING
Production department staff		Dima Shynkarenko	WORKING
Marketing staff		Dima Shynkarenko	WORKING
Collection and testing rules		Dima Shynkarenko	WORKING
Rules and procedures		Dima Shynkarenko	WORKING
Sold goods		Dima Shynkarenko	WORKING
The result of the collection of testing		Dima Shynkarenko	WORKING
Ordering system		Dima Shynkarenko	WORKING
Warehouse staff		Dima Shynkarenko	WORKING

Рисунок 2.17 – Словник стрілок

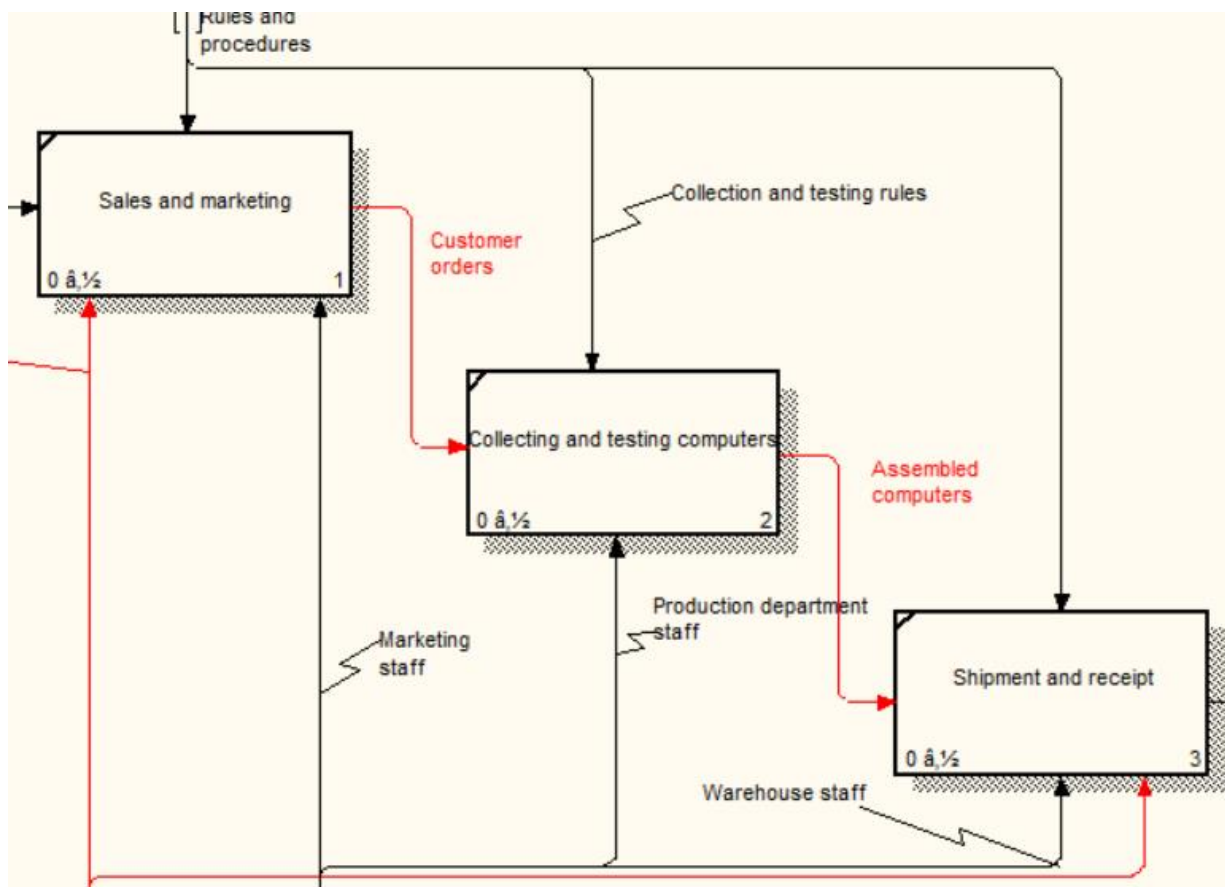


Рисунок 2.18 – Внутрішні стрілки діаграми А0

Результат можливих змін представлено на рис. 2.19.

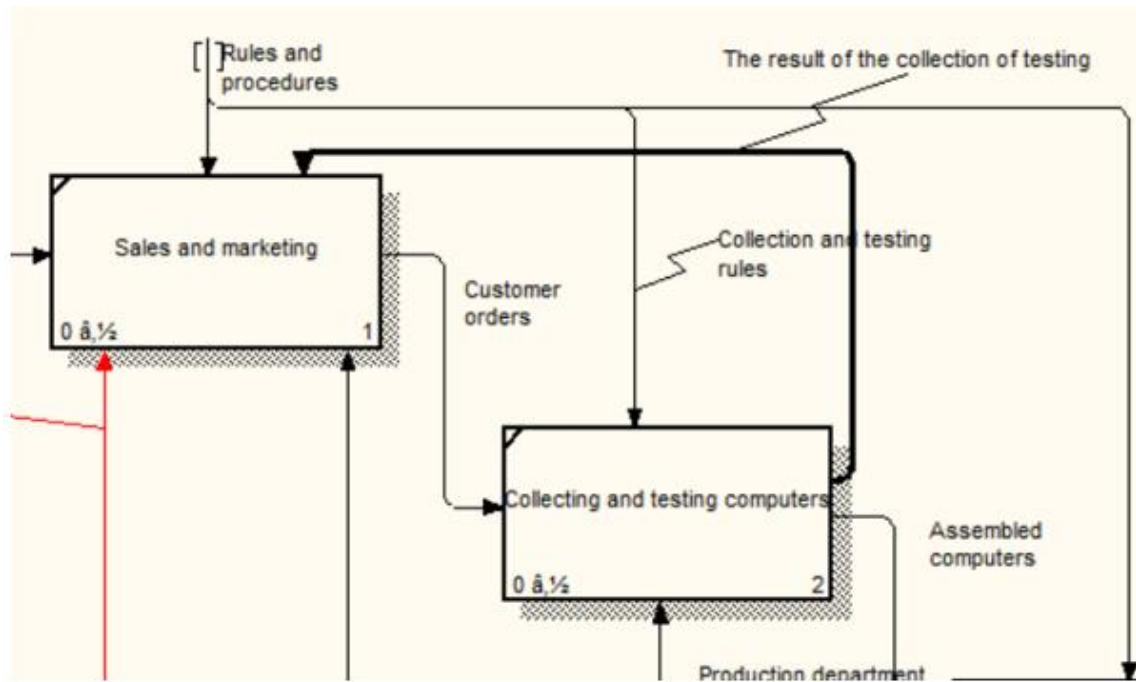


Рисунок 2.19 – Результат редагування стрілок на діаграмі A0

10. Створимо нову граничну стрілку виходу «Маркетингові матеріали», що виходить з роботи «Продажі і маркетинг». Ця стрілка автоматично не потрапляє на діаграму верхнього рівня і зображена з одного боку квадратними дужками $\left[\right]$ (рис. 2.20).

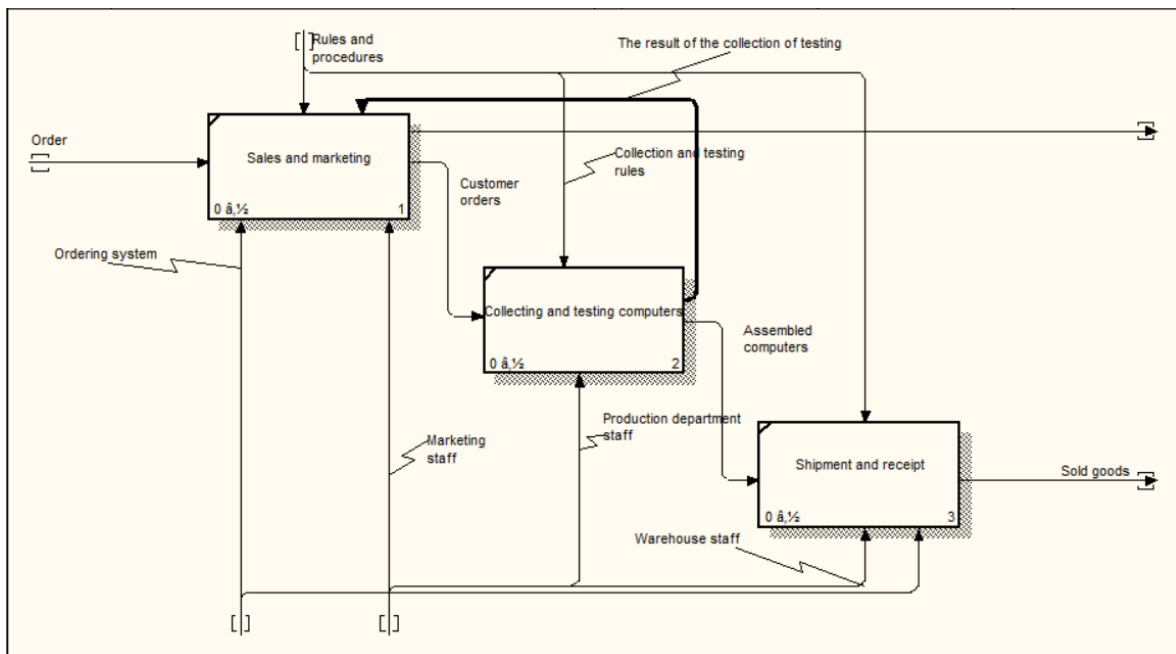


Рисунок 2.20 – Стрілка «Маркетингові матеріали»

11. Натиснемо правою кнопкою миші на квадратних дужках і виберемо пункт меню *Arrow Tunnel* (рис. 2.21). У діалоговому вікні *Border Arrow Editor* (Редактор граничних стрілок) виберемо опцію *Resolve it to Border Arrow* (Визначити як граничну стрілку) (рис. 2.22).

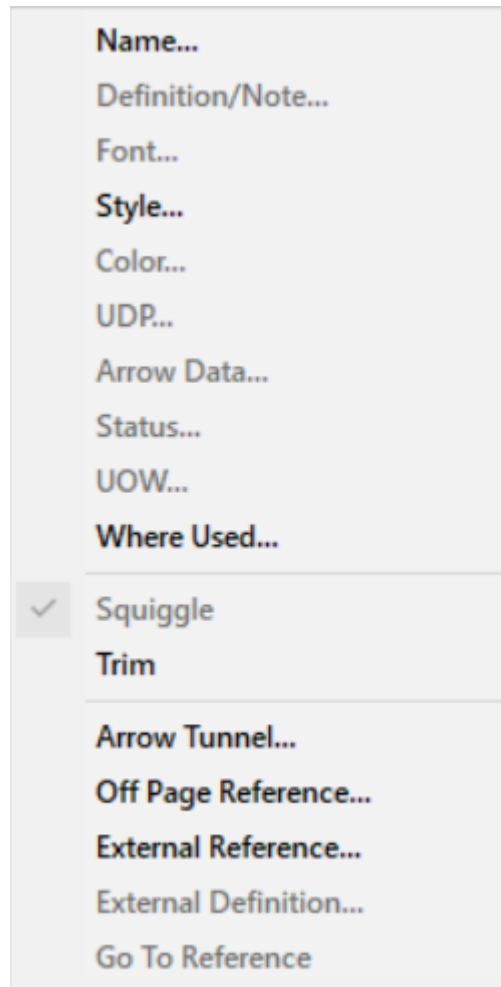


Рисунок 2.21 – Пункт меню *Arrow Tunnel*

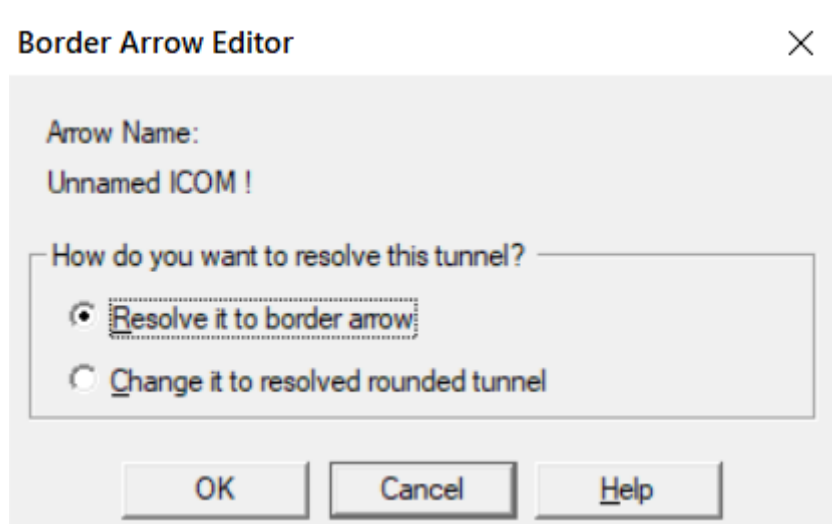


Рисунок 2.22 – Діалогове вікно *Border Arrow Editor*

12. Для стрілки «Маркетингові матеріали» виберемо опцію *Trim* (упорядкувати) з контекстного меню. Результат виконання завдання представлено на рис. 2.23.

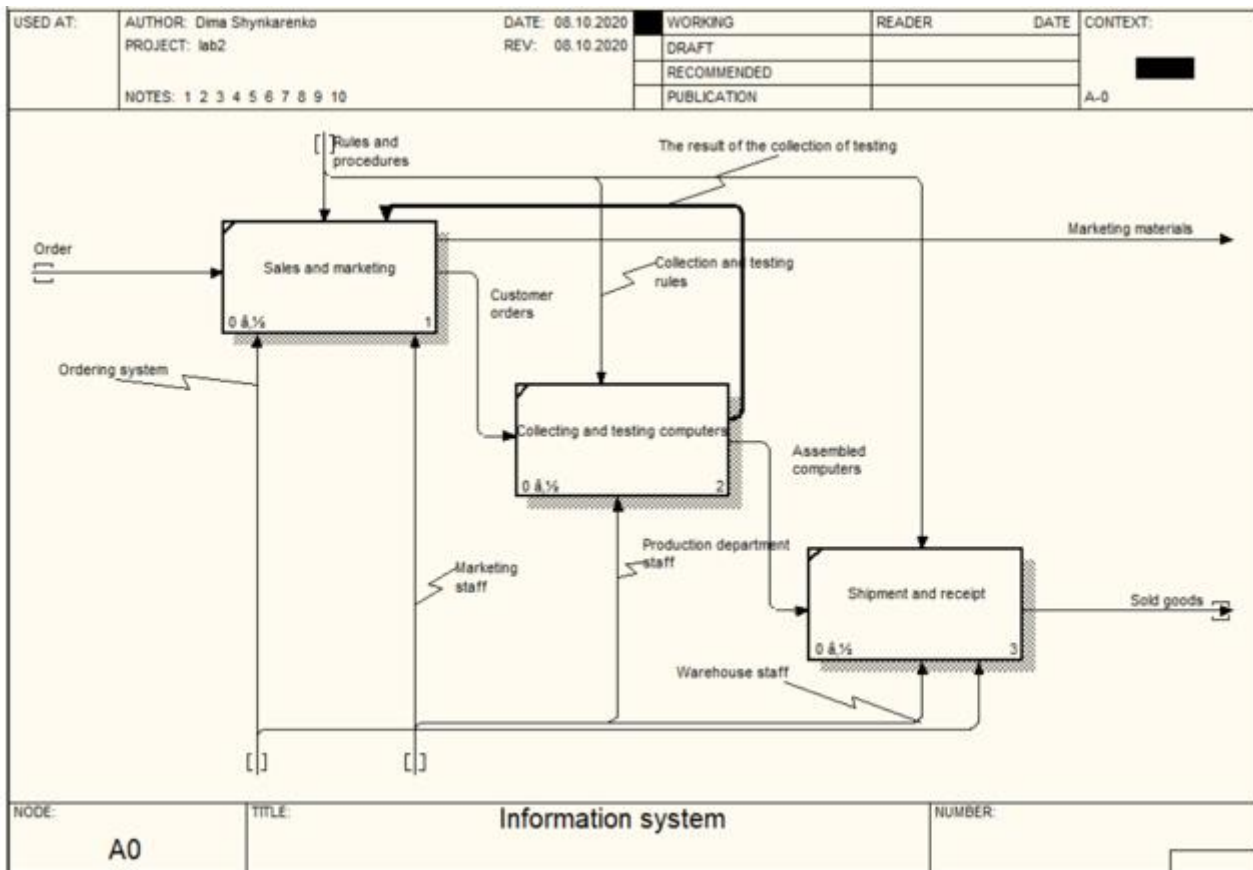


Рисунок 2.23 – Результат виконання завдання 1 – діаграма A0

Приклад 2. Побудова діаграми декомпозиції роботи A2 на прикладі діяльності комп’ютерної компанії.

1. На основі отриманої інформації вносимо нові роботи і стрілки згідно з таблицями 2.3, 2.4.

Таблиця 2.3 – Роботи діаграми декомпозиції A0

Назва роботи (Activity Name)	Визначення роботи (Activity Definition)
Відстежування розкладу і управління складанням і тестуванням	Перегляд замовлень, створення графіка виконання замовлень, перегляд результатів тестування, формування груп замовлень на збір і відвантаження.
Збірка настільних комп’ютерів	Збирання настільних комп’ютерів відповідно до інструкцій і вказівок диспетчера.
Збірка ноутбуків	Збирання ноутбуків відповідно до інструкцій вказівок диспетчера.
Тестування комп’ютерів	Тестування комп’ютерів і компонентів. Заміна непрацюючих компонентів.

Таблиця 2.4 – Стрілки діаграми декомпозиції A2

Назва стрілки (Arrow Name)	Джерело стрілки (Arrow Source)	Тип джерела стрілки (Arrow Source)	Приймач стрілки (Arrow Dest.)	Тип стрілки приймача (Arrow Dest. Type)
1	2	3	4	5
Диспетчер	Персонал виробничого відділу		Відстеження розкладу і управління складанням і тестуванням	Mechanism
Замовлення клієнтів	Межа діаграми	Control	Відстеження розкладу і управління складанням і тестуванням	Control
Замовлення на настільні комп'ютери	Відстеження розкладу і управління збиранням і тестуванням	Output	Збирання настільних комп'ютерів	Control
Замовлення на ноутбуки	Відстеження збиранням і управління складанням і тестуванням	Output	Збирання ноутбуків	Control
Компоненти	«Tunnel»	Input	Збирання настільних комп'ютерів	Input
			Збирання ноутбуків	Input
			Тестування комп'ютерів	Input
Комп'ютери	Збирання настільних	Output	Тестування комп'ютерів	Input
Настільні комп'ютери	«Tunnel»	Output	Збирання настільних комп'ютерів	Mechanism
Ноутбуки	Збирання ноутбуків	Output	Тестування комп'ютерів	Input

1	2	3	4	5
Персонал виробничого відділу			Збирання настільних комп'ютерів	Mechanism
			Збирання ноутбуків	Mechanism
Правила збору і тестування	Межа діаграми		Збирання настільних комп'ютерів	Control
			Збирання ноутбуків	Control
			Тестування комп'ютерів	Control
Результати збору і тестування	Збирання настільних комп'ютерів	Output	Межа діаграми	Output
	Збирання			
	Тестування комп'ютерів			
Результати тестування	Тестування комп'ютерів	Output	Відстеження розкладу і управління з збиранням і тестуванням	Input
Складені комп'ютери	Тестування комп'ютерів	Output	Межа діаграми	Output
Тестувальник	Персонал виробничого відділу		Тестування комп'ютерів	Mechanism
Вказівка передати комп'ютери на відвантаження	Відстеження розкладу і управління збиранням і тестуванням	Output	Тестування комп'ютерів	Control

2. Тунелюємо і зв'язуємо граничні стрілки на верхньому рівні. Результат виконання завдання 2 представлено на рис. 2.24.
3. Проаналізуємо отримані результати.

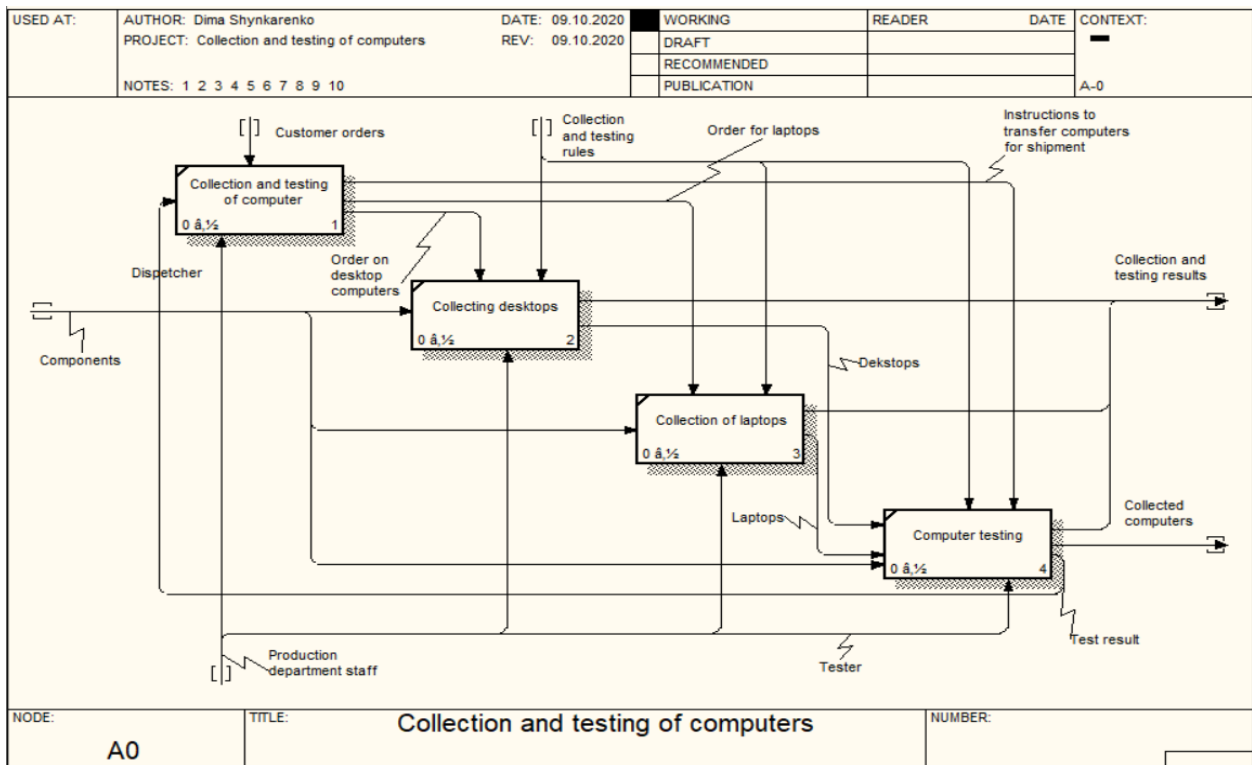


Рисунок 2.24 – Результат виконання завдання 2 – діаграма A2

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв’язання завдання №2.
5. Протокол розв’язання завдання №3.
6. Протокол розв’язання завдання №4.
7. Протокол розв’язання завдання №5.
8. Висновки.

Контрольні питання

1. Що таке модель IDEF0?
2. Розкрийте сутність понять: «порядком домінування», «дочірна діаграма», «батьківська діаграма».
3. Який порядок іменування робіт?
4. Яка кількість робіт може бути присутньою на одній діаграмі?
5. Як розташовуються роботи за принципом домінування?
6. Яке призначення сторін прямокутників робіт на діаграмах?
7. Назвіть типи стрілок.
8. Назвіть види взаємозв’язків між роботами.
9. Поясніть принцип іменування стрілок, що розгалужуються і зливаються.
10. Як створити зв’язок між роботами?
11. Опишіть процес декомпозиції роботи.

ЛАБОРАТОРНА РОБОТА №3

Створення діаграми дерева вузлів та FEO діаграми

Мета роботи: Набути практичних навичок застосування інструментів засобу AllFusion Process Modeler з використанням стандарту функціонального моделювання IDEF0 при створенні діаграми дерева вузлів та FEO-діаграми.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Створення дерева вузлів

Діаграма дерева вузлів відображає ієрархію робіт у моделі і дозволяє розглянути модель у цілому, але не відображає взаємозв'язки між роботами (стрілки) (рис. 3.1). Процес створення моделі робіт є ітераційним, отже, роботи можуть міняти своє розташування в дереві вузлів багато разів. Щоб не заплутатися і перевірити спосіб декомпозиції, слід після кожної зміни створювати діаграму дерева вузлів. Втім, AFPM має могутній інструмент навігації по моделі – Model Explorer, який дозволяє представити ієрархію робіт і діаграм у зручному і компактному вигляді, проте на цей інструмент не поширюються правила стандарту IDEF0.

Для створення діаграми дерева вузлів слід вибрати в меню *Diagram/Add Node Tree* (рис. 3.2). З'явиться діалогове вікно майстра створення діаграми дерева вузлів *Node Tree Wizard* (рис. 3.3 – 3.4).

У діалоговому вікні слід вказати глибину дерева – *Number of Levels* (за замовчуванням – 3) і корінь дерева (за замовчуванням – батьківська робота поточної діаграми). За замовчуванням нижній рівень декомпозиції відображається у вигляді списку, останні роботи – у вигляді прямокутників. Для відображення всього дерева у вигляді прямокутників слід вимкнути опцію *Bullet Last Level* (зняти прапорець).

Примітка: Під час створення дерева вузлів слід вказати ім'я діаграми, оскільки, якщо в декількох діаграмах як корінь дерева вузлів використовувати одну і ту ж роботу, в цьому випадку всі ці діаграми отримають однаковий номер (номер вузла + постфікс N, наприклад AON) і в списку відкритих діаграм (пункт меню Windows) їх можна буде розрізнити тільки за ім'ям.

Дерево вузлів – це представлення відношень між батьківськими і дочірніми вузлами моделі IDEF0 у формі деревовидного графа. Має те ж значення і зміст, що і перелік вузлів.

Перелік вузлів – це список, що відображає вузли моделі IDEF0 у впорядкованому вигляді.

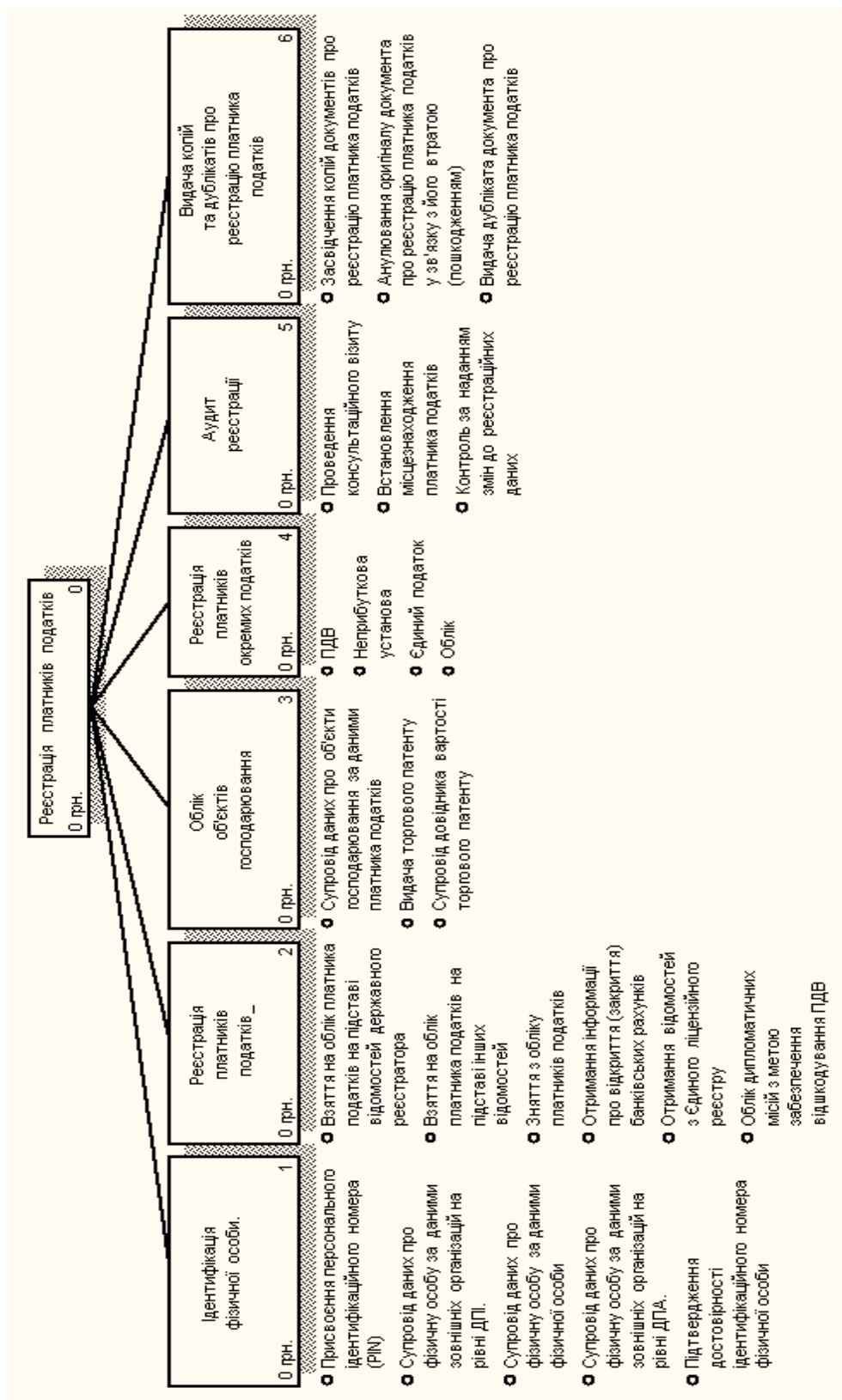


Рисунок 3.1 – Приклад діаграми дерева вузлів

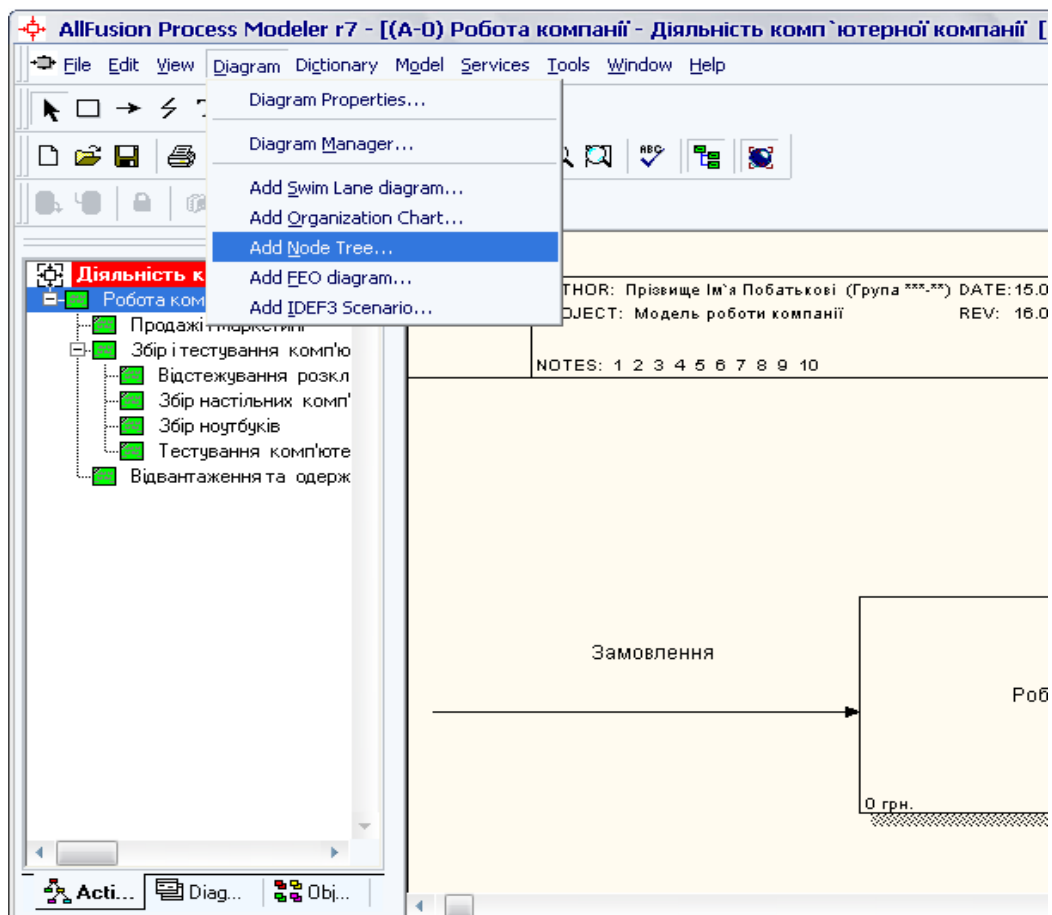


Рисунок 3.2 – Пункт головного меню *Diagram/Add Node Tree*

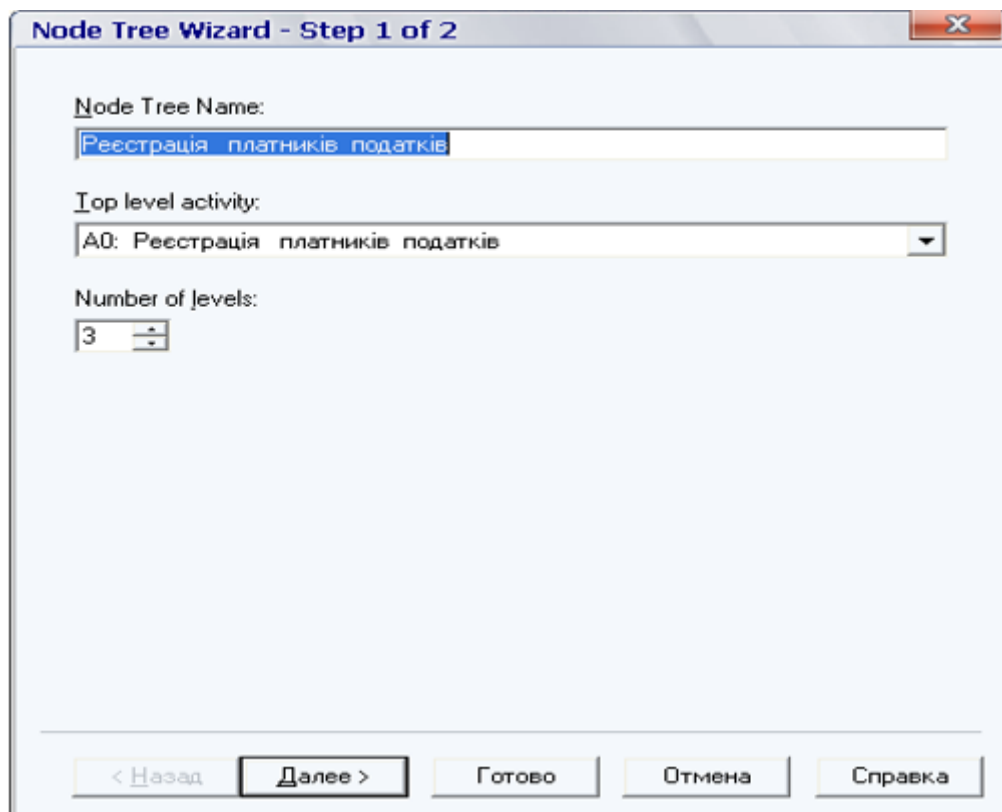


Рисунок 3.3 – Майстер створення діаграми дерева вузлів, крок 1



Рисунок 3.4 – Майстер створення діаграми дерева вузлів, крок 2

2. Створення FEO-діаграм

Діаграми FEO (For Exposition Only – FEO) – «тільки для експозиції» – часто використовуються у моделі для ілюстрації інших точок зору, для відображення окремих деталей, які не підтримуються синтаксисом IDEF0. Діаграми FEO дозволяють порушити будь-яке синтаксичне правило, оскільки, по суті, є просто картинками – копіями стандартних діаграм і не включаються в аналіз синтаксису.

Наприклад, робота на діаграмі FEO може не містити стрілок управління і виходу. З метою обговорення певних аспектів моделі з експертом предметної області може бути створена діаграма тільки з однією роботою і однією стрілкою, оскільки стандартна діаграма декомпозиції містить безліч деталей, що не відносять до теми обговорення і дезорієнтують експерта.

Але, якщо FEO використовується для ілюстрації альтернативних точок зору (альтернативний контекст), рекомендується все-таки дотримуватися синтаксису IDEF0. Для створення діаграми FEO слід вибрати пункт меню *Diagram/Add FEO Diagram*. У виникаючому вікні діалогу *Add New FEO Diagram* слід вказати ім'я *FEO-діаграми* і тип батьківської діаграми (рис. 3.5).

Нова діаграма отримує номер, який генерується автоматично (номер батьківської діаграми по вузлу + постфікс F, наприклад A1F).

Діаграма-ілюстрація (FEO) – це графічний опис, що використовується для повідомлення специфічних фактів про діаграму IDEF0. При побудові діаграм FEO можна не дотримуватися правила IDEF0.

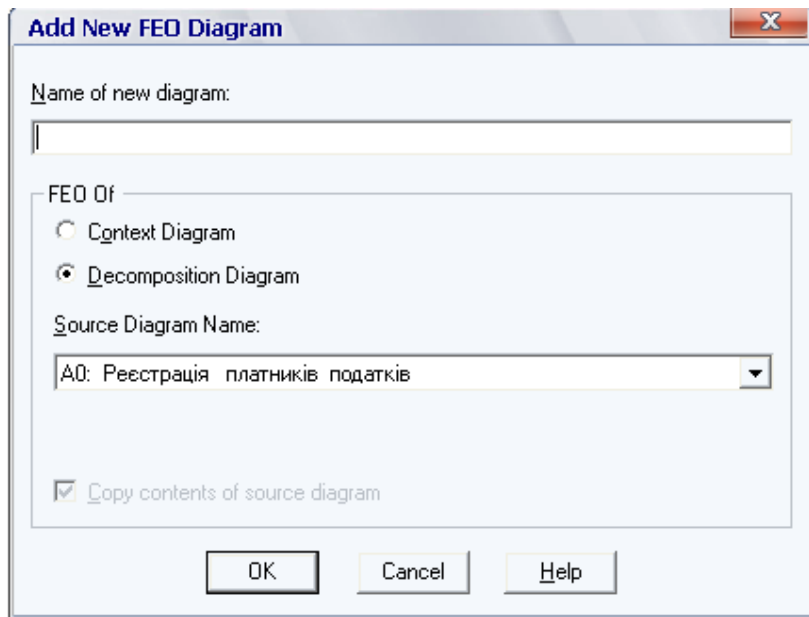


Рисунок 3.5 – Діалогове вікно створення FEO-діаграми

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler з використанням стандарту функціонального моделювання IDEF0 та згідно номеру варіанту (табл. 1.1). Робота є логічним продовженням попередніх лабораторних робіт.

Завдання 1. Розглянути приклади 1 та 2.

Завдання 2. Побудувати діаграму вузлів.

Завдання 3. Побудувати FEO-діаграму.

Приклад 1. Побудова діаграми вузлів на прикладі діяльності ТОВ «Буєрак» по виготовленню тютюнових виробів.

1. Відкриємо розроблену раніше модель.

2. Виберемо пункт головного меню *Diagram/Add Node Tree*.

3. У першому діалоговому вікні майстра *Node Tree Wizard* внесемо ім'я діаграми, вкажемо діаграму кореня дерева і кількість рівнів (рис. 3.6).

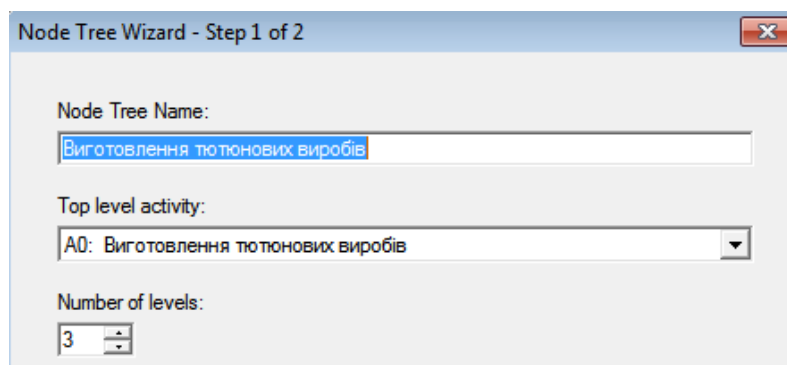


Рисунок 3.6 – Перше діалогове вікно майстра *Node Tree Wizard*

4. У другому діалоговому вікні майстра *Node Tree Wizard* встановлюємо опції, як вказано на рис. 3.7, і натиснемо кнопку *OK*.

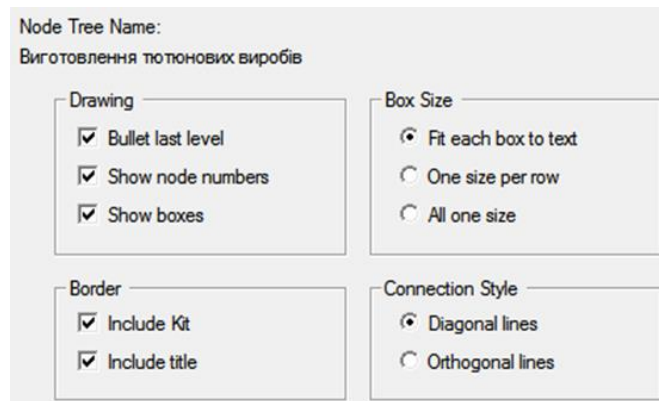


Рисунок 3.7 – Друге діалогове вікно гіда *Node Tree Wizard*

5. У результаті буде створена діаграма дерева вузлів (рис. 3.8).

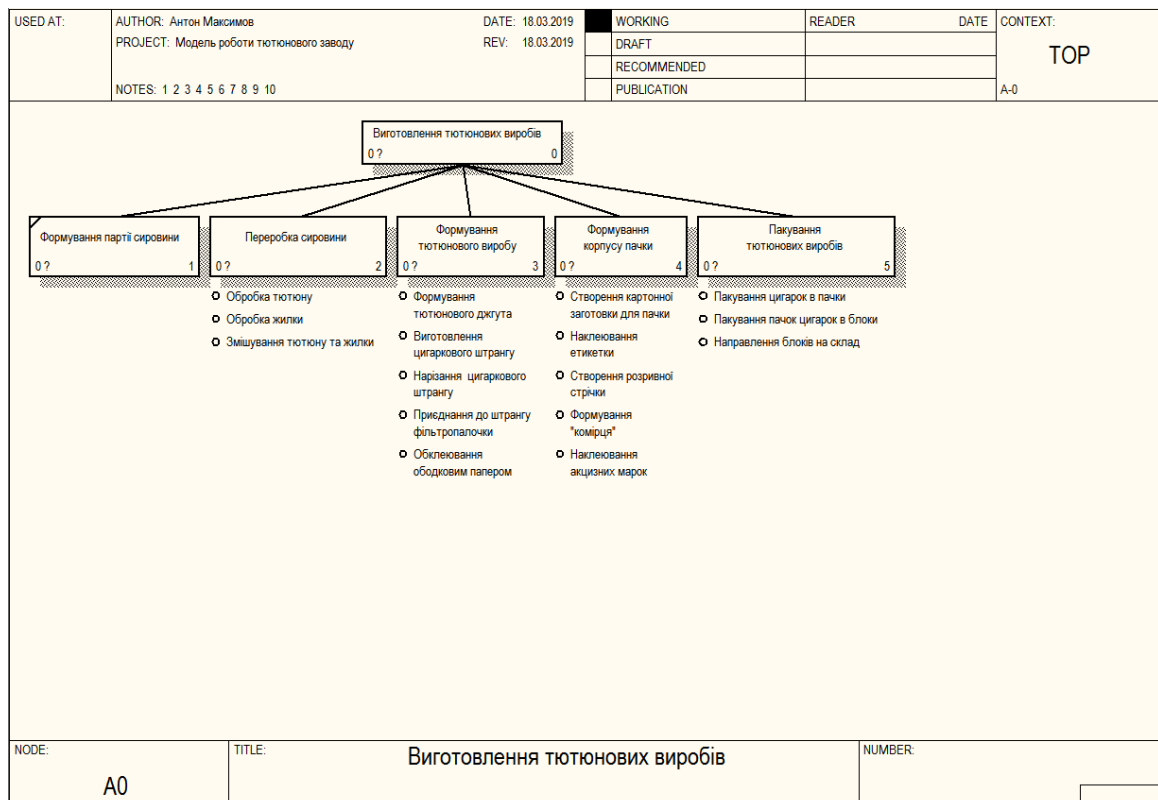


Рисунок 3.8 – Діаграма дерева вузлів

6. Діаграму дерева вузлів можна модифікувати. Нижній рівень може бути відображений не у вигляді списку, а у вигляді прямокутників, так само як і верхні рівні. Для модифікації діаграми правою кнопкою миші натиснемо по вільному місці, не зайнятому об'єктами, виберемо меню *Node tree Diagram Properties* і на вкладці *Style* діалогу *Node Tree Properties* відключимо опцію *Orthogonal lines*. Результат модифікації діаграми дерева вузлів показано на рис. 3.9.

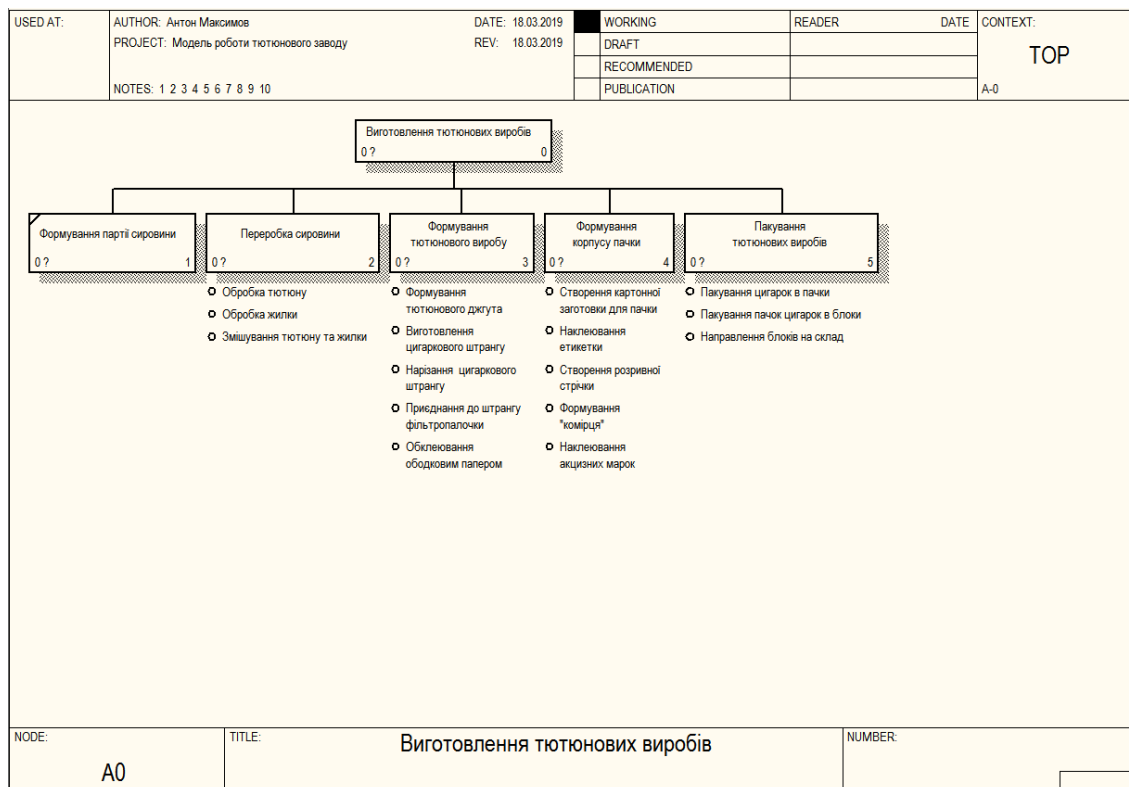


Рисунок 3.9 – Дерево вузлів після внесення змін

Приклад 2. Побудова FEO-діаграм на прикладі діяльності ТОВ «Буєрак» по виготовленню тютюнових виробів (побудова контекстної діаграми розглянута в лабораторній роботі № 1).


Припустимо, що при обговоренні бізнес-процесів виникла необхідність детально розглянути взаємодію роботи «Формування тютюнового виробу» з іншими роботами. Щоб не псувати діаграму декомпозиції, можна створити FEO-діаграму, на якій будуть тільки стрілки роботи «Формування тютюнового виробу».

1. Виберемо пункт головного меню *Diagram/Add FEO Diagram*.

2. У діалоговому вікні *Add New FEO Diagram* виберемо тип і внесемо ім'я діаграми FEO, як вказано на рис. 3.10.

3. Для визначення змісту діаграми перейдемо у пункт меню *Diagram/Diagram Properties* і на вкладці *Diagram Text* внесемо визначення, як вказано на рис. 3.11.

4. Видалимо зайві стрілки на діаграмі FEO (рис. 3.12).

5. Для переходу між стандартною діаграмою, деревом вузлів і FEO-діаграмою можна використовувати кнопку  на панелі інструментів.

6. Результат відображення FEO-діаграми представлено на рис. 3.12.

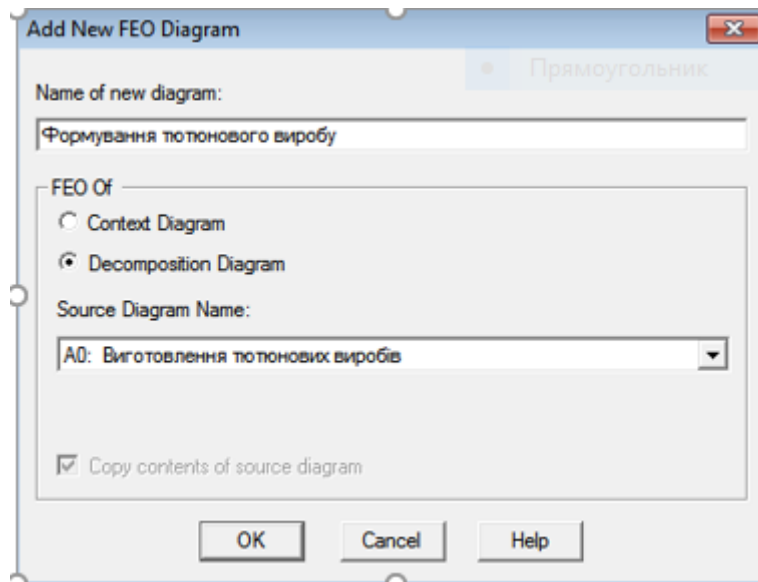


Рисунок 3.10 – Діалогове вікно *Add New FEO Diagram*

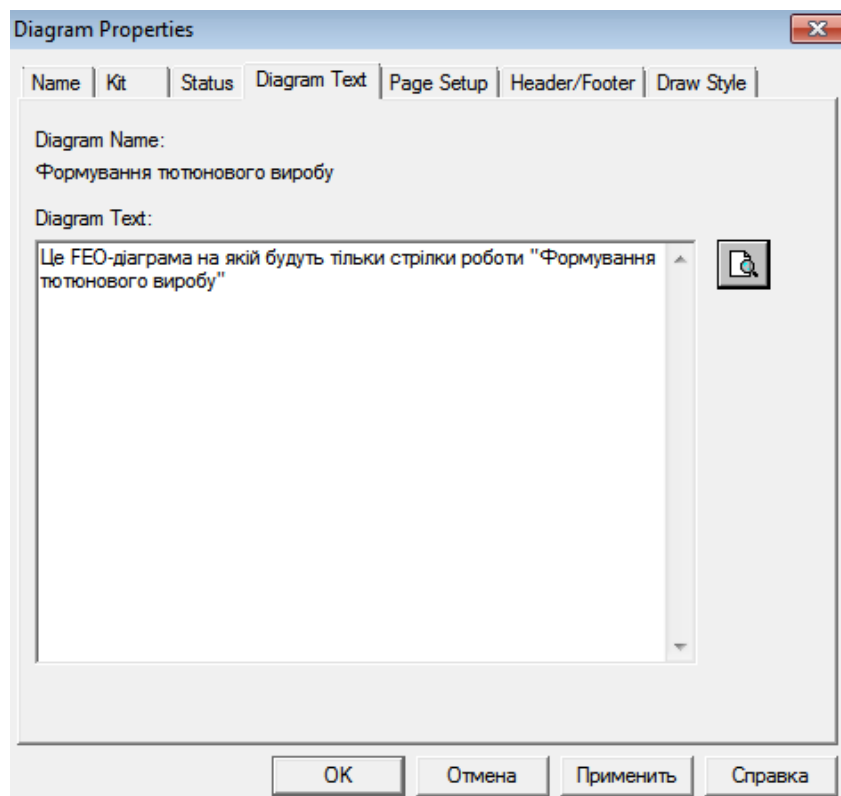


Рисунок 3.11 – Вкладка *Diagram Text* діалогового вікна *Diagram Properties*

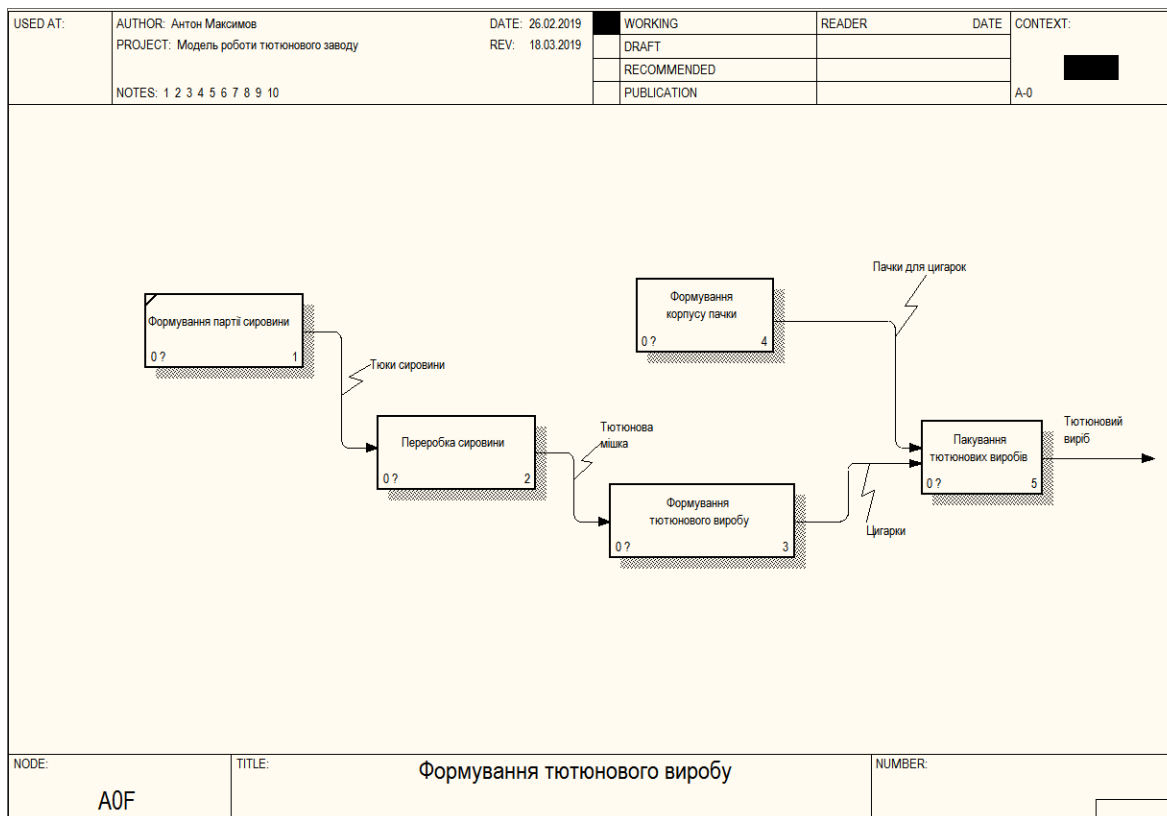


Рисунок 3.12 – Відображення FEO-діаграми

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №2.
5. Протокол розв'язання завдання №3.
6. Висновки.

Контрольні питання

1. Дайте визначення діаграми дерева вузлів та вкажіть їх застосування.
2. Дайте визначення FEO-діаграми та вкажіть їх застосування.
3. Назвіть різницю між контекстною і FEO-діаграмою.
4. Поясніть процес відображення декомпозованої роботи на діаграмі.
5. Поясніть формування номеру діаграми дерева вузлів та FEO-діаграми.
6. Чи поширюються правила IDEF0 на діаграму дерева вузлів?
7. Чи поширюються правила IDEF0 на FEO-діаграму?
8. Яка різниця між моделлю-джерелом і моделлю-метою?
9. Як відбувається тунелювання стрілки?

ЛАБОРАТОРНА РОБОТА №4

Злиття і розщеплення моделей

Мета роботи: Набути практичних навичок застосування інструментів засобу AllFusion Process Modeler при виконанні процесу розщеплення і злиття створеної функціональної моделі.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

Можливість злиття і розщеплення моделей забезпечує колективну роботу над проектом. Так, керівник проекту може створити декомпозицію верхнього рівня і дати завдання аналітикам продовжити декомпозицію кожної гілки дерева у вигляді окремих моделей. Після закінчення роботи над окремими гілками всі підмоделі можуть злитися в єдину модель. З іншого боку, окрема гілка моделі може бути відокремлена для використання як незалежна модель, для доопрацювання або архівації.

AFPM використовує для злиття і розщеплення моделей стрілки виклику.

Для злиття моделей необхідно виконати такі умови:

- обидві моделі повинні бути відкриті в AFPM;
- ім'я моделі-джерела, яку приєднують до основної моделі (моделі-мети), повинно зберігатися з ім'ям стрілки виклику роботи в основній моделі (рис. 4.1);

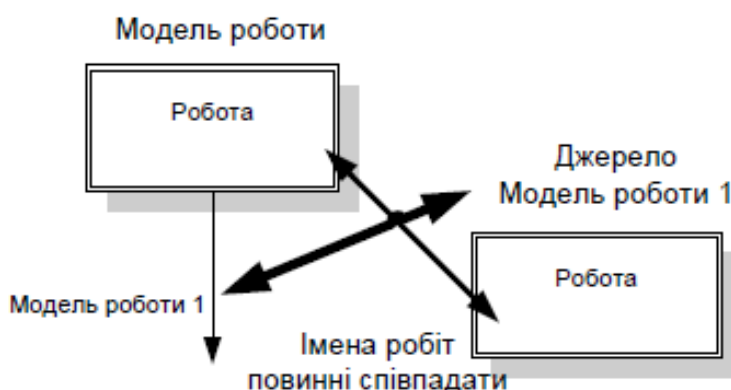


Рисунок 4.1 – Умови злиття моделей

- стрілка виклику повинна виходити з недекомпозованої роботи (робота повинна мати діагональну косу в лівому верхньому куті) (рис. 4.2);

- імена контекстної роботи, що додається (моделі-джерела) і роботи на основній моделі, до якої ми приєднуємо модель-джерело, повинні зберігатися (рис. 4.1);
- модель-джерело повинна містити принаймні одну діаграму декомпозиції.



Рисунок 4.2 – Стрілка виклику роботи «Збір і тестування комп'ютерів» в основній моделі

Для злиття моделей потрібно натиснути правою кнопкою миші на роботі із стрілкою виклику в основній моделі і в спливаючому меню вибрати пункт *Merge Model*.

З'являється діалогове вікно, в якому слід вказати опції злиття моделі. Під час злиття моделей об'єднуються словники стрілок і робіт. У разі однакових визначень можливий перезапис визначень або додавання визначень з моделі-джерела.

Після підтвердження злиття модель-джерело приєднується до моделі-мети, стрілка виклику зникає, а робота, від якої відходила стрілка виклику, стає декомпозованою – до неї приєднується діаграма декомпозиції першого рівня моделі-джерела. Стрілки, що стосуються роботи на діаграмі моделі-мети, автоматично не мігрують у декомпозицію, а відображаються як недозволені. Їх потрібно *тунелювати вручну*.

На рис. 4.3 показано, як виглядають моделі у вікні *Model Explorer* після злиття.

У процесі злиття модель-джерело залишається незмінною і до моделі-мети підключається фактично її копія (якщо вибрано опцію у вікні злиття *Cut/Paste entire Dictionaries*). Не потрібно плутати злиття моделей із синхронізацією. Якщо надалі модель-джерело редагуватиметься, ці зміни автоматично не потраплять у відповідну гілку основної моделі.

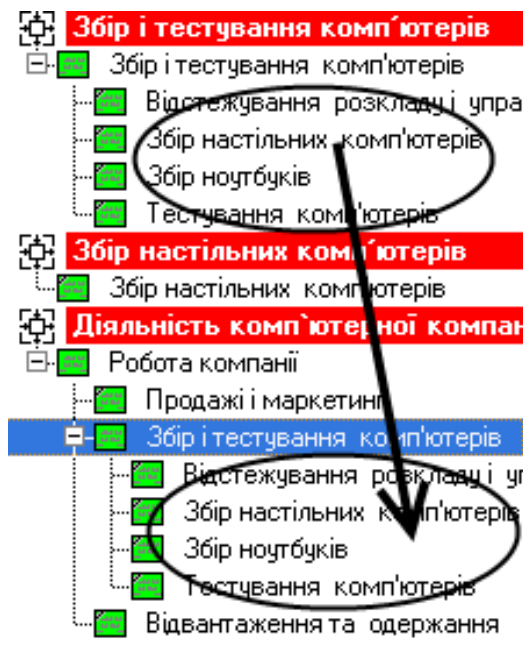


Рисунок 4.3 – Вид моделей у *Model Explorer* після злиття. Виділена модель-джерело і приєднана гілка основної моделі

Розщеплення моделей проводиться аналогічно.

Для розщеплення гілки від моделі слід натиснути правою кнопкою миші по декомпованій роботі (*ознака – робота не повинна мати діагональної коси в лівому верхньому куті*) і вибрати у спливаючому меню пункт *Split Model*.

У вікні діалогу *Split Options*, що з'явився, слід вказати ім'я створюваної моделі. Після підтвердження розщеплення у старій моделі робота стане недекомпованою (*ознака – діагональна коса в лівому верхньому куті*), буде створена стрілка виклику, причому її ім'я збігатиметься з ім'ям нової моделі та буде створена нова модель, причому ім'я контекстної роботи збігатиметься з ім'ям роботи, від якої була «відірвана» декомпозиція.

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler з використанням стандарту функціонального моделювання IDEF0 та згідно номеру варіанту (табл. 1.1). Робота є логічним продовженням попередніх лабораторних робіт.

Завдання 1. Розглянути приклади 1 та 2.

Завдання 2. Виконати розщеплення моделей.

Завдання 3. Виконати злиття моделей.

Приклад 1. Виконання розщеплення моделей на прикладі діяльності ТОВ «Буєрак» по виготовленню тютюнових виробів.

1. Перейдемо на діаграму A0. Правою кнопкою миші натиснемо на роботі «Формування тютюнового виробу» та виберемо у спливаючому меню пункт *Split model* (рис. 4.4).

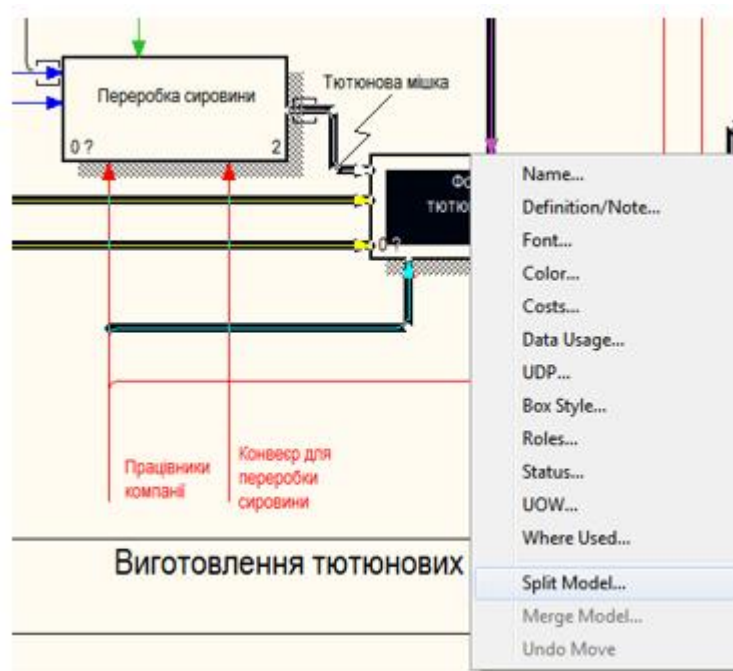


Рисунок 4.4 – Пункт контекстного меню *Split Model*

2. У діалоговому вікні *Split Option* (опції розділення) вносимо ім'я нової моделі «*Формування тютюнового виробу*» та встановимо опцію *Copy entire dictionaries* (копіювати повністю словник) як на рис. 4.5.

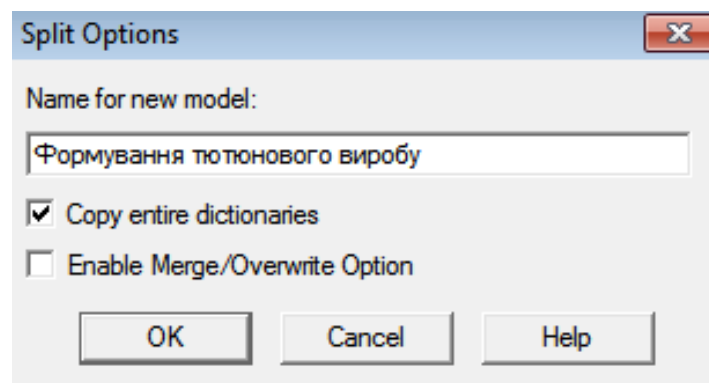


Рисунок 4.5 – Діалогове вікно *Split Option*

3. У Model Explorer з'явилася нова модель (рис. 4.6), а на діаграмі A0 моделі «*Виготовлення тютюнових виробів*» з'явилася стрілка виклику «*Формування тютюнового виробу*» (рис. 4.7).

4. Створимо у моделі «*Формування тютюнового виробу*» нову стрілку «*Виробничий брак*». На діаграмі A0 це буде гранична стрілка виходу від робіт «*Формування тютюнового джгута*», «*Виготовлення цигаркового штрангу*», «*Нарізання цигаркового штрангу*», «*Приєднання до штрангу фільтрпалочки*» та «*Обклеювання ободковим папером*» (рис. 4.8).

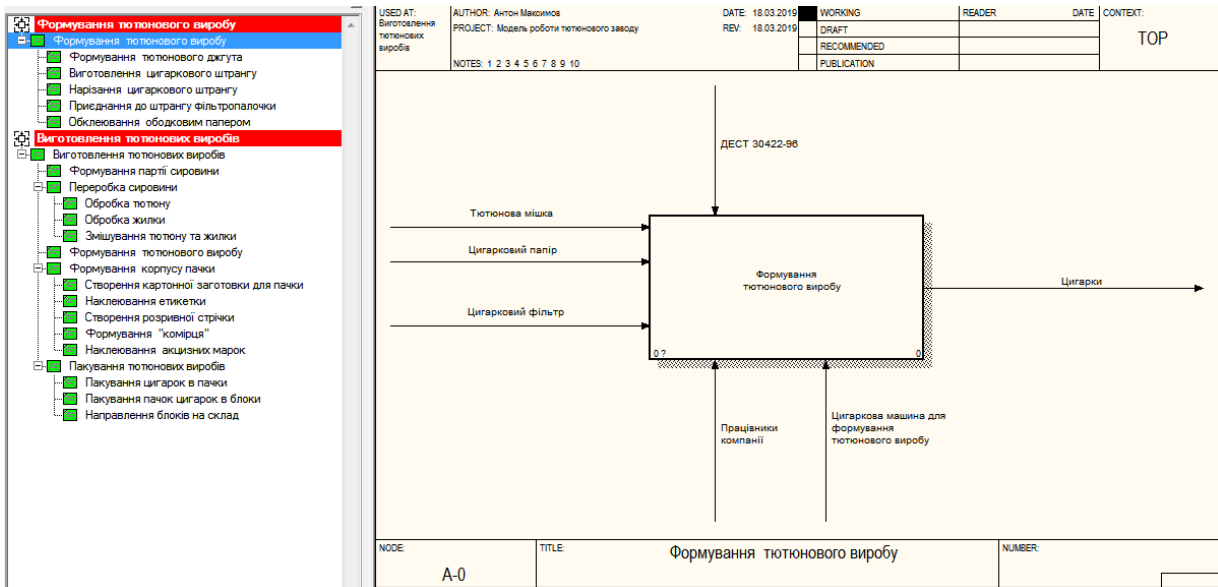


Рисунок 4.6 – Відображення нової моделі «Формування тютюнового виробу» у Model Explorer

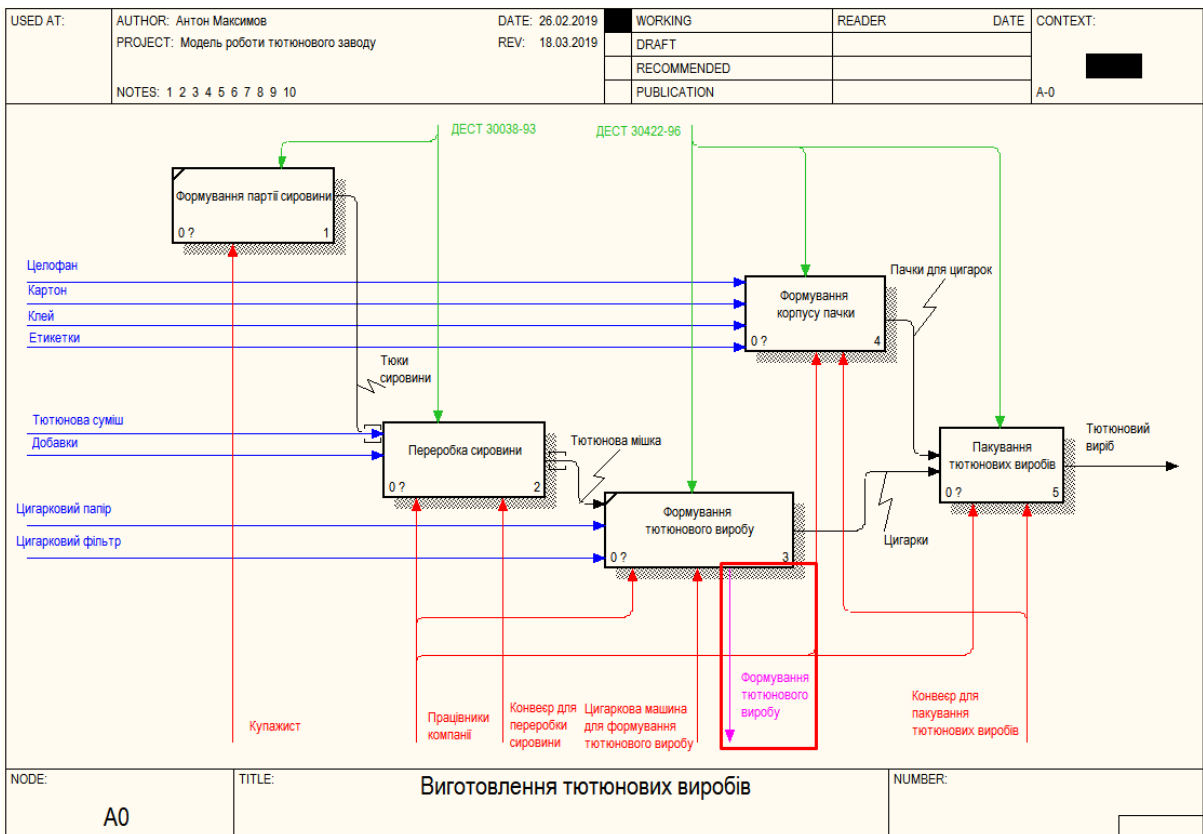


Рисунок 4.7 – Діаграма A0 з «Виготовлення тютюнових виробів» стрілкою виклику «Формування тютюнового виробу»

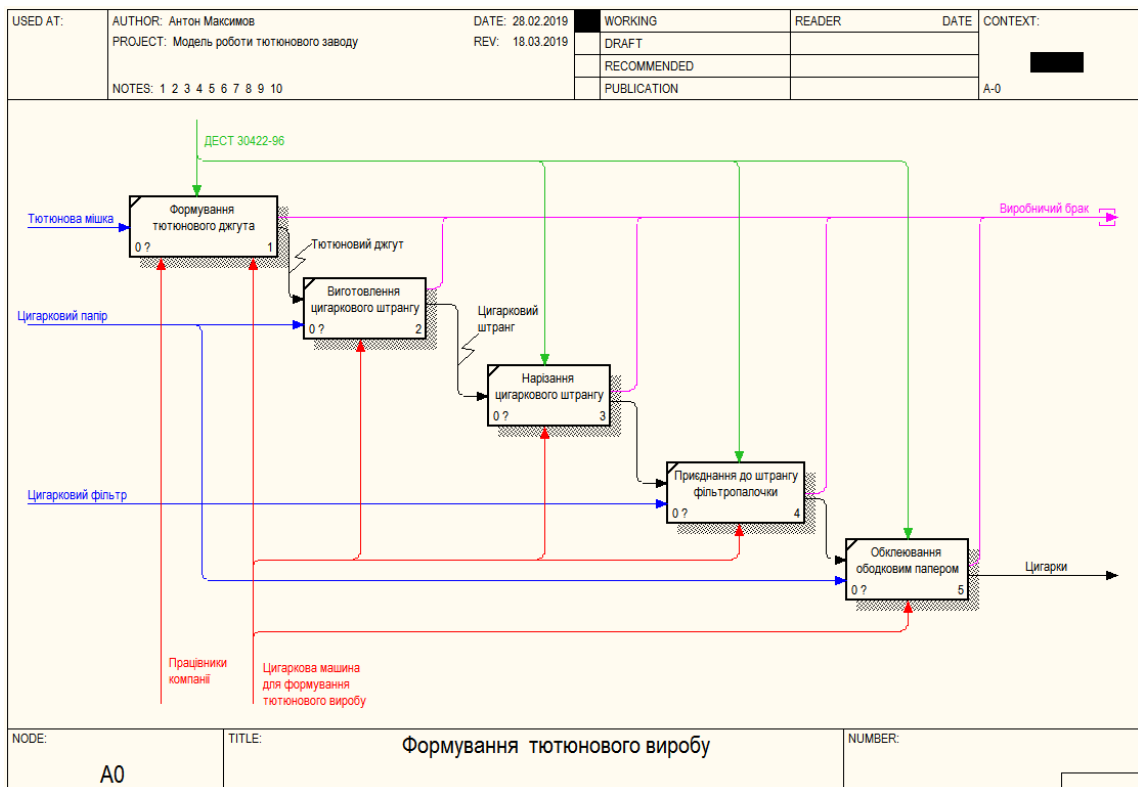


Рисунок 4.8 – Діаграма А0 з граничною стрілкою виходу від вказаних робіт «Виробничий брак»

Приклад 2. Виконати злиття моделей на прикладі діяльності ТОВ «Буєрак» по виготовленню тютюнових виробів.

1. Перейдемо на діаграму А0 моделі «Формування тютюнового виробу».
2. Правою кнопкою миші натиснемо на роботі «Формування тютюнового виробу» та виберемо у контекстному меню опцію *Merge model* (рис. 4.9).
3. У діалоговому вікні *Merge Model* виберемо опцію *Cut/Paste entire dictionaries* (рис. 4.10).
4. Звернемо увагу на результат. У Model Explorer помітно, що дві моделі злилися (рис. 4.11).
5. Модель «Формування тютюнового виробу» залишилася і може бути збережена в окремому файлі. На діаграмі А0 моделі «Виготовлення тютюнових виробів» зникла стрілка виклику «Формування тютюнового виробу» (рис. 4.12).
6. З'явилася недозволена гранична стрілка «Несправні компоненти». Направимо цю стрілку на вихід, як показано на рис. 4.13.

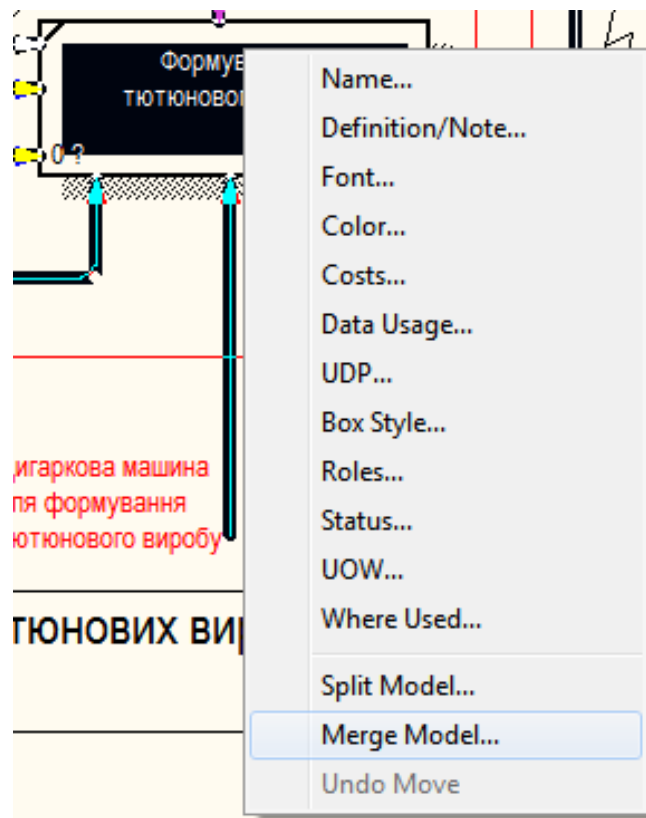


Рисунок 4.9 – Пункт контекстного меню *Merge model*

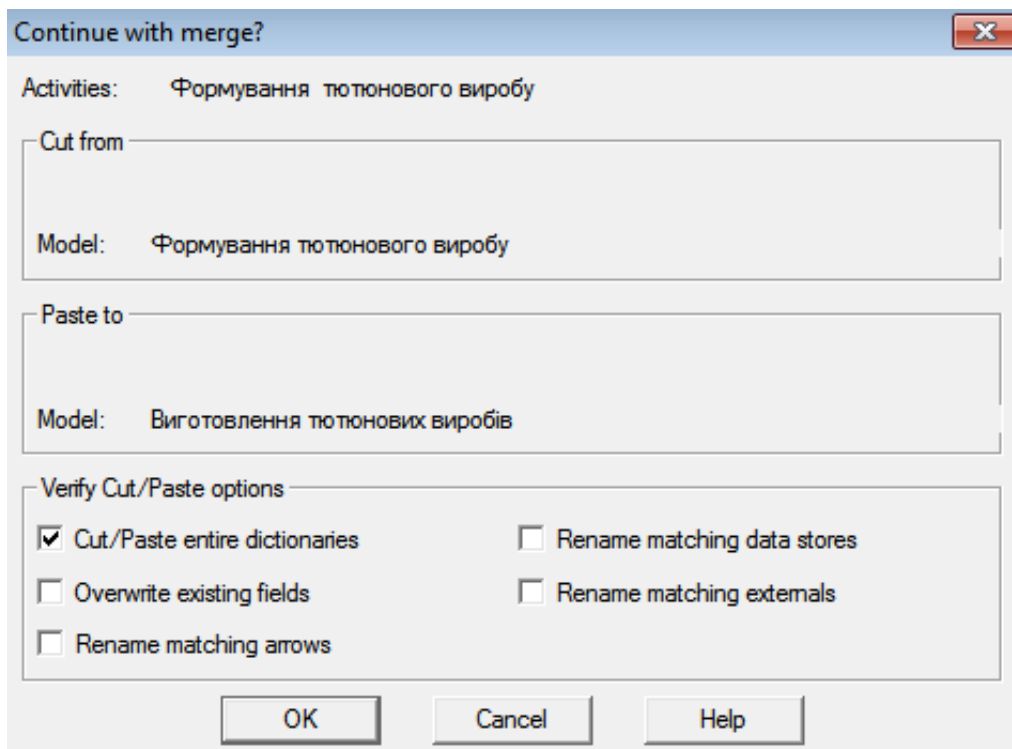


Рисунок 4.10 – Включення опції *Cut/Paste entire dictionaries*

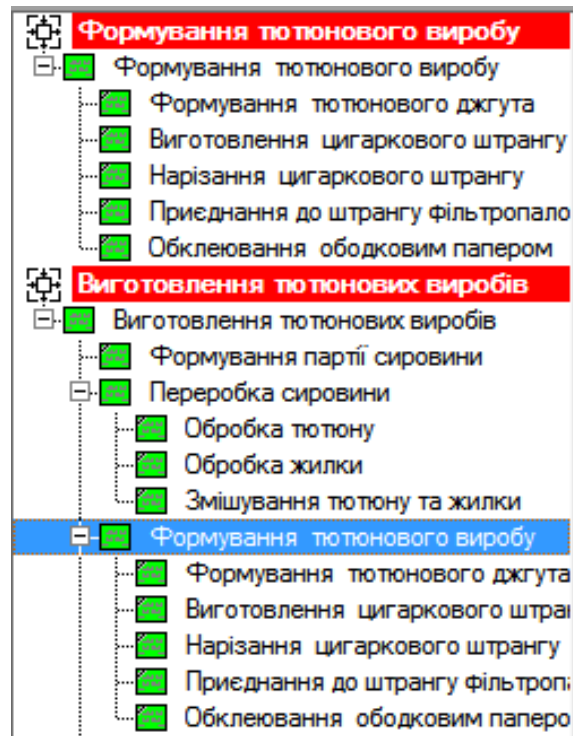


Рисунок 4.11 – Злиття моделей «Виготовлення тютюнових виробів» і «Формування тютюнового виробу»

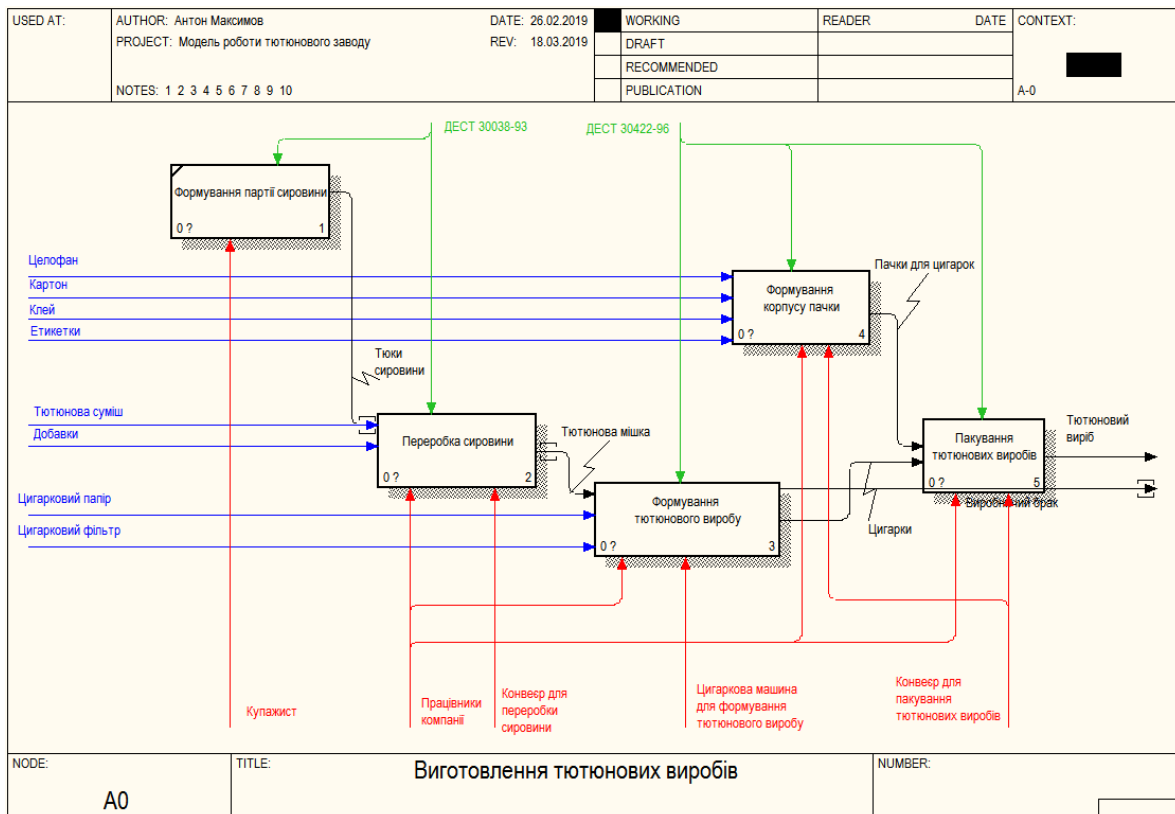


Рисунок 4.12 – Відображення на діаграмі A0 зникнення стрілки виклику «Виробничий брак»

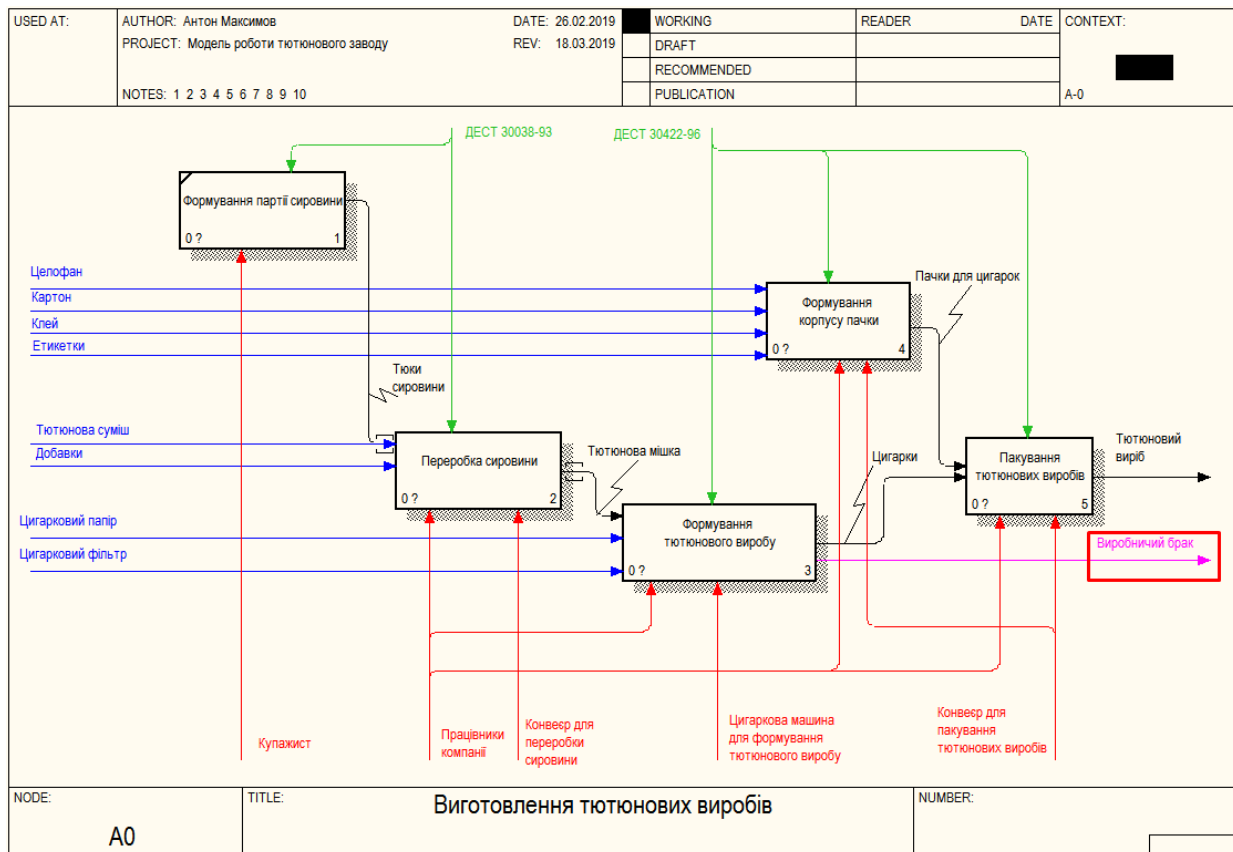


Рисунок 4.13 – Відображення на діаграмі А0 стрілки «Виробничий брак» на вихід

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №1.
5. Протокол розв'язання завдання №2.
6. Висновки.

Контрольні питання

1. Які умови злиття моделей?
2. Які умови розділення моделей?
3. Поясніть процес злиття моделей.
4. Поясніть процес розділення моделей.

ЛАБОРАТОРНА РОБОТА №5

Створення діаграм та сценарію з використанням стандарту функціонального моделювання IDEF0

Мета роботи: Ознайомитися з методологією нотації IDEF3. Набути практичних навичок застосування інструментів засобу AllFusion Process Modeler для створення діаграми IDEF3 та діаграми сценарію для попередньо розробленої функціональної моделі.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Метод опису процесів IDEF3

Наявність в діаграмах DFD елементів для опису джерел, приймачів і сховищ даних дозволяє ефективніше і наочно описати процес документообігу. Проте для опису логіки взаємодії інформаційних потоків більш підходить IDEF3, що звана також *workflow diagramming* методологією моделювання. Вона використовує графічний опис інформаційних потоків, взаємодію між процесами обробки інформації і об'єктів, що є частиною цих процесів.

Діаграми Workflow можуть бути використані в моделюванні бізнес-процесів для аналізу завершеності процедур обробки інформації. З їх допомогою можна описувати сценарії дій співробітників організації, наприклад послідовність обробки замовлення або події, які необхідно обробити за кінцевий час. Кожен сценарій супроводжується описом процесу і може бути використаний для документування кожної функції.

IDEF3 – це метод, що має на основній меті дати можливість аналітикам описати ситуацію, коли процеси виконуються в певній послідовності, а також описати об'єкти, що беруть участь сумісно в одному процесі.

Техніка опису набору даних IDEF3 є частиною структурного аналізу. На відміну від деяких методик описів процесів IDEF3 не обмежує аналітика надмірно жорсткими рамками синтаксису, що може привести до створення неповних або суперечливих моделей.

IDEF3 може бути також використаний як метод створення процесів. Він доповнює IDEF0 і містить все необхідне для побудови моделей, які надалі можуть бути використані для імітаційного аналізу.

Кожна робота в IDEF3 описує який-небудь сценарій бізнес-процесу і може бути складовою іншої роботи. Оскільки сценарій описує мета і рамки моделі,

важливо, щоб роботи іменувалися віддієслівним іменником, що позначають процес дії, або фразою, що містить такий іменник.

Діаграми. Діаграма є основною одиницею опису в IDEF3. Важливо правильно побудувати діаграми, оскільки вони призначені для читання іншими людьми (а не тільки автором).

Одиниці роботи – Unit of Work (UOW). UOW, також звані роботами (activity), є центральними компонентами моделі. В IDEF3 роботи зображуються прямокутниками з прямими кутами і мають ім'я, виражене віддієслівним іменником, що позначає процес дії, одиночним або у складі фрази, і номер (ідентифікатор); інший іменник у складі тієї ж фрази зазвичай відображає основний вихід (результат) роботи, наприклад, «Виготовлення виробу». Ідентифікатор роботи привласнюється при створенні і не міняється ніколи. Навіть якщо робота буде видалена, її ідентифікатор знов не використовуватиметься для інших робіт. Зазвичай номер роботи складається з номера батьківської роботи і порядкового номера на поточній діаграмі.

Зв'язки. Зв'язки показують взаємини робіт. Всі зв'язки в IDEF3 однонаправлені і можуть бути направлені куди завгодно, але зазвичай діаграми IDEF3 прагнуть побудувати так, щоб зв'язки були направлені зліва направо. У IDEF3 розрізняють три типи стрілок, що зображають зв'язки, стиль яких встановлюється через меню *Edit/Arrow Style*:

Старша (Precedence) – суцільніша лінія, що зв'язує одиниці робіт (UOW), Малюється зліва направо або зверху вниз. Показує, що робота-джерело повинна закінчитися перш, ніж робота-мета почнеться.

Відносини (Relational Link) – пунктирна лінія, що використовується для зображення зв'язків між одиницями робіт (UOW) а також між одиницями робіт і об'єктами посилань.

Потоки об'єктів (Object Flow) – стрілка з двома наконечниками, застосовується для опису того факту, що об'єкт використовується в двох або більш одиницях роботи, наприклад, коли об'єкт породжується в одній роботі і використовується в іншій.

Старший зв'язок і потік об'єктів. Старший зв'язок показує, що робота-джерело закінчується раніше, чим починається робота-мета. Часто результатом роботи-джерела стає об'єкт, необхідний для запуску роботи-мети. В цьому випадку стрілку, що позначає об'єкт, зображають з подвійним наконечником. Ім'я стрілки повинне ясно ідентифікувати об'єкт, що відображається. Потік об'єктів має ту ж семантику, що і старша стрілка.

Відношення показує, що стрілка є альтернативою старшій стрілці або потоку об'єктів в сенсі завдання послідовності виконання робіт: робота-джерело не обов'язково повинна закінчитися, перш ніж робота-мета почнеться. Робота-мета може закінчитися перш, ніж закінчиться робота-джерело.

Перехрестя (Junction). Закінчення однієї роботи може служити сигналом до початку декількох робіт, або ж одна робота для свого запуску може чекати закінчення декількох робіт. Перехрестя використовуються для відображення логіки взаємодії стрілок при злитті і розгалуженні або для відображення

множини подій, які можуть або повинні бути завершені перед початком наступної роботи. Розрізняють перехрестя для злиття (*Fan-in Junction*) і розгалуження (*Fan-out Junction*) стрілок. Перехрестя не може використовуватися одночасно для злиття і для розгалуження. Для внесення перехрестя служить кнопка в палітрі інструментів – додати в діаграму перехрестя *Junction*. У діалозі *Junction Type Editor* необхідно вказати тип перехрестя.

Об'єкт посилання. Об'єкт посилання в IDEF3 виражає якусь ідею, концепцію або дані, які не можна пов'язати із стрілкою, перехрестям або роботою. Для внесення об'єкту посилання служить кнопка (додати в діаграму об'єкт посилання – *Referent*) в палітрі інструментів. Об'єкт посилання зображається у вигляді прямокутника, схожого на прямокутник роботи. Ім'я об'єкту посилання задається в діалозі *Referent* (пункт спливаючого меню *Name Editor*), як ім'я можна використовувати ім'я якої-небудь стрілки з інших діаграм або ім'я суті з моделі даних. Об'єкти посилання повинні бути пов'язані з одиницями робіт або перехрестями пунктирними лініями. Офіційна специфікація IDEF3 розрізняє три стилі об'єктів посилань – безумовні (*unconditional*), синхронні (*synchronous*) і асинхронні (*asynchronous*).

При внесенні об'єктів посилань крім імені слід указувати тип об'єкту посилання. Типи об'єктів посилань приведені в табл. 5.1.

Таблиця 5.1 – Типи об'єктів посилань

Тип об'єкту посилання	Мета опису
ОБ'ЄКТ	Описує участь важливого об'єкту в роботі.
GOTO	Інструмент циклічного переходу (у послідовності робіт, що повторюється), можливо на поточній діаграмі, але не обов'язково. Якщо всі роботи циклу присутні на поточній діаграмі, цикл може також зображуватися стрілкою, що повертається на стартову роботу. GOTO може посилатися на перехрестя.
UOB (Unit of behavior)	Застосуються, коли необхідно підкреслити множинне використання якої-небудь роботи, але без циклу. Наприклад, робота «Контроль якості» може бути використаний в процесі «Виготовлення виробу» кілька разів, після кожною одиницею операції. Зазвичай цей тип посилання не використовується для моделювання робіт, що автоматично запускаються.
NOTE	Використовується для документування важливої інформації, що відноситься до яких-небудь графічних об'єктів на діаграмі. NOTE є альтернативою внесенню текстового об'єкту до діаграми.
ELAB (Elaboration)	Використовується для удосконалення графіків або їх детальнішого опису. Зазвичай вживається для детального опису розгалуження і злиття стрілок на перехрестях.

Декомпозиція робіт. У IDEF3 декомпозиція використовується для деталізації робіт. Методологія IDEF3 дозволяє декомпонувати роботу багато разів, тобто робота може мати безліч дочірніх робіт. Це дозволяє в одній моделі описати альтернативні потоки. Можливість множинної декомпозиції пред'являє додаткові вимоги до нумерації робіт.

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler з використанням стандарту функціонального моделювання IDEF0 та згідно номеру варіанту (табл. 1.1).

Завдання 1. Створити діаграму IDEF3.

Завдання 2. Створити діаграму сценарію.

Приклад 1. Створення діаграми IDEF3 на прикладі діяльності комп'ютерної компанії.

1. Перейдемо на діаграму A2 і виконаємо декомпозицію роботи «Збірка настільних комп'ютерів». У діалозі *Activity Box Count* (рис. 5.1) встановимо число робіт – 4 та нотацію IDEF3.

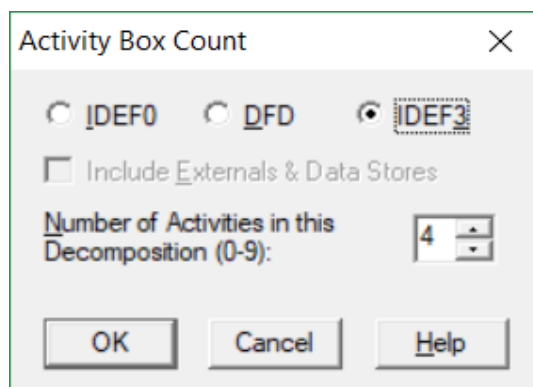


Рисунок 5.1 – Вибір нотації IDEF3 в діалозі *Activity Box Count*

Появляється діаграма IDEF3, що містить роботи (UOW). Правою кнопкою миші натиснемо по роботі, виберемо у контекстному меню *Name* і вносимо ім'я роботи «Підготовка компонентів». Потім у вкладці *Definition* вносимо визначення «Готуються всі компоненти комп'ютера згідно специфікації замовлення».

2. У вкладці *UOW* вносимо властивості роботи (табл. 5.2).

Таблиця 5.2 – Властивості UOW

Objects	Компоненти: вінчестери, корпус, материнські плати, відео карти, звукові карти, дисководи CD-ROM та флопі, модеми, програмне забезпечення.
Facts	Доступні операційні системи: Windows 98, Windows NT, Windows 2000.
Constrains	Встановлення модему передбачає встановлення додаткового програмного забезпечення.

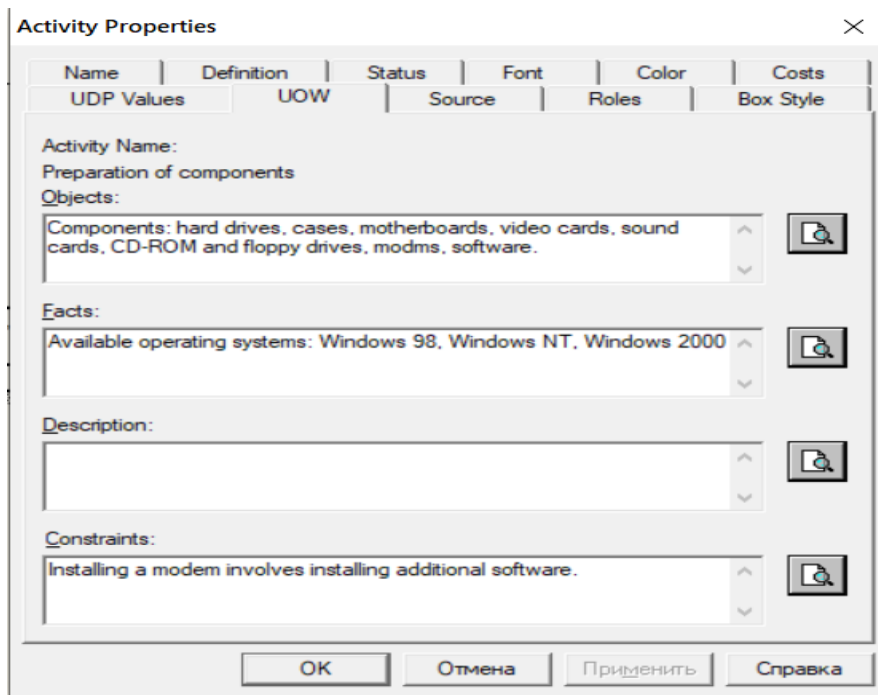


Рисунок 5.2 – Внесення властивостей роботи UOW

3. Вносимо в діаграму ще 3 роботи (кнопка).

4. Вносимо імена робіт:

- Установка материнської плати і вінчестера;
- Встановлення модему;
- Установка дисководу CD-ROM;
- Установка флопі-дисководу;
- Інсталяція операційної системи;
- Інсталяція додаткового програмного забезпечення.

5. За допомогою кнопки *Палітри інструментів* створимо об'єкт посилання. Вносимо ім'я об'єкта зовнішнього посилання «Компоненти». Зв'яжемо стрілкою об'єкт посилання і роботу «Підготовка компонентів».

6. Зв'яжемо стрілкою роботи «Підготовка компонентів» (вихід) та «Установка материнської плати і вінчестера». Змінюємо стиль стрілки на *Object Flow*.

7. У IDEF3 ім'я стрілки може бути відсутнім, хоча AllFusion Process Modeler r7 показує відсутність імені як помилку. Результат виконання подано на рис. 5.3.

8. Правою кнопкою натиснемо по перехрестю для розгалуження (*fan-out*), вибравши *Name* вносимо ім'я «Компоненти, необхідні в специфікації замовлення». Діаграму IDEF3 після створення перехресть показано на рис. 5.4.

9. Створимо два перехрестя типу виключає «АБО» і зв'яжемо роботи, як показано на рис. 5.5.

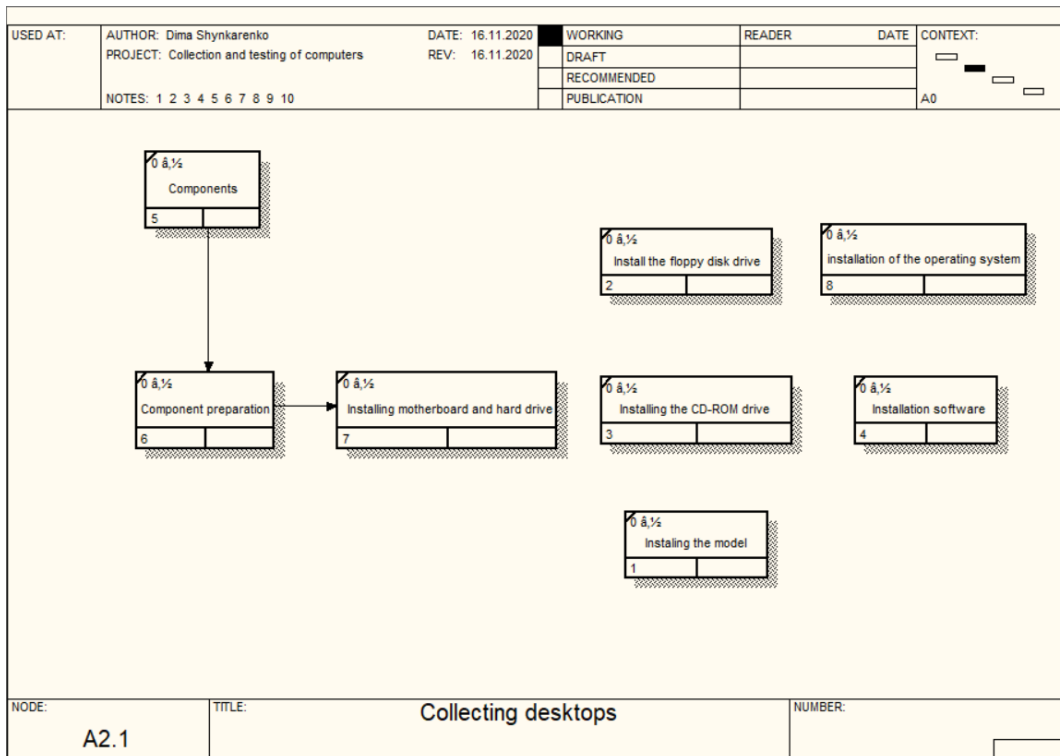


Рисунок 5.3 – Результат створення VOW і об'єкта посилання

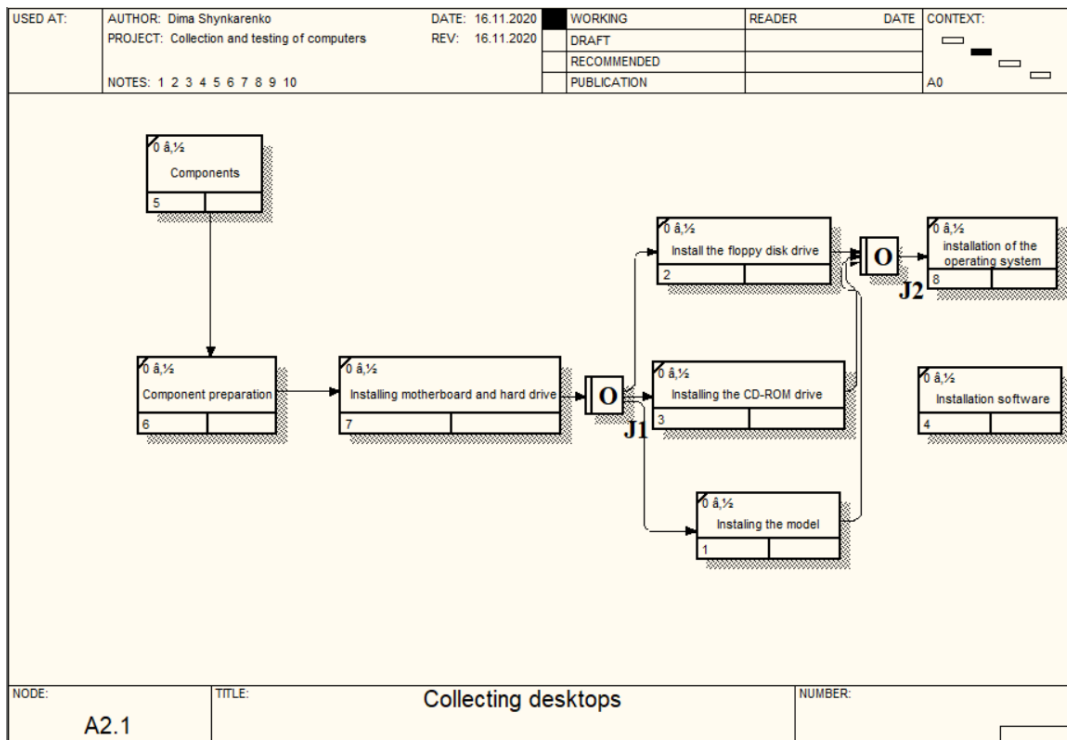


Рисунок 5.4 – Діаграма IDEF3 після створення перехресть

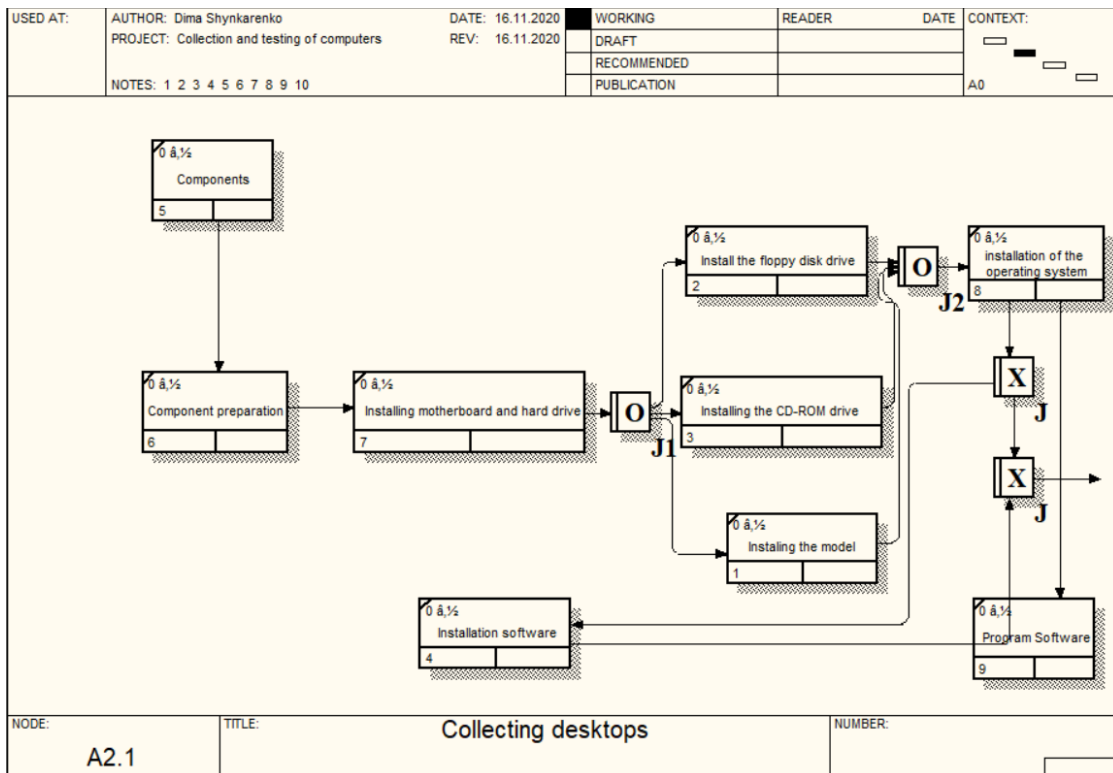


Рисунок 5.5 – Результат виконання завдання 1

Приклад 2. Створення діаграми сценарію на прикладі діяльності комп’ютерної компанії.

1. Виберемо пункт меню *Diagram / Add IDEF3 Scenario* (рис. 5.6). Створимо діаграму сценарію на основі діаграми IDEF3 «Збірка настільних комп’ютерів» (A2.1).

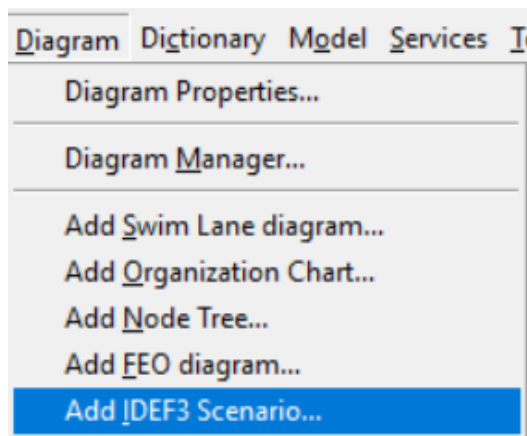


Рисунок 5.6 – Вибір пункту меню Diagram / Add IDEF3 Scenario

2. Видалимо елементи, що не входять в сценарій (рис. 5.7).
3. Результат виконання завдання 2 подано на (рис. 5.7).

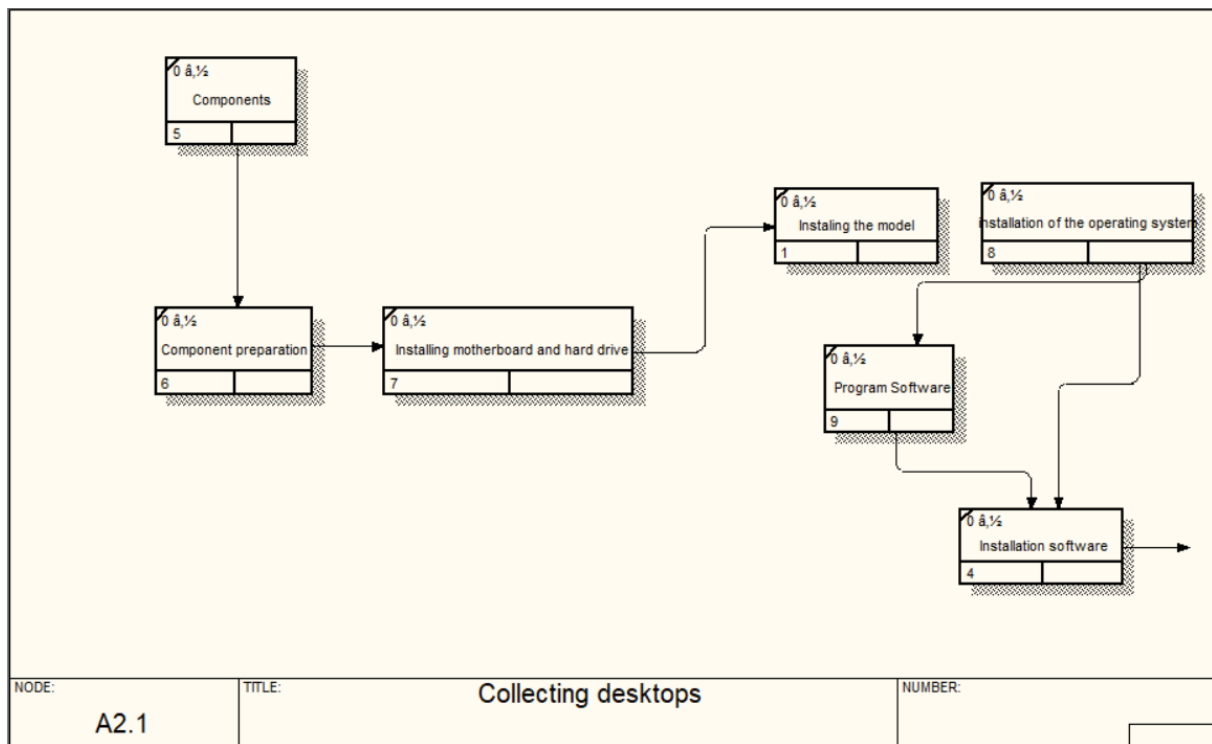


Рисунок 5.7 – Результат виконання завдання 2

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №1.
5. Протокол розв'язання завдання №2.
6. Висновки.

Контрольні питання

1. Що таке модель IDEF0?
2. Розкрийте сутність понять: «порядком домінування», «дочірня діаграма», «батьківська діаграма».
3. Нумерація робіт і діаграм.
4. Каркас діаграми.
5. Проектування екранних форм електронних документів.
6. Інформаційна база і способи її організації.
7. Що являють собою діаграми IDEF3?
8. Як здійснюється декомпозиція роботи IDEF0 або DFD у діаграму IDEF3?

ЛАБОРАТОРНА РОБОТА №6

*Вартісний аналіз (Activity Based Costing).
Використання категорій UDP.*

Мета роботи: Виконати вартісний аналіз моделі, що розробляється, в грошових та часових одиницях.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Особливості функціонально-вартісного аналізу

AllFusion Process Modeler надає аналітикові два інструменти для оцінки моделі – вартісний аналіз, заснований на роботах (Activity Based Costing, ABC), і властивості, обумовлені користувачем (User Defined Properties, UDP). Вартісний аналіз заснований на моделі робіт, тому що кількісна оцінка неможлива без детального розуміння функціональності підприємства.

Функціонально-вартісний аналіз (ФВА) ґрунтується на оцінці та аналізі об'єктів витрат за бізнес-процесами та формування центрів витрат для них. Найчастіше для функціонально-вартісного аналізу використовується аналіз собівартості без врахування прибутку, тому що його розрахунок та прогнозування для окремих етапів бізнес-процесу є достатньо важким.

Собівартість – це узагальнюючий показник, що характеризує роботу всіх видів ланок. Собівартість складається з витрат, різнорідних і по своєму складу, і по економічному призначенню, і по своїй ролі у виробництві і реалізації продукції. Всі витрати групуються в двох напрямках – по економічних елементах і по статтях витрат.

AllFusion Process Modeler надає аналітику два інструменти для оцінки такої моделі:

- 1) функціонально-вартісний аналіз, заснований на описі існуючих робіт проекту;
- 2) використання характеристик і властивостей, котрі визначаються користувачем (UDP).

ФВА-аналіз є широко поширеною методикою, яка використовується міжнародними корпораціями і державними організаціями для ідентифікації центрів витрат в організації.

Вартісний аналіз є методикою обліку витрат, пов'язаних з роботами, з метою визначення загальної собівартості процесу. Вартісний аналіз заснований

на моделі робіт, тому що кількісна оцінка неможлива без детального розуміння функціональності підприємства. За допомогою вартісного аналізу можна вирішити такі задачі, як визначення дійсної собівартості виробництва продукту, визначення вартості інвестиційного проекту, визначення найбільш витратних робіт (які повинні бути поліпшені насамперед) та ін.

ФВА може проводитися лише тоді, коли модель роботи послідовна, коректна (відображає бізнес), повна (охоплює всю дану область) і стабільна (проходить цикл експертизи без змін).

ФВА включає основні поняття:

об'єкт витрат – процес, котрий досліджується, структура формування процесів на основі використання функціонального підходу; робота, на яку провадяться витрати;

властивості витрат – характеристики входів і управління роботи, які впливають на час її виконання (наприклад, частота робіт, тривалість);

центри витрат, які можна трактувати як статті витрати по об'єкту управління (дослідження) (об'єктах витрат).

При проведенні вартісного аналізу в AllFusion Process Modeler визначаються одиниці вимірювання часу і витрат.

У діалоговому вікні *Model Properties* (викликається з меню *Mode/Model Properties*) у вкладці *ABC Units* (Рис. 6.1) встановите одиниці грошей і часу – валюти і години.

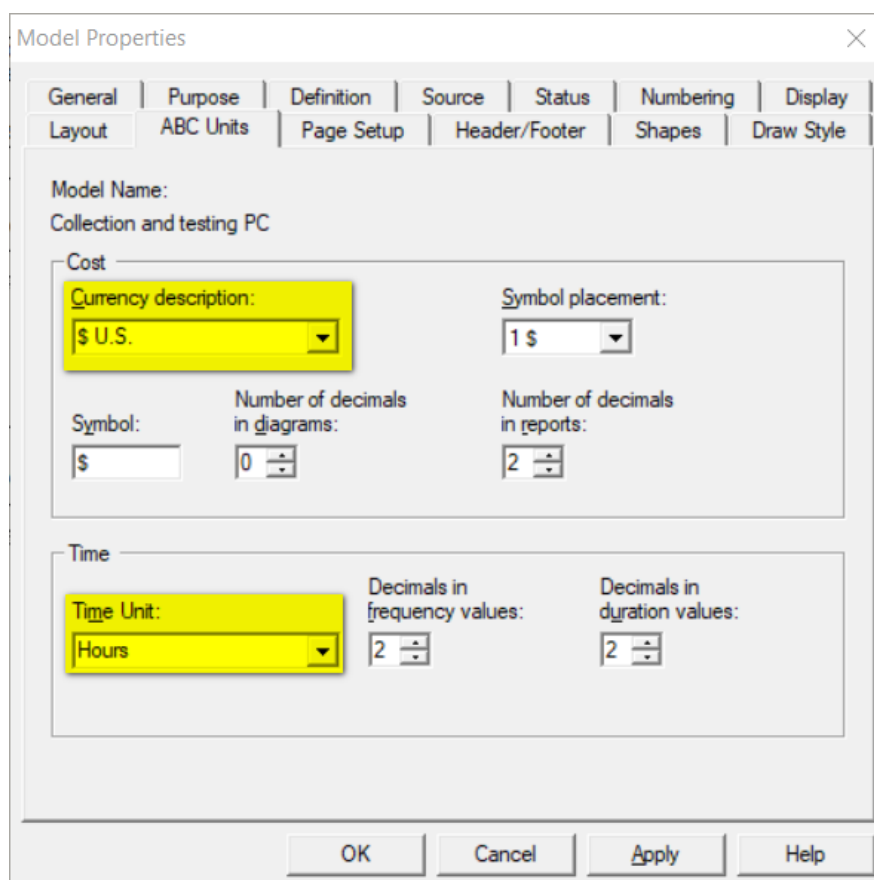


Рисунок 6.1 – Настроювання одиниць виміру валюти й часу

Якщо в списку вибору відсутня необхідна валюта (наприклад, гривня), її можна додати. Діапазон виміру часу в списку *Unit of measurment* достатній для більшості випадків – від секунд до років.

Потім описуються центри витрат (*cost centers*). Для внесення центрів витрат необхідно викликати діалог *Cost Center Editor* з меню *Model* (рис. 6.2).

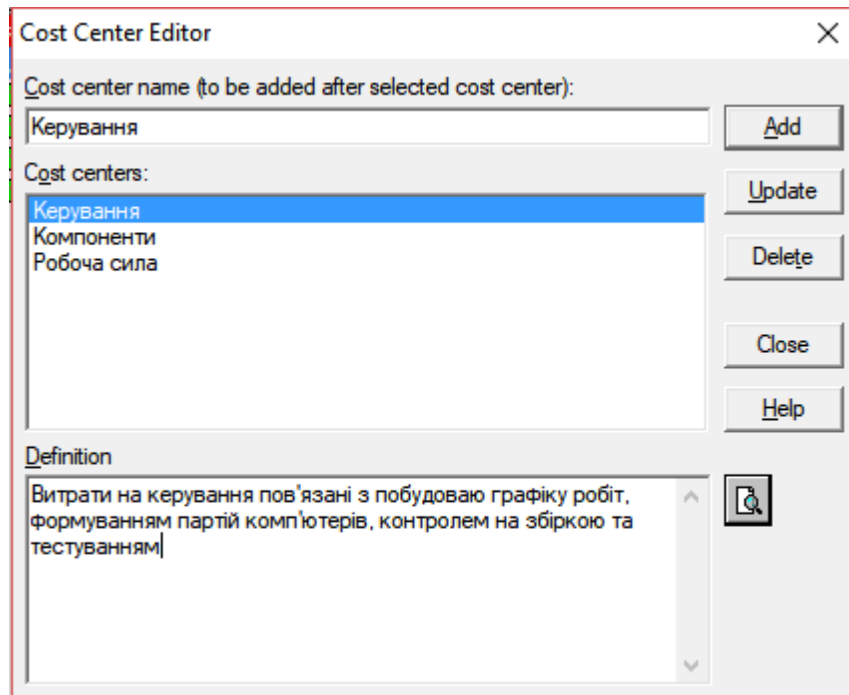


Рисунок 6.1 – Діалог Cost Center Editor

Кожному центру витрат варто дати докладний опис у вікні *Definition*. Список центрів витрат упорядкований. Порядок у списку можна міняти за допомогою стрілок, розташованих праворуч від списку.

Для завдання вартості роботи (для кожної роботи на діаграмі декомпозиції) варто клацнути правою кнопкою миші по роботі й на спливаючому меню вибрати *Cost* (рис. 6.2).

У діалозі *Activity Cost* указується частота проведення даної роботи в рамках загального процесу (вікно *Frequency*) і тривалість (*Duration*). Потім варто вибрати в списку один із центрів витрат і у вікні *Cost* задати його вартість. Аналогічно призначаються суми по кожному центру витрат, тобто задається вартість кожної роботи з кожної статті витрати. Якщо в процесі призначення вартості виникає необхідність внесення додаткових центрів витрат, діалог *Cost Center Editor* викликається прямо з діалогу *Activity Properties/Cost* відповідною кнопкою.

Загальні витрати по роботі розраховуються як сума по всіх центрах витрат. При обчисленні витрат вищестоячої (батьківської) роботи спочатку обчислюється добуток витрат дочірньої роботи на частоту роботи (число раз, що робота виконується в рамках проведення батьківської роботи), потім результати складаються. Якщо у всіх роботах моделі включений режим *Compute from*

Decompositions, подібні обчислення автоматично проводяться по всій ієрархії робіт знизу нагору.

Cost Center	\$ UAH	Costs
Керування	0.00	
Компоненти	16 000.00	
Робоча сила	100.00	

Data is from this level. Total cost: 16 100,00

Override decompositions Total cost x Frequency: 128 800,00

Compute from decompositions

Frequency: 8,00

Duration: 2,00 Hours

Duration x Frequency 16,00 Hours

Рисунок 6.2 – Діалог Cost Center Editor

Результати відображаються безпосередньо на діаграмах. У лівому нижньому куті прямокутника роботи може показуватися або вартість (за замовчуванням), або тривалість, або частота проведення роботи. Настроювання відображення здійснюється в діалозі *Model Properties* (меню *Model/Model Properties*), закладка *Display (ABC Data, ABC Units)*.

2. Властивості, обумовлені користувачем (UDP)

ABC дозволяє оцінити вартісні й тимчасові характеристики системи. Якщо вартісних показників недостатньо, є можливість внесення власних метрик – властивостей певних користувачів – (*User Defined Properties, UDP*). *UDP* дозволяють провести додатковий аналіз, хоча й без підсумовуючих підрахунків.

Для опису *UDP* служить діалог *User-Defined Property Editor* (меню *Model/UDP Definition Editor*) (рис. 6.3). У верхньому вікні діалогу вноситься ім'я *UDP*, у списку вибору *Datatype* описується тип властивості. Є можливість задання 18 різних типів *UDP*, об'єднаних по категоріях. Для внесення категорії варто задати ім'я категорії у вікні *New Keyword* і клацнути по кнопці *Add Category*. Для присвоєння властивості категорії необхідно вибрати *UDP* зі списку, потім категорію зі списку категорій і клацнути по кнопці *Update*.

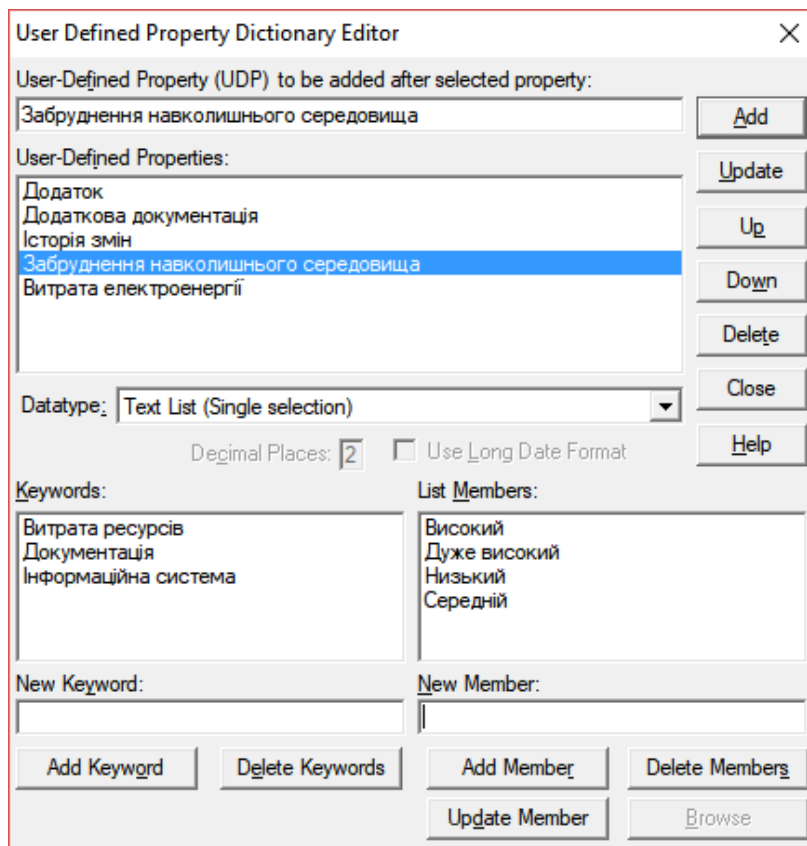


Рисунок 6.3 – Діалог опису UDP

Кожній роботі можна поставити у відповідність набір *UDP*. Для цього варто клацнути правою кнопкою миші по роботі й вибрати пункт меню *UDP*. У закладці *UDP Values* діалогу *IDEF0 Activity Properties* можна задати значення *UDP*.

Завдання на лабораторну роботу

Завдання 1.

У діалозі *Model Properties* (викликається з меню *Mode / Model Properties*) у вкладці *ABC Units* (рис. 6.4) встановіть одиниці виміру грошей і часу – валюта і години.

1. Перейдіть в *Dictionary / Cost Center* і в діалозі *Cost Center Dictionary* внесіть назву та визначення центрів витрат (табл. 6.1).

Для відображення вартості кожної роботи в нижньому лівому кутку прямокутника перейдіть в меню *Model / Model Properties* і у вкладці *Display* діалогу *Model Properties* увімкніть опцію *ABC Data* (рис. 6.5).

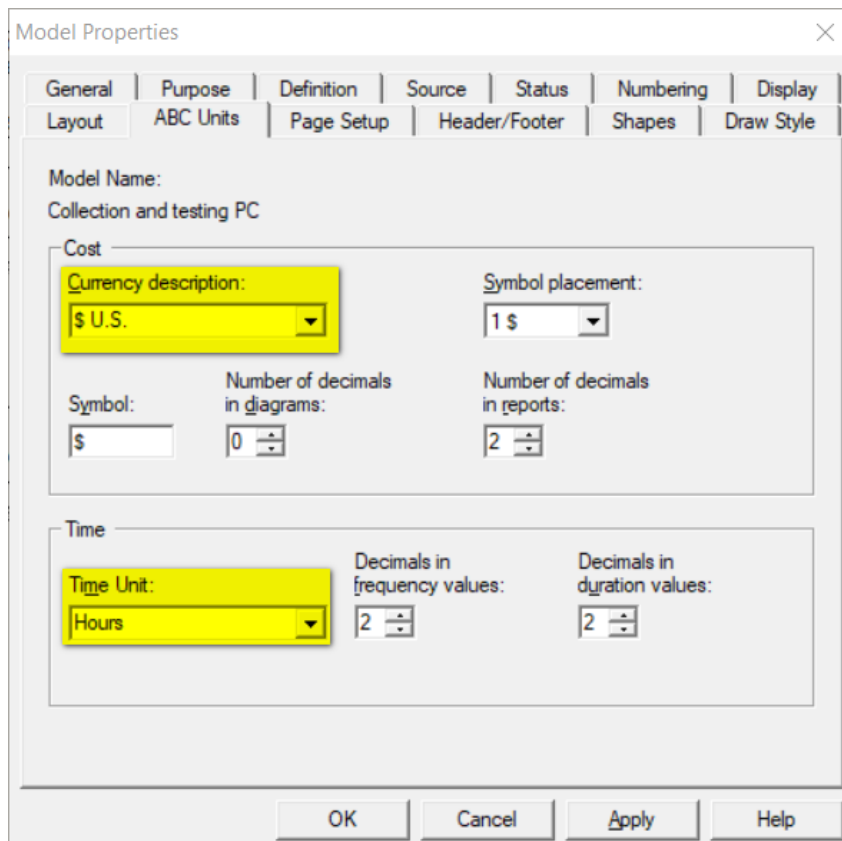


Рисунок 6.4 – Вкладка ABC Units діалогу Model Properties

Таблиця 6.1 – Центри витрат ABC

Центр витрат	Визначення
Керування	Витрати на керування, пов'язані з складанням графіку робіт, формуванням партій комп'ютерів, контролем за збіркою та тестуванням
Робоча сила	Витрати на оплату робітників, зайнятих збиранням та тестуванням комп'ютерів
Компоненти	Витрати на закупку компонентів.

Для відображення частоти або тривалості роботи перемкніть радіокнопки в групі *ABC Units*.

Для призначення вартості роботи слід клацнути по ній правою кнопкою миші і вибрати в контекстному меню *Cost* (рис. 6.6).

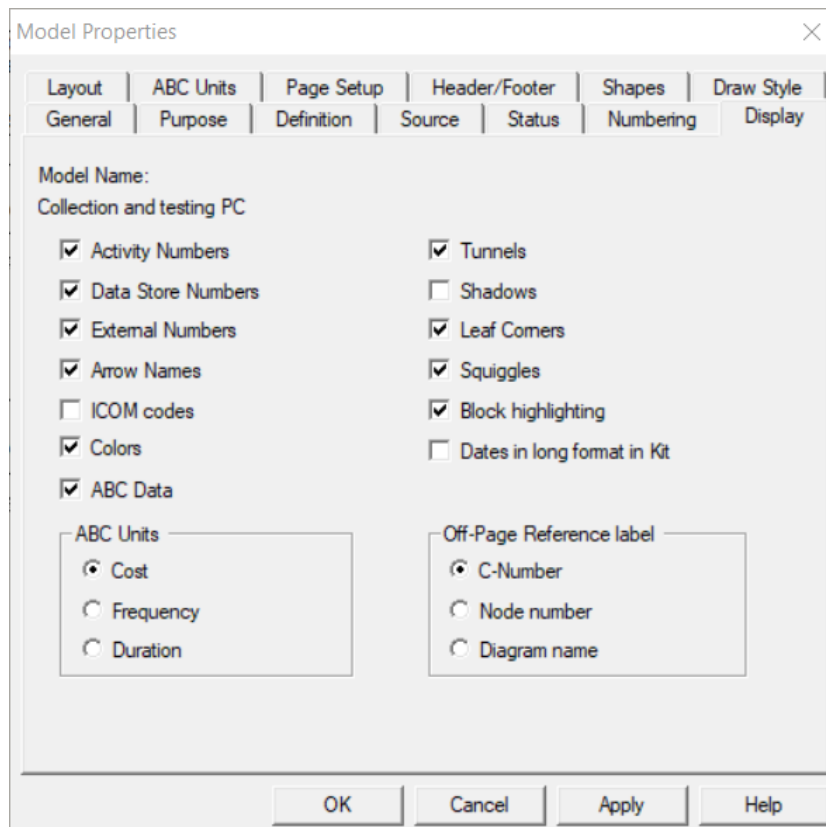


Рисунок 6.5 – Вкладка Display діалогу Model Properties

2. Для робіт на діаграмі A2 внесіть параметри ABC (табл. 6.2).

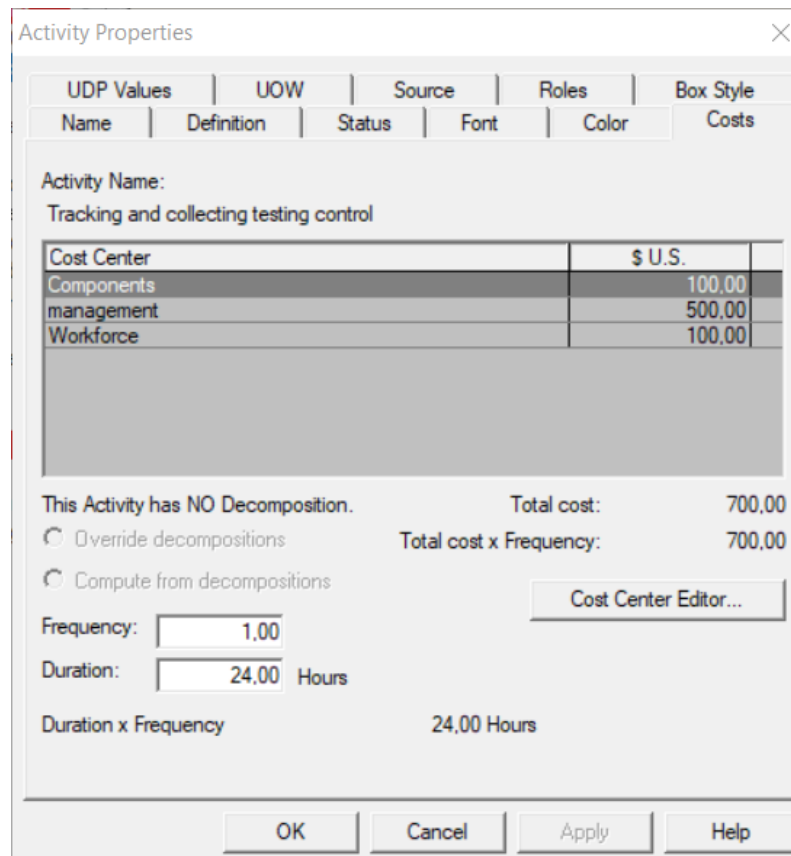


Рисунок 6.6 – Вкладка Cost діалогу Activity Properties 3

Таблиця 6.2 – Вартості робіт на діаграмі A2

Назва роботи (Activity Name)	Центр витрат (Cost Center)	Сумма центру затрат (Cost Center Cost) t, руб	Тривалість, днів (Duration), день	Частота (Frequency)
Відслідковування розкладу та керування збіркою та тестуванням	Керування	500,00	1,00	1,00
Збірка настільних комп'ютерів	Робоча сила	100,00	1,00	12,00
	Компоненти	16000,00		
Збірка ноутбуків	Робоча сила	140,00	1,00	20,00
	Компоненти	28000,00		
Тестування комп'ютерів	Робоча сила	60,00	1,00	32,00

Результат – вартість роботи верхнього рівня показано на рис. 6.7.

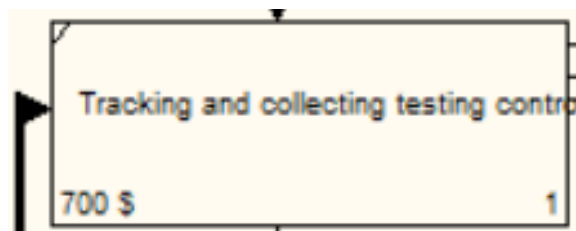


Рисунок 6.7 – Відображення вартості в нижньому лівому кутку прямокутника роботи

3. Згенеруйте звіт *Activity Cost Report* (рис. 6.8).

The image shows a window titled "Activity Cost Report Preview". At the top, it says "Report Format: Column". Below this is a table with three columns: "Name", "Activity Cost (\$ U.S.)", and "Cost Cent". The table contains two rows of data:

Name	Activity Cost (\$ U.S.)	Cost Cent
Collection and testing PC	700,00	Component managemen Workforce
Tracking and collecting testing control	700,00	Component managemen

At the bottom of the window, there are "Close" and "Help" buttons.

Рисунок 6.8 – Звіт Activity Cost Report

Завдання 2. Використання категорій UDP

1. Перейдіть в меню *Dictionary / UDP Keywords* і в діалозі *UDP Keyword List* внесіть ключові слова *UDP* (рис. 6.9):

- Витрата ресурсів;
- Документація;
- Інформаційна система.

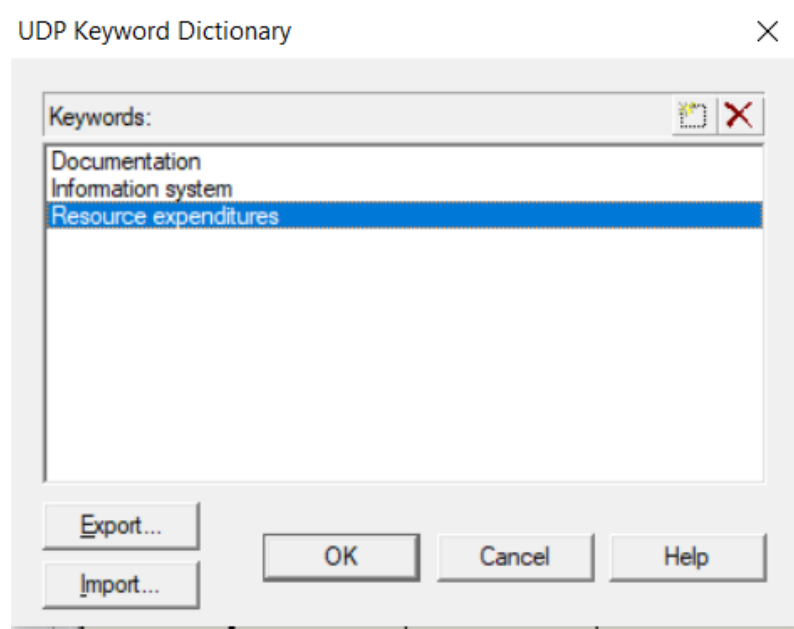


Рисунок 6.9 – Словник ключових слів UDP

2. Створіть UDP. Для цього перейдіть в *Dictionary / UDP* і в словнику внесіть ім'я UDP, наприклад «Додаток».

3. Для UDP типу List необхідно в полі *Value* задати список значень.

4. Для UDP – «Виготовлення». Внесіть значення «*Використання ресурсів*» (рис. 6.10).

Name	Definition	UDP Datatype	Settings	Value	Keyword
Making		Text			Resource expenditure
History of changes		Text			Information system
		Text			

Рисунок 6.10 – Словник UDP

Потім внесіть інші значення відповідно до табл. 6.3. Для підключення до UDP ключового слова перейдіть до поля *Keyword* і клацніть по полю вибору.

Таблиця 6.3 – Найменування і властивості UDP

Найменування UDP	Тип	Значення	Ключове слово
Додаток	Text List(Multiple Selection)	Модуль оформлення замовлень. Модуль створення та контролю розкладу виконання робіт. Модуль обліку комплектуючих та обладнання. Модуль процедур збірки та пошуку недоліків.	Інформаційна система
Додаткова документація	Command List	Winword.EXE sample1.doc Winword.EXE sample2.doc POWERPNT.EXE sample3.ppt	Документація
Історія змін	Paragraph Text		Документація
Забруднення навколишнього середовища	Text List (Single Selection)	Дуже високе Високе Середнє Низьке	
Витрати електроенергії	Real Number		Витрати ресурсів

5. Для призначення UDP роботі слід клацнути по ній правою кнопкою миші та вибрати в контекстному меню UDP. З'являється вкладка *UDP Values* діалогу *Activity Properties* (рис. 6.11).

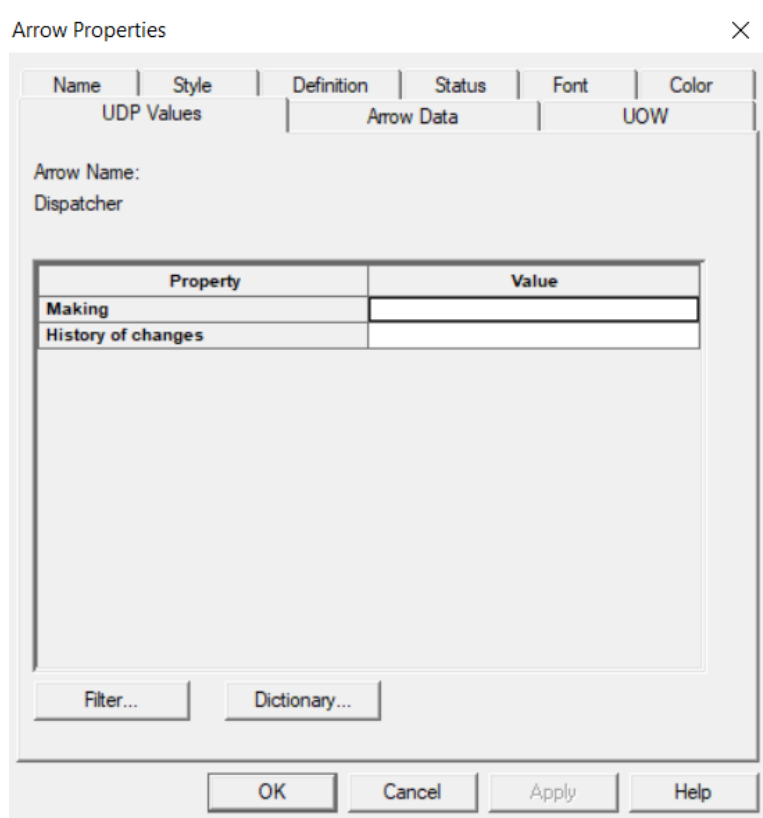



Рисунок 6.11 – Вкладка UDP Values діалогу Activity Properties

Внесіть значення UDP для робіт (таблиця 6.4).

Таблиця 6.4 – Значення UDP

Назва роботи (Activity Name)	Додаткова документація	Додатки	Історія змін	Витрати електроенергії	Забруднення навколишнього середовища
Збірка настільних комп'ютерів		Модуль обліку комплектуючих та обладнання. Модуль процедур збирання і пошуку недоліків		20,00	Середнє
Збірка ноутбуків		Модуль обліку комплектуючих та обладнання. Модуль процедур збирання і		25,00	Середнє

		пошуку недоліків			
Тестування комп'ютерів		Модуль обліку комплектуючих та обладнання. Модуль процедур збирання і пошуку недоліків		40,00	Середнє
Відстежу- вання розкладу та керування збиранням, тестуван- ням	Winword.EXE sample2.doc	Модуль створення та контролю розкладу виконання робіт	Історія зміни специ- фікацій	10,00	Низьке

6. Після внесення UDP типу *Command* або *Command List* клацання по кнопці  призведе до запуску програми.

7. У діалозі *Activity Properties* клацніть по кнопці *Filter*. У діалозі який з'явився – *Diagram object UDP filter* (рис. 6.12) відключіть ключові слова «Інформаційна система». Клацніть по ОК. У результаті в діалозі *Activity Properties* не будуть відображатися UDP з ключовими словами «Інформаційна система».

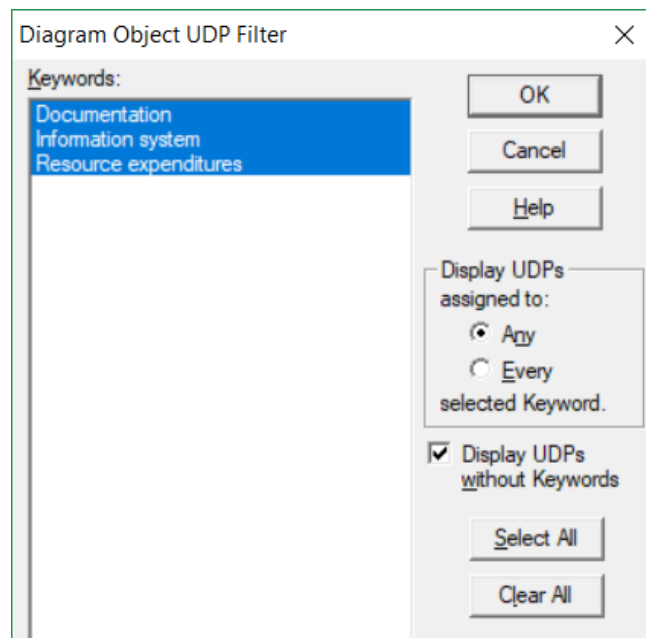


Рисунок 6.12 – Діалог *Diagram object UDP filter*

Зауважимо, що властивості UDP можна привласнити не лише роботам, але й стрілкам.

8. Подивіться звіт по UDP. Меню *Tools / Report / Diagram Object Report*.
 Виберіть опції звіту:
Start from Activity: A2. Збирання і тестування комп'ютерів
Number of Levels: 2
User Defined Properties: Витрата електроенергії Report Format: RPTwin.

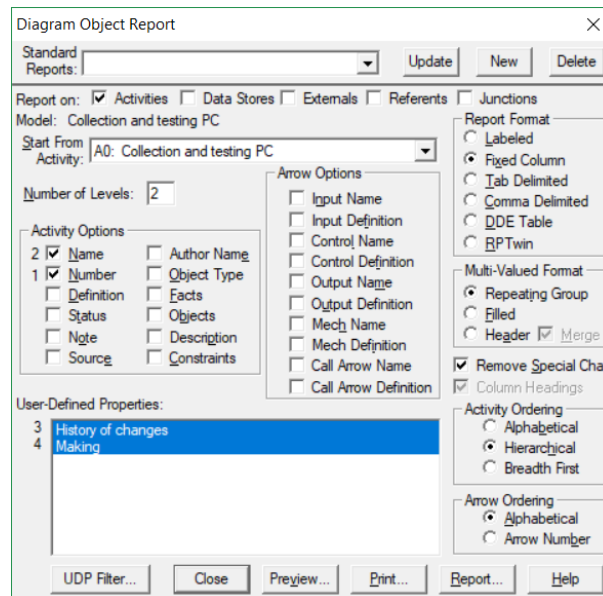


Рисунок 6.12 – Звіт по UDP

9. Клацніть по кнопці *Report*. У діалозі «Збереження файла» клацніть по кнопці «Зберегти».

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №1.
5. Протокол розв'язання завдання №2.
6. Висновки.

Контрольні питання

1. Які умови злиття моделей?
2. Які умови розділення моделей?
3. Поясніть процес злиття моделей.
4. Поясніть процес розділення моделей.

ЛАБОРАТОРНА РОБОТА №7

Створення діаграми потоків даних DFD заданої предметної області.

Мета роботи: Набути практичних навичок застосування методології DFD з використанням AllFusion Process Modeler при створенні функціональної моделі предметної області.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Діаграми потоків даних (Data Flow Diagramming)

Діаграми потоків даних (Data flow diagramming, DFD) використовуються для опису документообігу і обробки інформації. Подібно IDEF0, DFD представляє модельну систему як мережу зв'язаних між собою робіт.

Їх можна використовувати як доповнення до моделі IDEF0 для наочного відображення поточних операцій документообігу в корпоративних системах обробки інформації. У результаті доповнення діаграм IDEF0 діаграмами DFD і IDEF3 може бути створена змішана модель, що найкраще описує всі сторони діяльності підприємства. Декомпонувати роботу DFD на діаграму IDEF0 не можна, так само як декомпонувати роботу IDEF3 на діаграму будь-якої іншої нотації. Основними елементами діаграми потоків даних є: об'єкти, зв'язки, сховища даних.

DFD описує:

1) *роботи* – функції обробки інформації. У DFD роботами є функції системи, що перетворюють входи у виходи. Вони не підтримують управління і механізми, як IDEF0. Стрілки можуть входити і виходити з різних граней;

2) *дуги* – документи (стрілки, arrow), об'єкти, співробітників або відділи, які беруть участь в обробці інформації. Стрілки описують рух об'єктів з однієї частини системи в іншу. Оскільки в DFD кожна сторона роботи не має чіткого призначення, як в IDEF0, стрілки можуть входити виходити з будь-якої грані роботи. У DFD також застосовуються двонаправлені стрілки для опису діалогів типу «команда-відповідь» між роботами, між роботою і зовнішньою суттю і між зовнішньою суттю. Підтримується злиття і розгалуження стрілок.

3) *зовнішні посилання* (external references), які забезпечують інтерфейс із зовнішніми об'єктами, що знаходяться за межами модельованої системи. Зазвичай розташовуються по краях діаграми. Одна зовнішня сутність може бути використана багато разів на одній або декількох діаграмах, цей прийом використовують, щоб не малювати дуже довгих і заплутаних стрілок.

4) *сховища даних* – таблиці для зберігання документів (сховище даних, data store). На відміну від стрілок, що описують об'єкти в русі, сховища даних зображають об'єкти у спокої. У матеріальних системах сховища даних зображуються там, де об'єкти чекають обробки, наприклад в черзі. У системах обробки інформації сховища даних є механізмом, який дозволяє зберегти дані для подальших процесів.

Стрілки DFD показують, як об'єкти (включаючи дані) рухаються від однієї роботи до іншої. Це представлення потоків сумісно зі сховищами даних і зовнішніми посиланнями робить моделі DFD більш схожими на фізичні характеристики системи.

На відміну від IDEF0, де система розглядається як взаємопов'язані роботи, DFD розглядає систему як сукупність предметів. Контекстна діаграма часто включає роботи і зовнішні посилання. Але це не відміння вимоги методології чітко визначити мету, область і єдину точку зору на модельовану систему.

2. Побудова діаграм DFD

Діаграми DFD можуть бути побудовані з використанням традиційного структурного аналізу, подібно до того, як будуються діаграми IDEF0. Спочатку будується фізична модель, що відображає поточний стан справ. Потім ця модель перетвориться в логічну модель, яка відображає вимоги до існуючої системи. Після цього будується модель, що відображає вимоги до майбутньої системи та фізична модель, на основі якої повинна бути побудована нова система.

Альтернативним підходом є підхід, популярний при створенні програмного забезпечення, званий подієвим розділенням (*event partitioning*), в якому різні діаграми DFD вибудовують модель системи.

По-перше, логічна модель будується як сукупність робіт і документування того, що вони (ці роботи) повинні робити.

Потім модель оточення (*environment model*) описує систему як об'єкт, що взаємодіє з подіями із зовнішньої суті. Модель оточення зазвичай містить опис мети системи, одну контекстну діаграму і список подій. Контекстна діаграма містить один прямокутник роботи, що зображає систему в цілому, і зовнішню суть, з якою система взаємодіє.

Нарешті, модель поведінки (*behavior model*) показує, як система обробляє події. Ця модель складається з однієї діаграми, в якій кожен прямокутник зображає кожна подія з моделі оточення. Сховища можуть бути додані для моделювання даних, які необхідно запам'ятовувати між подіями. Потоки додаються для зв'язку з іншими елементами, і діаграма перевіряється з погляду відповідності моделі оточення.

Отримані діаграми можуть бути перетворені з метою наочного представлення системи, зокрема роботи на діаграмах можуть бути декомпоновані.

3. Нумерація об'єктів

У DFD номер кожної роботи може включати префікс, номер батьківської роботи (A) і номер об'єкту.

Номер об'єкту – це унікальний номер роботи на діаграмі. Наприклад, робота може мати номер A.12.4. Унікальний номер мають сховища даних і зовнішню суть незалежно від їх розташування на діаграмі. Кожне сховище даних має префікс D і унікальний номер, наприклад D5. Кожна зовнішня суть має префікс E і унікальний номер, наприклад E5.

4. Декомпозиція роботи IDEF0 у діаграму DFD

Для створення дочірньої діаграми DFD треба при декомпозиції в діалозі *Activity Box Count* вибрати варіант DFD. Створюється нова діаграма DFD і стрілки, які стосуються батьківської роботи, мігрують на діаграму нижнього рівня так, ніби це була діаграма IDEF0.

Стрілки входу батьківської роботи на дочірній діаграмі DFD показуються вхідними стрілками з лівої сторони діаграми DFD, стрілки керування – вхідними стрілками з верхньої сторони діаграми й т.д. Хоча нотація DFD не включає поняття «керування» і «механізм» і можна створювати внутрішні стрілки вихідними з будь-якої грані роботи й вхідними в будь-яку грань, AllFusion Process Modeler не дозволяє зв'язати граничні стрілки на діаграмі DFD довільним чином. Стрілки можна зв'язати тільки так, ніби це була діаграма IDEF0.

Відповідно до нотації DFD діаграма не повинна мати граничних стрілок – всі стрілки повинні починатися і закінчуватися на роботах, сховищах даних або зовнішніх сутностях. Тому якщо строго дотримуватися правил нотації, треба:

Видалити всі граничні стрілки на діаграмі DFD.

- Створити відповідні зовнішні сутності й сховища даних.
- Створити внутрішні стрілки, що починаються із зовнішніх сутностей замість граничних стрілок.
- Стрілки на діаграмі IDEF0 затунелювати.

Строго дотримуватися правил нотації DFD при створенні змішаних моделей не завжди зручно, тому BPwin дозволяє створювати граничні стрілки на діаграмах DFD і не ідентифікує такі стрілки як синтаксичну помилку.

5. Міжсторінкові посилання (Off-Page Reference) й зовнішні сутності (External Reference) на діаграмах DFD і IDEF0

Нотація DFD включає міжсторінкові посилання – інструмент, що дозволяє описати перехід стрілки (тобто передачу даних або об'єктів) з однієї діаграми на іншу.

Деякі стрілки з діаграм IDEF0 і DFD (не тільки з батьківських) можуть показуватися на діаграмі DFD. Для відображення таких стрілок використовується інструмент *Off-Page Reference*.

Для створення міжсторінкового посилання на діаграмі DFD слід створити нову граничну стрілку. У границі діаграми ця стрілка буде позначена квадратними дужками, так само як заборонена стрілка на діаграмі IDEF0. Потім

варто клацнути правою кнопкою миші по квадратних дужках і вибрати в контекстному меню пункт *Off-Page Reference*.

З'являється діалог *Off-Page Arrow Reference*. У ньому необхідно вказати діаграму, на яку буде спрямована стрілка, і, якщо це діаграма в нотації IDEF0, границю, від якої буде виходити стрілка (*Destination border*).

У результаті буде створене міжсторінкове посилання як на діаграмі-джерелі, так і на діаграмі-призначенні. Міжсторінкове посилання може бути позначена як *C-number* діаграми, як номер діаграми у вузлі або як ім'я діаграми.

Для зміни мітки треба перейти в меню *Edit/Model Properties* і в закладці *Display* діалогу *Model Properties* та в групі *Off-Page Reference label* вибрати потрібну опцію.

AllFusion Process Modeler дозволяє створити на границі діаграми не тільки міжсторінкове посилання, але й зовнішню сутність і тунель.

Для створення зовнішньої сутності на діаграмі DFD слід створити нову граничну стрілку. У границі діаграми ця стрілка буде позначена квадратними дужками. Потім клацнути правою кнопкою миші по квадратних дужках і вибрати в контекстному меню пункт *External Reference*. У діалозі *External Reference* слід вибрати або внести ім'я зовнішньої сутності.

На діаграмі DFD можна також створити тунельну стрілку, хоча нотація DFD не передбачає створення такого елемента. Для цього треба клацнути правою кнопкою миші по квадратних дужках і вибрати в контекстному меню пункт *Arrow Tunnel*.

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler на основі розробленої студентом функціональні моделі за нотацією IDEF0, побудувати діаграми потоків даних відповідно до нотації DFD та згідно номеру варіанту (табл. 1.1).

Завдання 1. Створити діаграму DFD.

Завдання 2. Побудувати міжсторінкове посилання на діаграмі DFD за допомогою *Off-Page Reference*.

Приклад 1. Створення діаграми DFD на прикладі діяльності комп'ютерної компанії.

«Оформлення замовлень» починається з дзвінка клієнта. У процесі оформлення замовлення база даних клієнтів може проглядатися й редагуватись. Замовлення повинно включати як інформацію про клієнта, так й інформацію по замовлених продуктах. Оформлення замовлення та запис інформації по інших замовленнях. У процесі декомпозиції згідно з правилами DFD необхідно перетворити граничні стрілки у внутрішні, що починаються і закінчуються на зовнішніх посиланнях.

1. Декомпозуємо роботу «Оформлення замовлень» на діаграмі A2.

2. Для цього в діалозі *Activity Box Count* виберемо кількість робіт 2 і нотацію DFD, як на рис. 7.1.

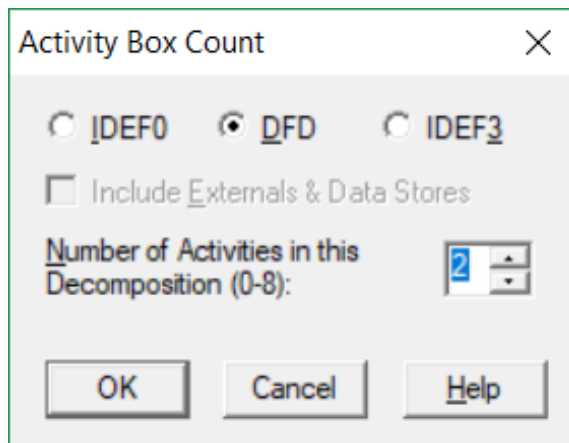


Рисунок 7.1 – Вибір нотації DFD в діалозі *Activity Box Count*


3. Після вибору нотації DFD, підтверджуючи натиском кнопки «Ок», в нову діаграму DFD A22 вносимо імена наступних робіт:

- Перевірка та внесення клієнта;
- Внесення замовлення.

4. Використовуючи кнопку  палітри інструментів, вносимо сховища даних:

- Список клієнтів;
- Список продуктів;
- Список замовлень.

5. Видалимо граничні стрілки з діаграми DFD A22.

6. Використовуючи кнопку  панелі інструментів, вносимо зовнішнє посилання:

- Дзвінки клієнтів.

7. Створимо внутрішні посилання (рис. 7.2). При іменуванні стрілок використовуємо словник.

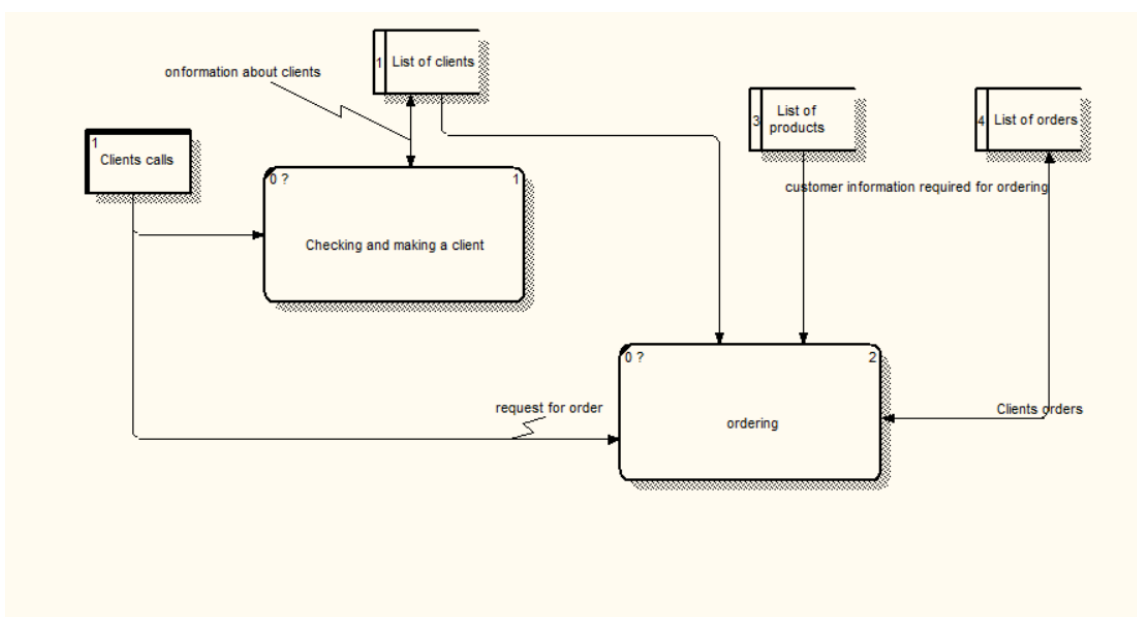


Рисунок 7.2 – Діаграма A22

8. Стрілки «Інформація про клієнтів» та «Замовлення клієнтів» – двонаправлені. Для того, щоб зробити стрілку двобічною, клацнемо правою кнопкою по стрілці та виберемо в контекстному меню пункт *Style*, а у вкладці *Style* – опцію *Bidirectional*.

9. На батьківській діаграмі A2 тунелюємо (*Change to Tunnel*) стрілки, що підходять і виходять з роботи «Оформлення замовлень» (рис. 7.3).

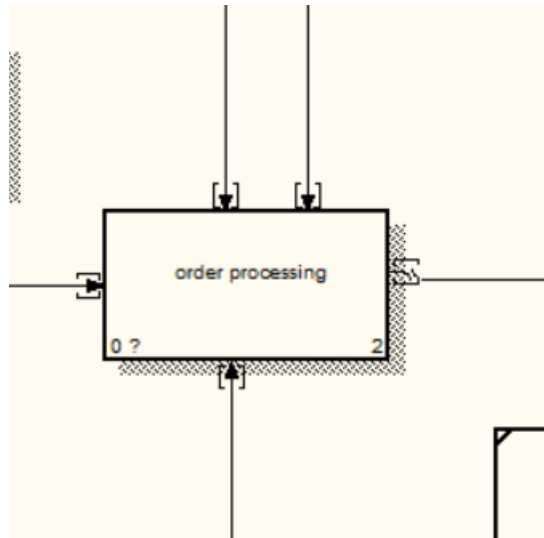



Рисунок 7.3 – Робота «Оформлення замовлень» на діаграмі A2

Приклад 2. Побудова міжсторінкового посилання на діаграмі DFD за допомогою Off-Page Reference.

1. Декомпозуємо роботу «Дослідження ринку» діаграми A2 на діаграму DFD. Видалимо граничні стрілки та створимо такі роботи:

- Розробка прогнозів продажів;
- Розробка маркетингових матеріалів;
- Залучення нових клієнтів.

2. Використовуючи кнопку  палітри інструментів, вносимо сховища даних:

- Список клієнтів;
- Список продуктів;
- Список замовлень.

3. Додамо два зовнішні посилання:

- Маркетингові матеріали;
- Прогноз продажів.

4. Зв'яжемо об'єкти діаграми DFD стрілками, як показано на рис. 7.4.

5. На батьківській діаграмі A2 тунелюємо (*Change to Tunnel*) стрілки, вхідні та вихідні з роботи «Дослідження ринку». У разі внесення нових клієнтів у роботі «Перевірка та внесення клієнта» на діаграмі A22 «Оформлення замовлень» інформація повинна направлятися до роботи «Залучення нових клієнтів» діаграми A23 «Дослідження ринку». Для цього необхідно використовувати інструмент Off-Page Reference.

6. На діаграмі A22 «Оформлення замовлень» створимо нову граничну стрілку, яка виходить від роботи «Перевірка та внесення клієнта», і назвемо її «Інформація про нового клієнта».

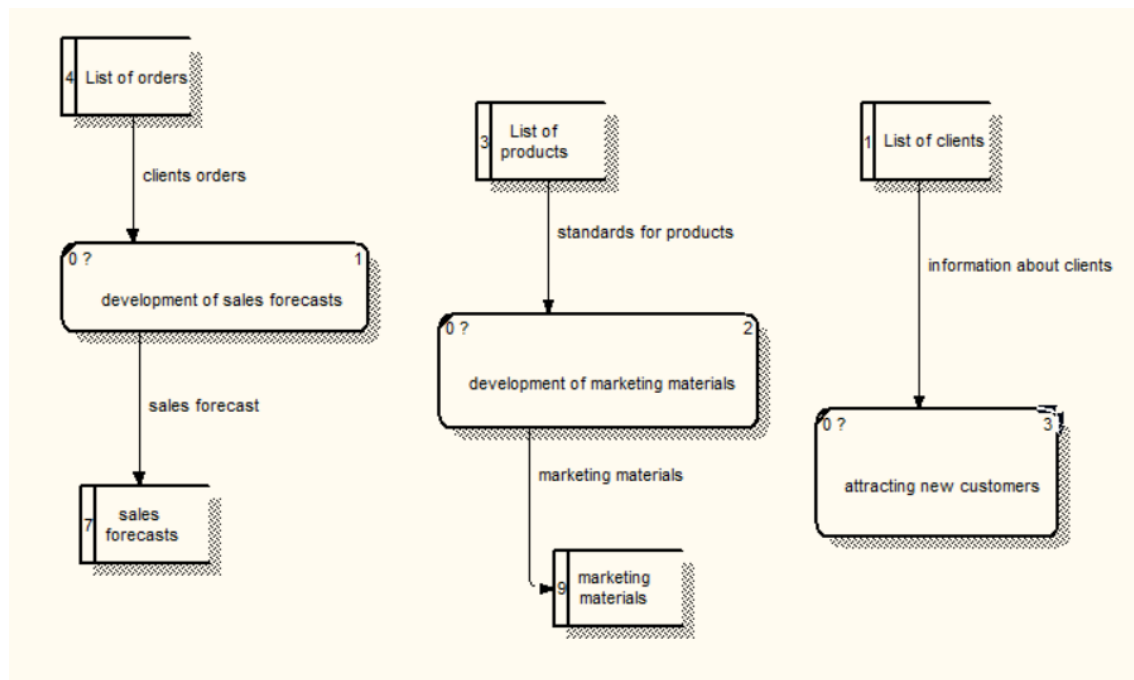


Рисунок 7.4 – Діаграма A23

7. Правою кнопкою клацнемо по наконечнику стрілки і виберемо в меню *Off-Page Reference*. У діалозі *Off-Page Arrow Reference* (рис. 7.5) виберемо як діаграми A23D «Дослідження ринку».

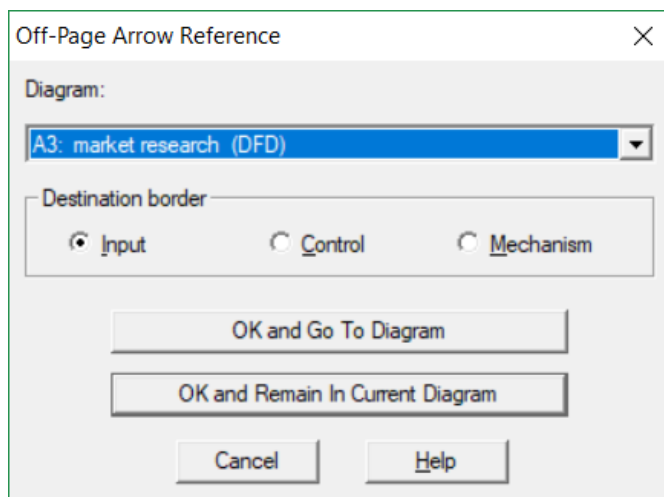


Рисунок 7.5 – Діалог *Off-Page Arrow Reference*

8. Перейдемо в меню *Model / Model Properties*, далі – у вкладку *Display*.

9. Встановимо опцію *Off-Page Reference label – Node number*.

10. Перейдемо на діаграму A23D «Дослідження ринку» і направимо стрілку «Інформація про нового клієнта» на вхід роботи «Залучення нових клієнтів».

Результат представлений на рис. 7.6.

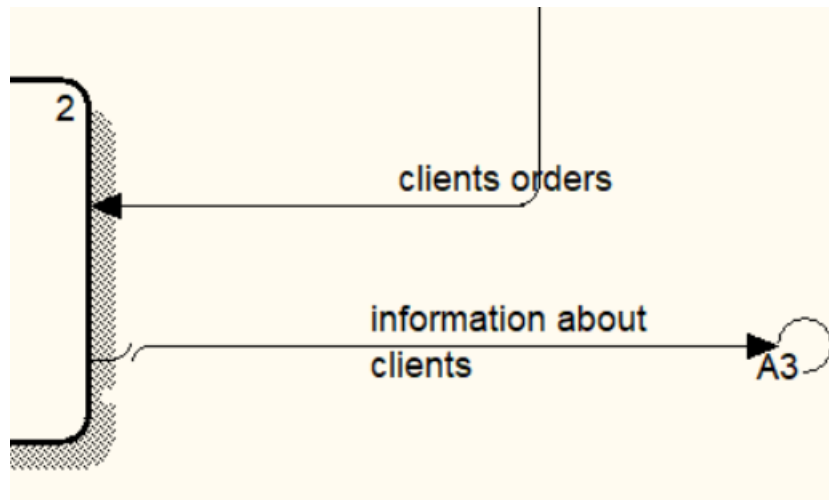


Рисунок 7.6 – Міжсторінкове посилання на діаграмі A23

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №1.
5. Протокол розв'язання завдання №2.
6. Висновки.

Контрольні питання

1. Визначте застосування діаграми потоків даних.
2. Розкрийте принцип побудови діаграм DFD.
3. Назвіть особливості діаграм DFD.
4. Як здійснюється нумерація об'єктів у діаграмі DFD?
5. Вкажіть основні елементи діаграми потоків даних.
6. Поясніть різницю між моделями IDEF0, DFD, IDEF3.
7. Опишіть сутність моделі DFD.
8. Поясніть створення міжсторінкового посилання.

ЛАБОРАТОРНА РОБОТА №8

Створення моделі TO-BE.

Мета роботи: Набути практичних навичок створення моделі TO-BE з використанням AllFusion Process Modeler на основі аналізу моделі AS-IS.

Порядок виконання роботи

1. Вивчити теоретичні відомості.
2. Разом з викладачем вибрати варіант завдання.
3. Виконати завдання до лабораторної роботи згідно свого варіанту.
4. Скласти та оформити звіт.

Теоретичні відомості

1. Аналіз функціональної моделі AS-IS

AS-IS (як є) – модель дозволяє виявити, «що ми робимо сьогодні» перед тим, як перескочити на те, «що ми будемо робити завтра».

Функціональна модель існуючої організації роботи – AS-IS («як є») створена за допомогою CASE-засобів, наприклад, програмного продукту AllFusion Process Modeler.

При її побудові використовуються такі джерела, як пропорції експертів (працівників підприємств), документація (покази, звіти, державні інструкції, положення про структурні підрозділи підприємств), анкетування, фотографія робочого дня та ін.

Аналіз функціональної моделі дозволяє:

- спеціалістам, що виконують роботу по реінжинірингу, виявити недоліки існуючого бізнесу та визначити ті області, в яких варто виробляти зміни;
- зрозуміти, як слід змінити бізнес-процеси (БПр), щоб вони задовольнили нові вимоги та цілі;
- оцінювати, на скільки глибоким змінам підвергається існуюча структура організації бізнесу;
- повідомити співробітникам, чому не задовольняють попередні процедури роботи та чому вони повинні бути приєднаними до нових способів роботи;
- вимірювати характеристики тих БПр, які слід змінити.

Одною з помилок при створенні моделей AS-IS є створення ідеалізованої моделі. Це відбувається в тому випадку, коли модель створюється на основі знаного керівника, а не на основі досвіду конкретного виконавця робіт.

2. Аналіз моделі TO-BE

TO-BE (як буде) – у цій моделі виправлені недоліки найдавніших моделей AS-IS. TO-BE – це моделі нової організації бізнес-процесу. Ця модель необхідна для оцінки результатів впровадження інформаційних систем та аналізу

альтернативних / найкращих шляхів виконання робіт та документації до того, як підприємство буде функціонувати в майбутньому.

Модель TO-BE створюється на основі аналізу моделі AS-IS. Аналіз може проводитися як за формальними ознаками (відсутність виходів або управлінь у робіт, відсутність зворотних зв'язків і т. д.), так і за неформальним – на основі знань предметної області.

Завдання на лабораторну роботу

Робота виконується за допомогою CASE-засобу AllFusion Process Modeler та згідно номеру варіанту (табл. 1.1).

Завдання 1. Виконати завдання 1.1-1.5 прикладу 1.

Завдання 2. Створити модель TO-BE на основі аналізу моделі AS-IS згідно номеру варіанту по аналогії прикладу 1.

Приклад 1. Створення моделі TO-BE на прикладі діяльності комп'ютерної компанії.

Припустимо, що в результаті аналізу приймається рішення реорганізувати функції виробництва і тестування комп'ютерів і залишити функціональності «Продажі та маркетинг» та «Відвантаження і отримання» поки без змін.

Прийнято рішення сформуванню відділу дизайну, який повинен:

- формувати конфігурацію комп'ютерів;
- розробляти корпоративні стандарти;
- підбирати прийнятних постачальників;
- розробляти інструкції по збірці;
- процедури тестування та усунення неполадок для всього виробничого відділу.

Робота «Збирання і тестування комп'ютерів» має бути реорганізована і названа «Виробництво продукту». Будуть створені роботи: «Розробити конфігурацію», «Планувати виробництво» та «Зібрати продукт».

Розглянемо нові ролі персоналу.

Функції дизайнера: дизайнер повинен розробляти систему, стандарти на продукцію, документувати і передавати специфікації у відділ маркетингу та продажів. Він повинен визначати, які компоненти (апаратні і програмні) повинні закуповуватися для складання комп'ютерів, забезпечувати документацією і керувати процедурами складання, тестування та усунення неполадок.

Функції диспетчера в роботі «Збирання і тестування комп'ютерів» повинні бути замінені на функції планувальника.

Планувальник повинен обробляти замовлення клієнтів і генерувати замовлення на збірку, отримати комерційний прогноз з відділу маркетингу і формувати вимоги на закупівлю компонентів та збирати інформацію від постачальників.

Диспетчер повинен складати розклад виробництва на підставі замовлень на складання, отриманих у результаті роботи «Планувати виробництво»,

отримувати копії замовлень клієнтів і відповідати за упаковку і комплектацію замовлених комп'ютерів, що передаються в роботу «Відвантаження і отримання».

1.1 Розщеплення і модифікація моделі.

1. Змініть властивості моделі «Діяльність компанії»:

- Model Name: Пропонована модель компанії;
- Time Frame: TO-BE;
- Purpose: Документувати пропоновані зміни бізнес-процесів компанії.

2. Переіменуйте роботу «Збирання і тестування комп'ютерів» в «Виробництво продукту». Розщепіть цю роботу в модель з тією ж назвою.

3. Модифікуйте відщеплену модель. Перемістіть роботу «Тестування комп'ютерів» з діаграми A0 «Виробництво продукції» на діаграму A2.1 «Збірка настільних комп'ютерів».

4. Переіменуйте роботу «Збірка настільних комп'ютерів» на діаграмі A0 в «Збірку продукту».

5. Видаліть роботу «Збірка ноутбуків».

6. Переіменуйте стрілку «Замовлення на настільні комп'ютери» в «Замовлення на виготовлення».

7. Переіменуйте «Відстеження розкладу і керування збиранням і тестування» в «Планування виробництва».

8. Створіть роботу «Розробити конфігурацію».

9. Створіть гілку стрілки «Персонал виробничого відділу», назвіть її «Дизайнер» та направте її як механізм роботи «Розробити конфігурацію».

10. Створіть стрілку «Стандарти на продукцію» і направте її від виходу «Розробити конфігурацію» до межі діаграми. Тунелюйте цю стрілку (*Resolve Border Arrow*). Створіть гілку цієї стрілки, що йде до управління роботи «Планування виробництва» і назви її «Списком необхідних компонентів».

11. Видаліть стрілку «Правила складання і тестування». Створіть гілку стрілки «Стандарти на продукцію», що йде до управління роботи «Збірка продукту» і назвіть її «Правилами складання і тестування».

12. Переіменуйте стрілку «Диспетчер» у «Планувальника виробництва».

13. Додайте стрілку «Прогноз продажів» як граничну керуючу до роботи «Планування виробництва».

14. Додайте стрілку «Інформація від постачальника» як граничну керуючу до роботи «Планування виробництва».

15. Додайте стрілку «Замовлення постачальникові» як граничну стрілку виходу від роботи «Планування виробництва».

16. Тунелюйте ці стрілки (*Resolve Border Arrow*).

17. На діаграмі A-0 тунелюйте стрілку (*Resolve Border Arrow*) «Зібрані комп'ютери» і зв'яжіть її на діаграмі A0 з виходом роботи «Збірка продукту».

Результат виконання завдання 1.1 наведено на рис. 8.1 та рис. 8.2.

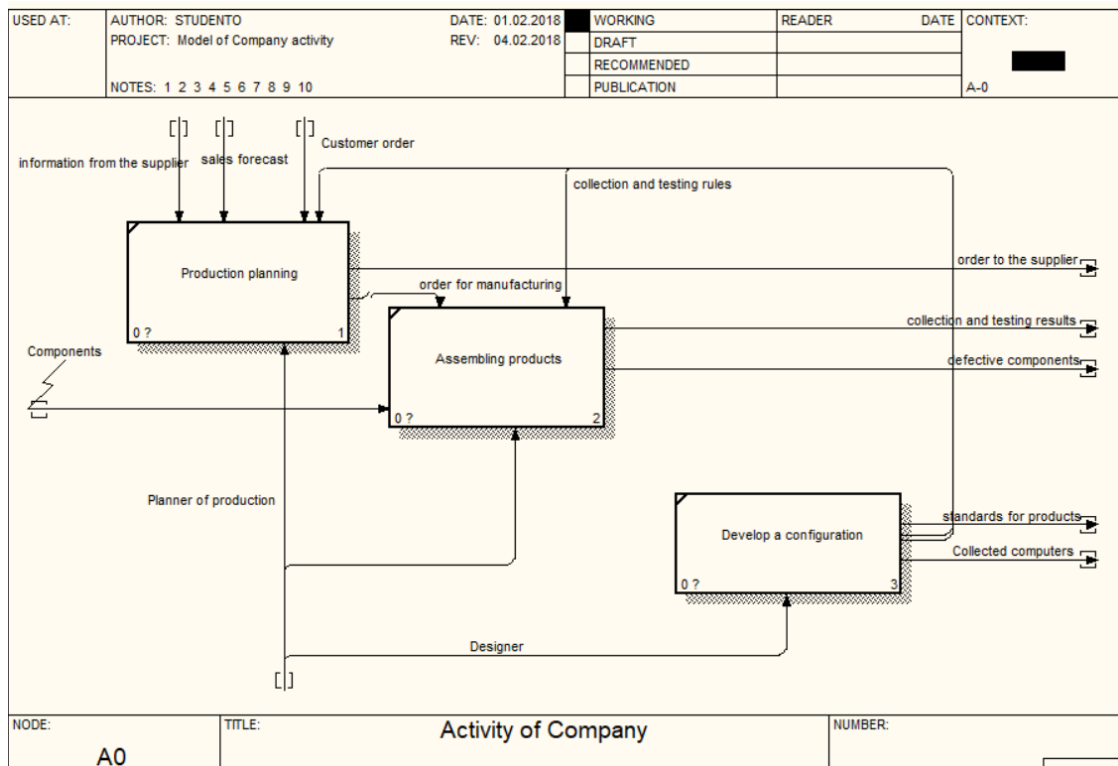


Рисунок 8.1 – Результат виконання завдання 1.1 (діаграма A0)

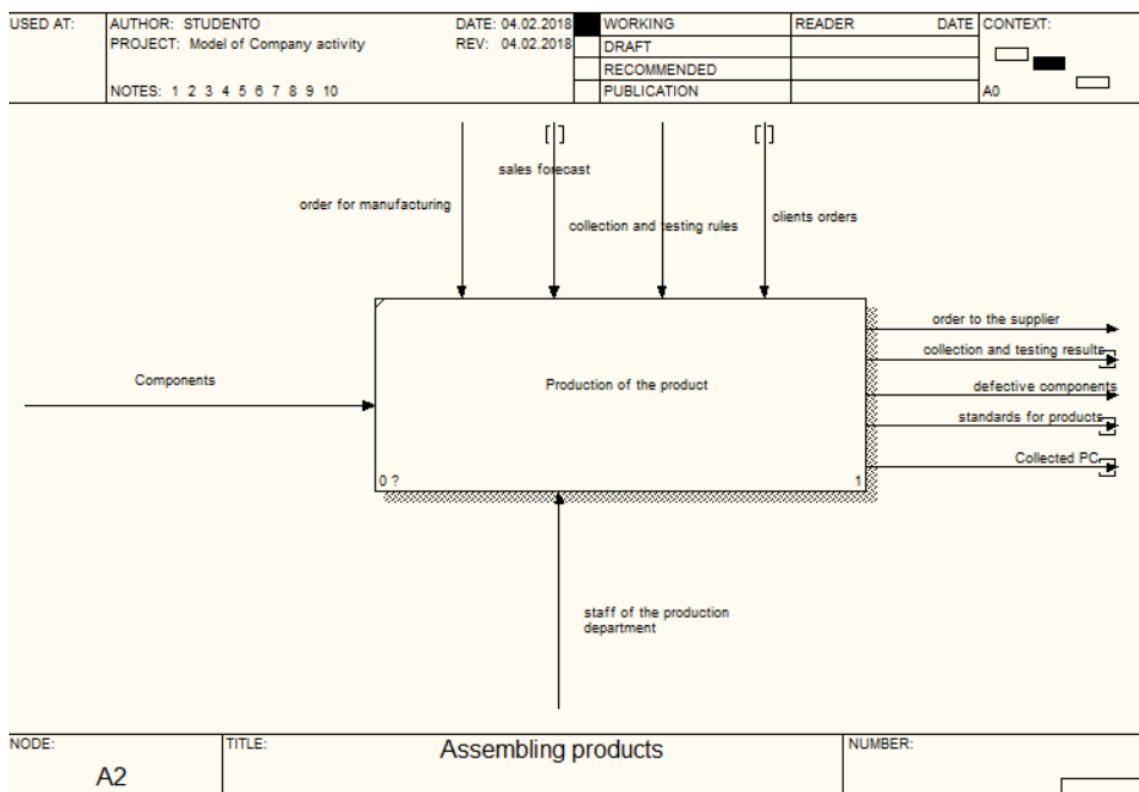


Рисунок 8.2 – Результат виконання завдання 1.1 (діаграма A-0)

1.2. Злиття моделі

1. Перейдіть до роботи «Виробництво продукту» в моделі «Діяльність компанії». Клацніть правою кнопкою миші по роботі. У контекстному меню виберіть *Merge Model*. У діалозі *Merge Model* встановіть опцію *Cut / Paste entire dictionaries*, опцію *Overwrite existing fields* і клацніть по *OK*. Моделі повинні злитися.

2. На діаграмі A0 тунелюйте стрілки (Resolve Border Arrow,) «Інформація від постачальника» та «Замовлення постачальникові».

3. Направте стрілку «Прогноз продажів» з виходу «Продажі та маркетинг» на управління «Виробництво продукту».

4. Направте стрілку «Стандарти на продукцію» з виходу «Виробництво продукту» на управління «Продажі та маркетинг».

5. Видаліть гілку стрілки управління «Правила і процедури» роботи «Виробництво продукту».

6. Закрийте модель «Виробництво продукту».

Результат виконання завдання 1.2 наведено на рис. 8.3 і рис. 8.4.

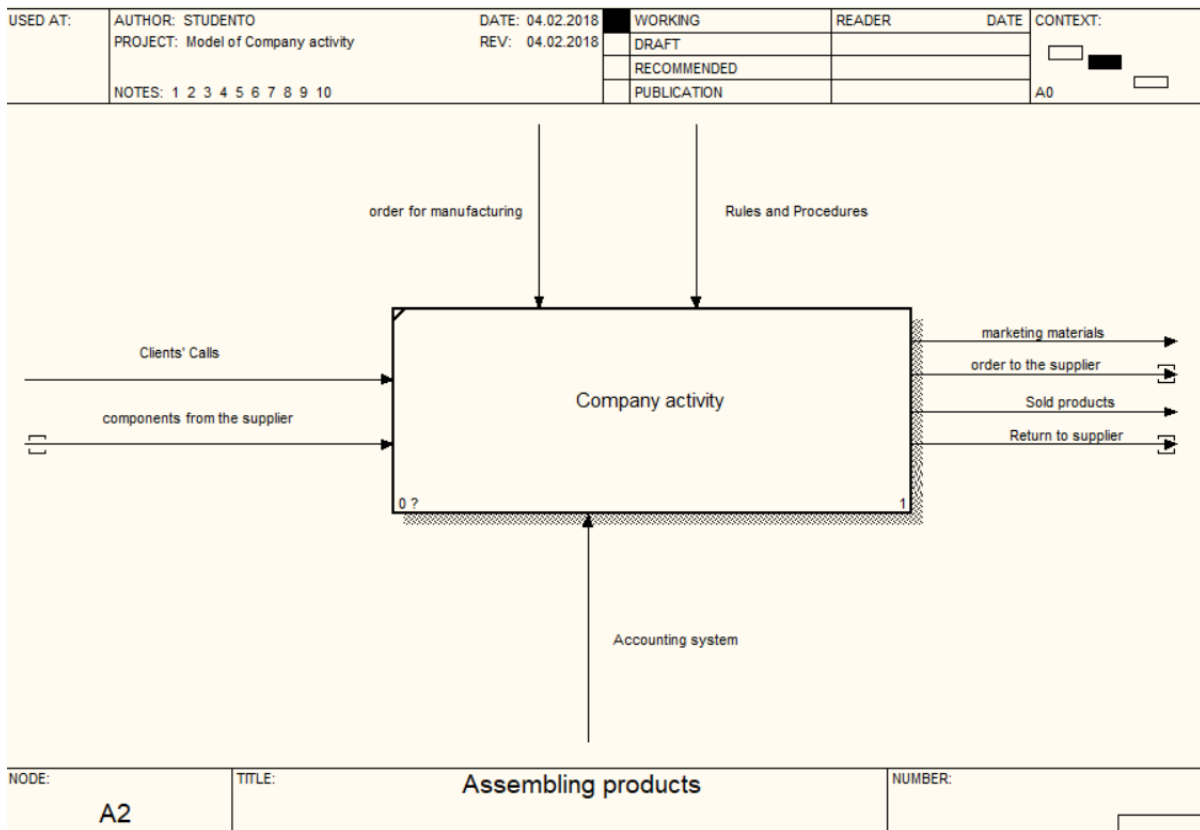


Рисунок 8.3 – Результат виконання завдання 1.2 (діаграма A0)

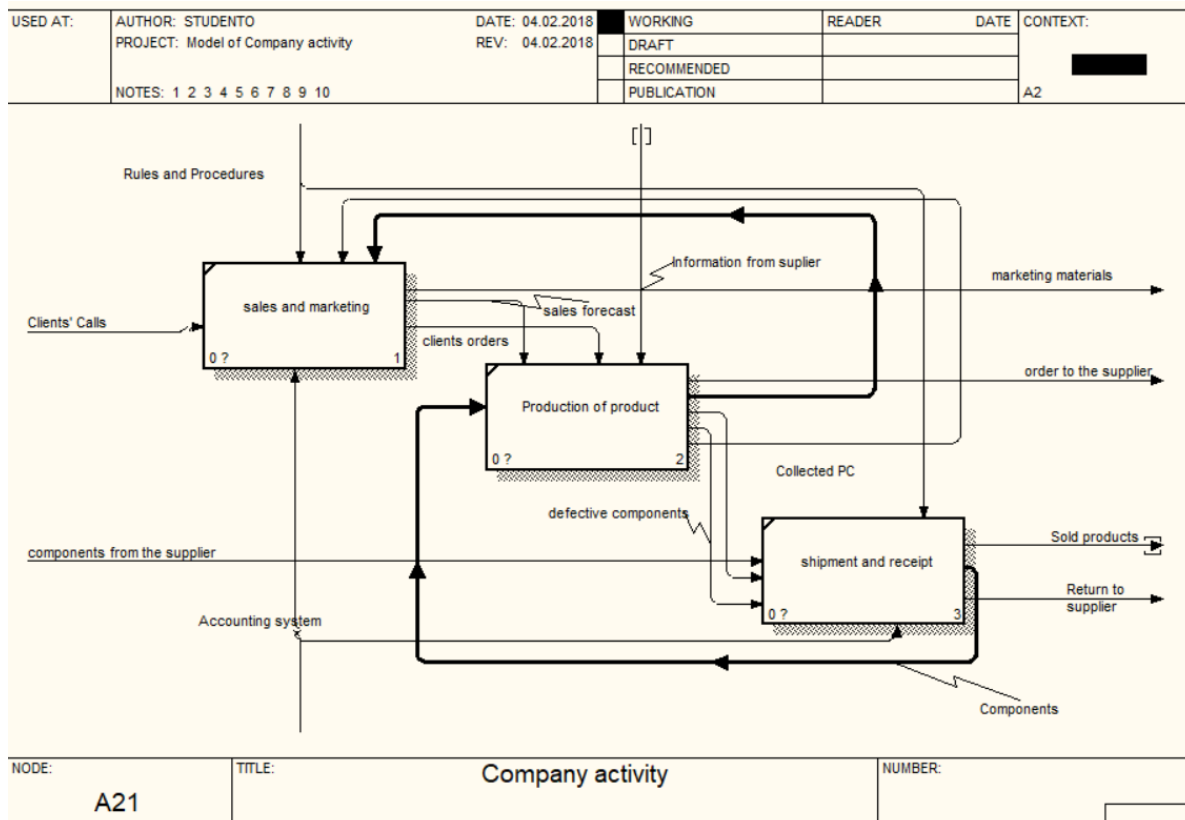


Рисунок 8.4 – Результат виконання завдання 1.2 (діаграма А-0)

1.3 Використання *Model Explorer* для реорганізації дерева декомпозиції

Існують причини, за якими робота «Розробити конфігурацію» повинна бути на верхньому рівні на діаграмі А0. Дійсно, дизайнер розробляє стандарти на продукцію, включаючи правила збирання та тестування, та список необхідних для закупівлі компонентів. Тим самим дизайнер управляє виробництвом продукту в цілому, крім того, управляє роботою «Продажі та маркетинг». Доцільним було б логічно перенести цю роботу на рівень вище.

Використовуючи можливості *Model Explorer*, перенесіть роботу «Розробити конфігурацію» з діаграми А2 «Виробництво продукту» на діаграму А0. Дозвольте і перенаправте стрілки як на рис. 8.5 і рис. 8.6.

1.4 Модифікація діаграми IDEF3 «Збірка продукту» з метою відображення нової інформації

Так само, як в моделі AS-IS, збірка продукту складається із збирання компонентів та встановлення програмного забезпечення. Однак, тепер в роботу «Збірка продукту» включена робота «Тестування комп'ютера».

Тестування починається після закінчення процесу збирання комп'ютера і закінчення процесу установки програмного забезпечення. Якщо комп'ютер має неполадки, то в процесі тестування у нього замінюють компоненти, інформація про несправні компоненти може бути направлена на роботу «Підготовка компонентів». Така інформація може допомогти більш ретельно готувати компоненти до складання. Результатом процесу тестування є замовлені комп'ютери і несправні компоненти.

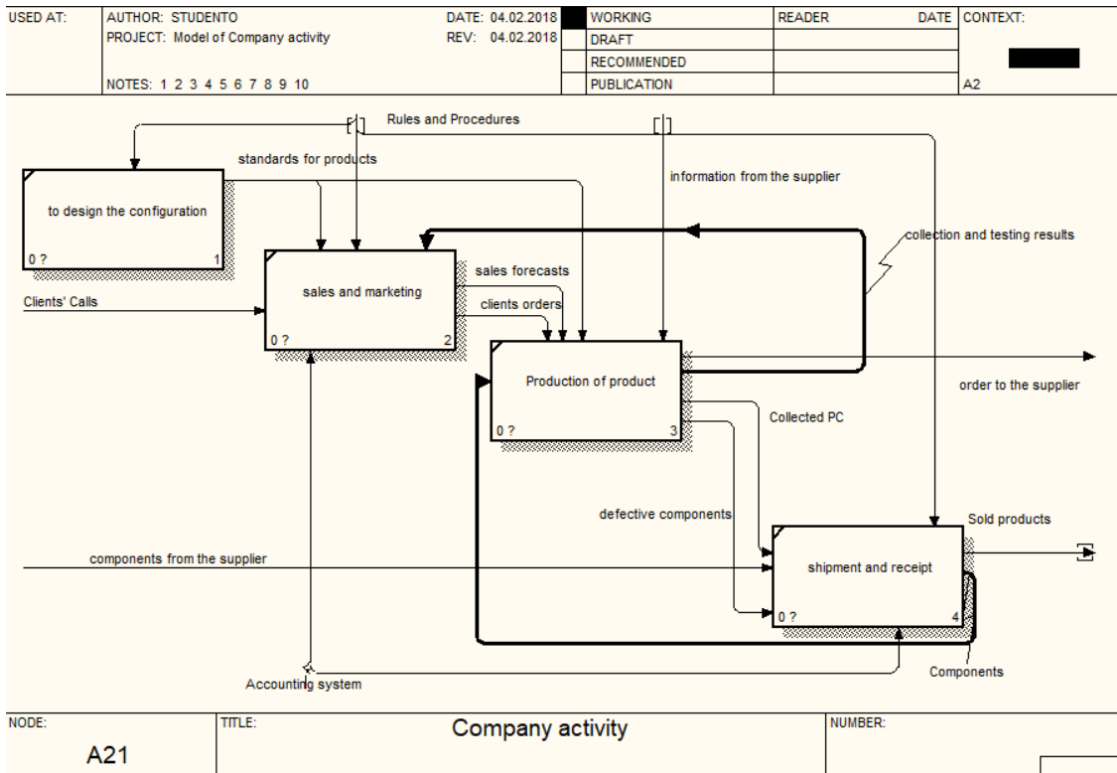


Рисунок 8.5 – Результат виконання завдання 1.3 (діаграма A0)

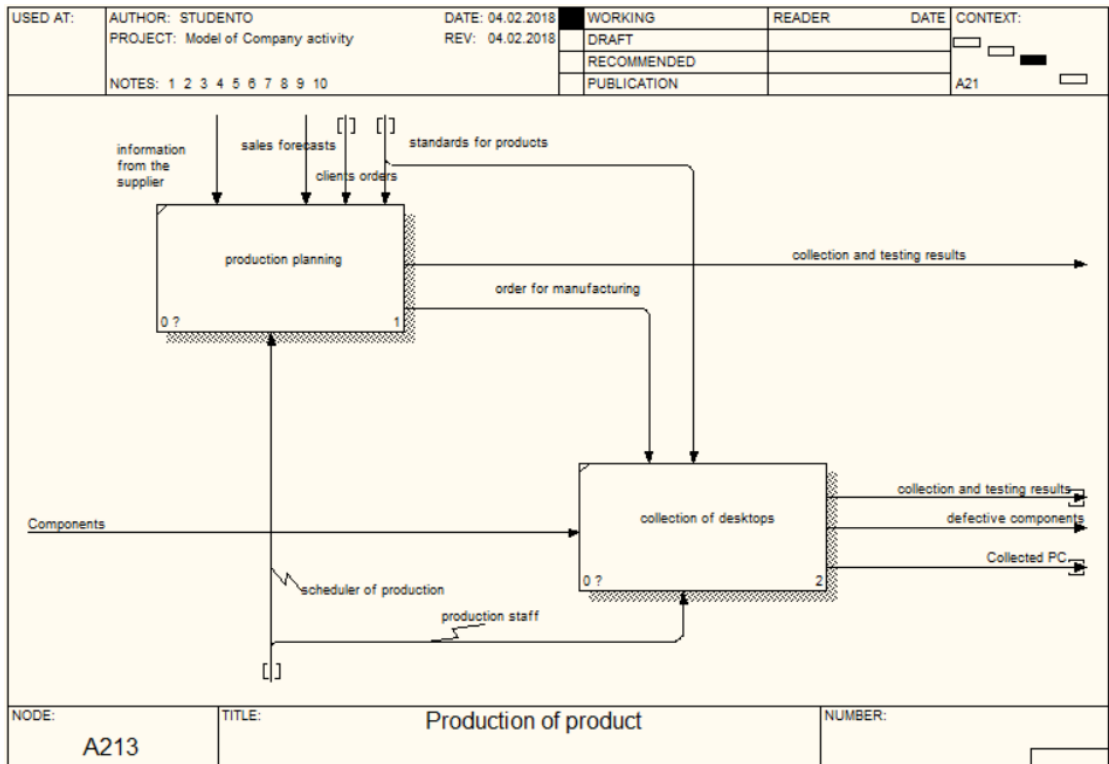


Рисунок 8.6 – Результат виконання завдання 1.2 (діаграма A3)

Модифікуйте діаграму IDEF3 «Збірка продукту» (завдання 1.4) відповідно з наведеною інформацією. Результат наведено на рис. 8.7.

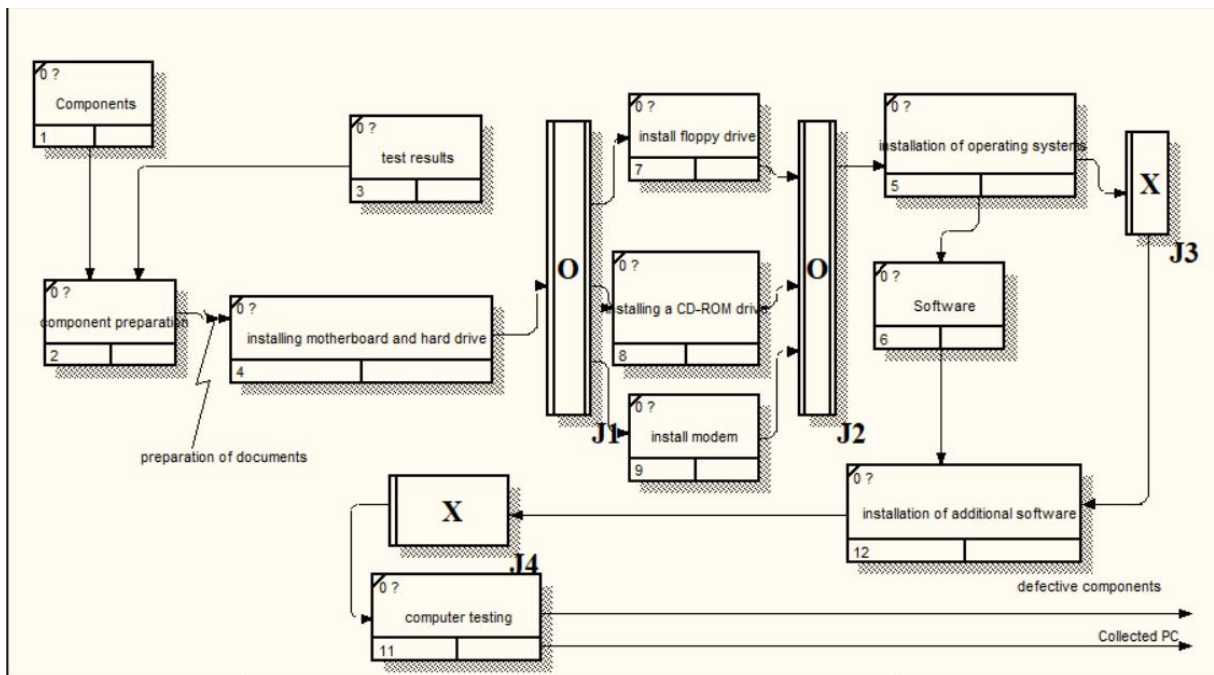


Рисунок 8.7 – Результат виконання завдання 1.4 (діаграма А3 2.1)

1.5 Декомпозиція роботи «Продажі та маркетинг»

Робота з продажу та маркетингу полягає у відповідях на телефонні дзвінки клієнтів, надання клієнтам інформації про ціни, оформленні замовлень, внесення замовлень в інформаційну систему і дослідженні ринку.

На основі цієї інформації декомпонуйте роботу «Продажі та маркетинг» (IDEF0).

Створіть наступні роботи: «Надання інформації про ціни»; «Оформлення замовлень»; «Дослідження ринку». Результат декомпозиції представлений на рис. 8.8.

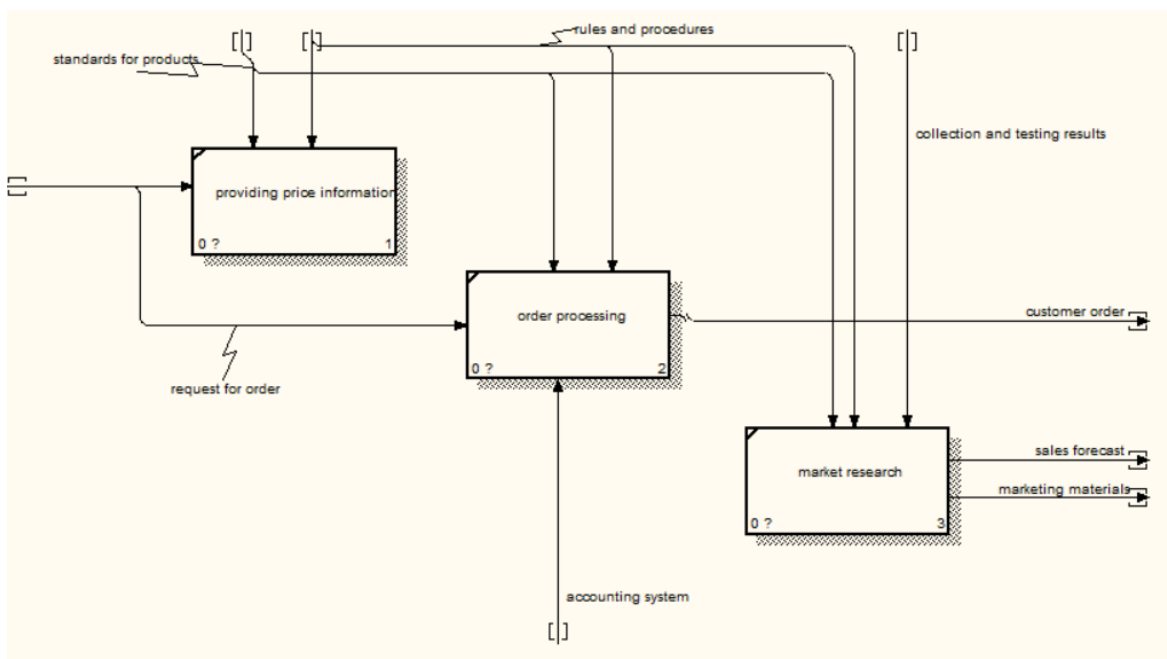


Рисунок 8.8 – Результат виконання завдання 1.5 (діаграма А 2)

Зміст звіту

1. Титульний аркуш.
2. Тема і мета роботи.
3. Короткі теоретичні відомості.
4. Протокол розв'язання завдання №1.
5. Протокол розв'язання завдання №2.
6. Висновки.

Контрольні питання

1. Визначте застосування діаграми потоків даних.
2. Розкрийте принцип побудови діаграм DFD.
3. Назвіть особливості діаграм DFD.
4. Як здійснюється нумерація об'єктів у діаграмі DFD?
5. Вкажіть основні елементи діаграми потоків даних.
6. Поясніть різницю між моделями IDEF0, DFD, IDEF3.
7. Опишіть сутність моделі DFD.
8. Поясніть створення міжсторінкового посилання.

ВИКОРИСТАНА ЛІТЕРАТУРА

1. Авраменко В. С., Авраменко А. С. Проектування інформаційних систем : навчальний посібник / за ред. В. С. Авраменко, А.С. Авраменко. Черкаси : Черкаський національний університет ім. Б. Хмельницького, 2017. 434 с.
2. Пасічник В. В., Литвин В. В., Шаховська Н. Б. Проектування інформаційних систем. Навчальний посібник. Львів : 2013. 380 с.
3. Томашевський О. М., Цегелик Г. Г., Вітер М. Б., Дудук В. І. Інформаційні технології та моделювання бізнес-процесів : навчальний посібник. К. : «Видавництво «Центр учбової літератури»», 2012. 296 с. URL: http://culonline.com.ua/Inform_tehn_ta_modeluv_biznes_procesiv.pdf (дата звернення 02.09.2020).
4. Соловьев И. В. Проектирование информационных систем / И. В. Соловьев, А. А. Майоров. М. : Академический Проект, 2009. 400 с.
5. Проектирование информационных систем : Курс лекций / за ред. Грекул В. И., Денищенко Г. Н., Коровкина Н. Л. М. : БИНОМ. Лаборатория знаний, 2008. 304 с. URL: <http://www.twirpx.com/file/12855/> (дата звернення 02.09.2020).
6. Похилько А. Ф. CASE-технология моделирования процессов с использованием средств BPWin и ERWin : учебное пособие /А. Ф. Похилько, И. В. Горбачев. Ульяновск : УЛГТУ, 2008. 120 с. URL: http://www.dut.edu.ua/uploads/1_768_96855158.pdf (дата обращения 28.08.2020).
7. Маклаков С. В. Моделирование бизнес-процессов с AllFusion Process Modeler (BPWin 4.1) / С. В. Маклаков. М. : ДИАЛОГ МИФИ, 2004. 240 с.
8. Маклаков С. В. Создание информационных систем с AllFusion Modeling Suite / С. В. Маклаков. М. : Диалог-МИФИ, 2003. 432 с.

ДОДАТОК А

**Міністерство освіти і науки України
Черкаський державний технологічний університет**

Факультет інформаційних технологій і систем

Кафедра інформаційних технологій проектування

Дисципліна:
«Проектування інформаційних систем»

З В І Т

**з лабораторної роботи № 1
Тема: «Вивчення програми AllFusion Process Modeler, інструментів
моделювання та основних фігур»**

студента групи WEB-1811
спеціальності 126 «Інформаційні системи та технології»

Шинкаренка Дмитра Олександровича

(Дата)

(Підпис студента)

Оцінка _____

Перевірено _____
(Дата)

Викладач _____ / Оксамитна Л.П. /
(Підпис) (Прізвище та ініціали)

Черкаси – 2020 р.

ДОДАТОК Б

З В І Т

про виконання завдань до лабораторної роботи № 1

Тема: Вивчення програми AllFusion Process Modeler, інструментів моделювання та основних фігур.

Мета роботи: Набути практичних навичок застосування інструментів засобу AllFusion Process Modeler для побудови контекстної діаграми бізнес-процесу.

- 1. Завдання для виконання.**
- 2. Протокол розв'язування завдань.**
- 3. Висновки:**

ЗМІСТ

ВСТУП	3
ЛАБОРАТОРНА РОБОТА №1	4
<i>Вивчення програми AllFusion Process Modeler, інструментів моделювання та основних фігур</i>	<i>4</i>
ЛАБОРАТОРНА РОБОТА №2.....	22
<i>Розробка моделі бізнес-процесу в нотації IDEF0. Створення діаграми декомпозиції.....</i>	<i>22</i>
ЛАБОРАТОРНА РОБОТА №3	40
<i>Створення діаграми дерева вузлів та FEO діаграми.....</i>	<i>40</i>
ЛАБОРАТОРНА РОБОТА №4.....	45
<i>Злиття і розщеплення моделей.....</i>	<i>45</i>
ЛАБОРАТОРНА РОБОТА №5.....	58
<i>Створення діаграм та сценарію з використанням стандарту функціонального моделювання IDEF0.....</i>	<i>58</i>
ЛАБОРАТОРНА РОБОТА № 6.....	66
<i>Вартісний аналіз (Activity Based Costing). Використання категорій UDP.....</i>	<i>66</i>
ЛАБОРАТОРНА РОБОТА №7.....	79
<i>Створення діаграми потоків даних DFD заданої предметної області.....</i>	<i>79</i>
ЛАБОРАТОРНА РОБОТА №8.....	87
<i>Створення моделі TO-BE.....</i>	<i>87</i>
ВИКОРИСТАНА ЛІТЕРАТУРА.	96
ДОДАТКИ	97