

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

Методичні рекомендації

до виконання лабораторної роботи
з теми "Використання CASE-технологій у проектуванні
інформаційних систем та БД"
навчальної дисципліни

"ІНФОРМАЦІЙНІ СИСТЕМИ
МІЖНАРОДНОГО БІЗНЕСУ"

для студентів напряму підготовки
"Міжнародна економіка"
всіх форм навчання

Харків. Вид. ХНЕУ, 2008

Затверджено на засіданні кафедри інформаційних систем.
Протокол №1 від 28.08.2007 р.

M54 Методичні рекомендації до виконання лабораторної роботи з теми "Використання CASE-технологій у проектуванні інформаційних систем та БД" навчальної дисципліни "Інформаційні системи міжнародного бізнесу" для студентів напряму підготовки "Міжнародна економіка" всіх форм навчання / Укл. Р. М. Чен. — Харків: Вид. ХНЕУ, 2008. — 52 с. (Укр. мов.)

Призначені для практичного вивчення питань організації інформаційних систем та баз даних з використанням засобів автоматизації проектування, призначених для моделювання бізнес-планів на підприємстві.

У систематизованому вигляді наводяться необхідні теоретичні відомості про побудову діаграми потоків даних (Data flow diagramming, DFD), які використовуються для опису документообігу й обробки інформації. Розглянуто можливість експорту даних з VPwin у ERwin і далі в СУБД MS Access.

Пропоновані матеріали можуть бути корисними для студентів і аспірантів, що вивчають і використовують сучасні новітні інформаційні технології для автоматизації процесів моделювання й проектування інформаційних систем і баз даних для підприємств та організацій.

Вступ

Широке використання назараді різних засобів автоматизації проектування інформаційних систем потребує від користувачів необхідності вивчення й упровадження в практику інструментальних засобів, які базуються на рішеннях фірми Computer Association, а именно: CASE-засобі AllFusion Process Modeler (BPwin), призначеному для проведення аналізу і реорганізації бізнес-процесів на підприємстві, а також AllFusion ERwin Data Modeler (ERwin), призначеному для побудови логічної та фізичної моделей даних. У пакеті AllFusion Modelling Suite реалізовано механізм двонаправленого зв'язку (BPwin-ERwin), що дозволяє побудувати логічну та фізичну моделі даних на основі функціональної моделі.

Наскрізне проектування інформаційної системи (ІС) від обстеження підприємства до створення моделі даних для реалізації в конкретній СУБД дає найбільш ефективні результати стосовно експлуатації й реінжинірингу ІС. Тому в даних методичних рекомендаціях і подані питання організації інформаційної системи та баз даних з використанням засобів автоматизації проектування, призначених для моделювання бізнес-процесів зовнішньоекономічної діяльності.

Ці методичні рекомендації спрямовані на вивчення CASE-засобу структурно-функціонального моделювання BPwin з використанням стандарту DFD. Структурно-функціональний підхід заснований на декомпозиції системи (її розбитті) на елементарні функції, тобто на функціональні підсистеми, котрі в свою чергу підрозділяються на підфункції, завдання тощо. Процес розбиття продовжується до конкретних процедур, котрі вже не має сенсу декомпонувати на операції. При цьому система зберігає цілісне уявлення, в якому всі складові компоненти взаємопов'язані.

Діаграми потоків даних (Data flow diagramming, DFD) представляють модульовану систему як мережу зв'язаних між собою робіт і використовуються для опису документообігу й обробки інформації. DFD розглядає систему як сукупність предметів, тому контекстна діаграма часто включає роботи, зовнішні посилання, а також сховища даних. Розглянуто можливість експорту сутностей та атрибутів, котрі створюються в словниках BPwin, в ERwin. В ERwin на базі імпортованих сутностей створюються логічна та фізична моделі даних, які використовуються для автоматичної генерації схеми даних у СУБД Ms Access.

1. Основні принципи моделювання на базі стандартів iso 9001:2000¹ і erp

ISO 9000 – серія міжнародних стандартів ISO, що регламентують управління якістю на підприємствах. Система стандартів розроблена Міжнародною організацією по стандартизації (ISO, International Organization for Standardization), що ґрунтувалася на розробках Британського інституту стандартів BS 5750. Стандарти ISO 9000, прийняті більш ніж 90 країнами світу, застосовні до будь-яких підприємств, незалежно від їхнього розміру і сфери діяльності. Фактично сертифікація виробляється не по ISO 9000, а по специфікації ISO 9001:2000.

Сертифікат ISO 9000 необхідний підприємствам:

працюючим на міжнародних ринках або з міжнародними постачальниками, що вимагають наявності такого сертифіката;

працюючим в секторах економіки, регульованих урядом, або з урядовими організаціями країн, у яких наявність сертифіката ISO 9000 є обов'язковим.

У дійсності ISO 9000 поєднує п'ять стандартів:

ISO 9001 – Модель контролю якості в проектуванні, розробці, виробництві, монтажі й обслуговуванні

ISO 9002 – Модель контролю якості в області виробництва, монтажу й обслуговування (скасований у редакції 2000 року)

ISO 9003 – Модель для забезпечення якості при контролі й іспитах готової продукції (скасована в редакції 2000 року)

і вказівки за реєстраційними стандартами:

ISO 9000-1 – Загальні посібники з вибору і використання

ISO 9004-2 – Управління якістю й елементи системи якості – вказівки

Кінцеві цифри в позначенні версії стандарту відповідають року прийняття, наприклад:

ISO 9000 : 1994, ISO 9000 : 2000

1.1. Моделювання як основа Business Process Improvement (BPI)

Діяльність на підприємстві можна пропонувати наступною класифікацією:

А – первинна діяльність: виготовлення конкретних виробів (наприклад, автомобілів) і сервіс (наприклад, гарантійне або післягарантійне обслуговування);

В – вторинна діяльність, спрямована на поліпшення основної діяльності групи А;

С – діяльність, спрямована на вдосконалювання діяльності групи В.

Очевидно, що критичною діяльністю для підприємства є діяльність групи А, але без наявності діяльності груп В и С в умовах конкурентного ринку неможливо забезпечити шанс одержання прибутку в майбутньому (підвищення потенціалу підприємства). У рамках розглянутих груп діяльності підприємства використовуються моделі, представлені на рис.1.1.

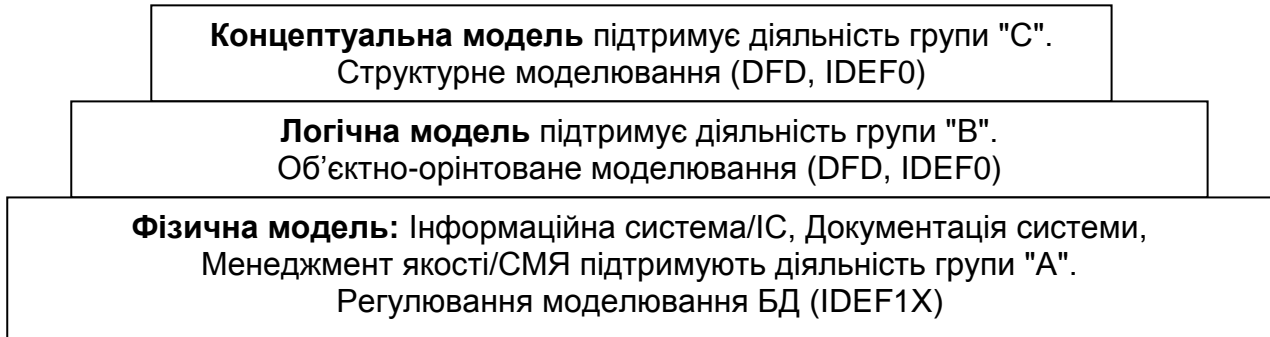


Рис. 1.1. Моделі діяльності підприємства

Діяльність групи **А** базується на фізичній платформі, під якою розуміються ІС підприємства і система документації менеджменту якості. Діяльність групи **В** зводиться до аналізу і проектування функціонування підприємства. Діяльність групи **С** ґрунтується на концепціях, що включають світовий досвід стосовно керування підприємством (ERP стандарт і стандарт ISO 9001:2000). Діяльність групи **С** базується на концептуальній моделі.

Вище перераховані групи діяльності підприємства припускають взаємозв'язок. Таким чином, моделі, використовувані в рамках даних груп діяльності підприємства, також повинні бути взаємозалежні (рис. 1.2).



Рис. 1.2. Трьохрівнева архітектура ІС підприємства

Концептуальна модель зв'язується з логічною моделлю через використання шаблону на етапі створення логічної моделі. Логічна модель також має зв'язки з фізичною моделлю. AllFusion Modelling Suite дозволяє перевести класи логічної моделі в реляційну модель IDEF1X (ERwin). Засобами ERwin формується опис структури бази даних, що перетворюється в базу даних. Засіб моделювання AllFusion Modelling Suite формує з логічної моделі звіти за заздалегідь описаним шаблоном з дотриманням бізнес-логіки підприємства. Логічна модель може бути створена на базі об'єктно-орієнтованого підходу:

через прямий (Forward Engineer) або зворотний інжиніринг (Reverse Engineer) відображаються в компоненті представлення (інтерфейс ІС);

через опис бізнес-логіки відображається в документації система менеджменту якості (СМЯ) підприємства в серверній частині ІС (вона функціонує з використанням сервера додатків);

через реляційну модель відображається в базі даних ІС.

Таким чином, відбувається стикування трьохрівневої архітектури ІС з бізнес-моделюванням діяльності підприємства.

При комплексній реалізації трьох видів діяльності підприємства (А, В і С) реалізується безперервне підвищення потенціалу підприємства або, іншими словами, безперервне поліпшення бізнес-процесів ВРІ.

У рамках циклу ВРІ головна роль приділяється логічному моделюванню. Логічна модель (розроблена на базі шаблону), включає як поточну діяльність, так і бажану (систему бізнес-процесів) для підприємства. Бажана система бізнес-процесів розробляється і досягається з використанням концептуальної моделі. Концептуальна модель базується на передовому досвіді керування підприємством (ISO 9001:2000, ERP, CALS, GAAP, MRP II і т.д.) і створюється за допомогою методологій структурного аналізу. Концептуальна модель (DFD і IDEF0 діаграми) не розробляється підприємством. Дана модель є продуктом зовнішніх консультантів.

1.2. Значимість концептуальної моделі (стандарти ERP і ISO 9001:2000)

Підприємство забезпечує безперервне поліпшення бізнес-процесів за рахунок чіткої організації бажаних бізнес-процесів або, іншими словами, за рахунок чіткої реорганізації поточних бізнес-процесів у бажані. Графічне представлення даних концепцій (DFD і IDEF0 діаграми) полегшує їх наочність і розуміння.

IDEF0 і DFD діаграми описують діяльність підприємства відповідно до вимог стандартів ERP і ISO 9001:2000. IDEF0 діаграма описує діяльність груп А і В (на базі стандартів ISO 9001:2000 і ERP). DFD діаграма описує діяльність групи А (на базі стандарту ERP). Дані діаграми використовуються як на стадії моделювання бажаної системи бізнес-процесів, так і на стадії ухвалення рішення про необхідність моделювання на базі зазначених вище стандартів (підприємства нерідко використовують DFD і IDEF діаграми для наочного представлення функціонування підприємства у відповідності зі стандартами ERP і ISO 9001: 2000).

1.3. DFD-діаграма

DFD-діаграма (модель потоків даних) відбиває світовий досвід щодо керування промисловими підприємствами.

В основі даної методології – методології Gane/Sarson (нотація Гейна-Сарсона) – лежить побудова моделі аналізованої ІС-проектованої або реально існуючої. Відповідно до методології модель системи визначається як ієрархія діаграм потоків даних (ДПД або DFD), що описують асинхронний процес перетворення інформації від її введення в систему до видачі користувачеві. Діаграми верхніх рівнів ієрархії (контекстні діаграми) визначають основні процеси або підсистеми ІС із зовнішніми входами та виходами. Модель деталізуються за допомогою діаграм нижнього рівня. Така декомпозиція продовжується, створюючи багаторівневу ієрархію діаграм, доти, доки не буде досягнуто такого рівня декомпозиції, на якому процеси стають елементарними, і деталізувати їх далі неможливо.

Джерела інформації (зовнішні сутності) породжують інформаційні потоки (потоки даних), що переносять інформацію до підсистем або процесів. Ті у свою чергу перетворюють інформацію і породжують нові потоки, що переносять інформацію до інших процесів або підсистем, нагромаджувачам даних або зовнішніх сутностей – споживачам інформації. Таким чином, основними компонентами діаграм потоків даних є:

- зовнішні сутності;
- системи і підсистеми;
- процеси;
- сховище даних;
- потоки даних.

Зовнішні сутності. Зовнішня сутність становить матеріальний предмет або фізичну особу, що представляє собою джерело або приймач інформації: наприклад, замовники, персонал, постачальники, клієнти, склад. Визначення деякого об'єкта або системи як зовнішньої сутності вказує на те, що вона перебуває за межами аналізованої ІС. У процесі аналізу деякі зовнішні сутності можуть бути перенесені всередину діаграми аналізованої ІС, якщо це необхідно, або, навпаки, частина процесів ІС може бути винесена за межі діаграми і представлена як зовнішня сутність.

Зовнішня сутність позначається квадратом, розташованим ніби "над" діаграмою, що кидає на неї тінь, для того, щоб можна було виділити цей символ серед інших позначень.

Системи і підсистеми. При побудові моделі складної ІС вона може бути представлена на так званій контекстній діаграмі у вигляді однієї системи як єдиного цілого або декомпозована на ряд підсистем.

Номер підсистеми може служити для її ідентифікації. У поле імені вводиться найменування підсистеми у вигляді пропозиції з підметом і відповідними визначеннями і доповненнями.

Процеси. Процес є перетворенням вхідних потоків даних у вихідні відповідно до визначеного алгоритму. Фізично процес може бути реалізований різними способами: це може бути підрозділ організації (відділ), що виконує обробку вхідних документів і випуск звітів, програма, апаратно реалізований логічний пристрій тощо.

Номер процесу може служити для його ідентифікації. У поле імені вводиться найменування процесу у вигляді напису з активним недвозначним дієсловом у невизначеній формі (обчислити, розрахувати, перевірити, визначити, створити, одержати), за яким указуються іменники в знахідному відмінку, наприклад:

- зібрати зведення про клієнтів;
- видати інформацію про укладені контракти;
- перевірити кредитоспроможність клієнта.

Використання таких дієслів, як "обробити", "модернізувати" або "відредагувати", означає, як правило, недостатньо глибоке розуміння даного процесу і вимагає подальшого аналізу.

Інформація в полі фізичної реалізації показує, який підрозділ організації, програма або апаратний пристрій виконує даний процес.

Сховище даних. Сховище даних становить собою абстрактний пристрій для збереження інформації, який можна в будь-який момент помістити в сховище і через якийсь час отримати, причому способи переміщення і отримання можуть бути будь-якими. Сховище даних може бути реалізовано фізично у вигляді шухляди в картотеці, таблиці в оперативній пам'яті, файлу на магнітному носії і т. д.

Сховище даних ідентифікується буквою "D" і довільним числом. Ім'я сховища вибирається з розуміння найбільшої інформативності для проєктувальника. Сховище даних у загальному випадку є прообразом майбутньої бази даних і опису даних, які в ньому зберігаються, повинен бути ув'язаний з інформаційною моделлю.

У матеріальних системах сховища даних зображаються там, де об'єкти чекають обробки, наприклад у черзі. У системах обробки інформації сховища даних є механізмом, який дозволяє зберегти дані для подальших процесів.

Потоки даних. Потік даних визначає інформацію, передану через деяке з'єднання від джерела до приймача. Реальний потік даних може бути інформацією, переданою по кабелю між двома пристроями, що пересилаються поштою аркушами, магнітними стрічками або дискетами, мережею з одного комп'ютера на інший і т. ін.

Потік даних на діаграмі зображується лінією, що закінчується стрілкою, яка показує напрямок потоків. Кожен потік даних має ім'я, що відбиває його зміст.

Оскільки в DFD кожна сторона роботи не має чіткого призначення, як в IDEF0, стрілки можуть входити і виходити з будь-якої грані прямокутника роботи. У DFD також застосовуються двонаправлені стрілки для опису діалогів типу команди-відповіді між роботами, між роботою і зовнішньою сутністю і між зовнішніми сутностями.

2. Використання діаграм потоків даних (dfd) для опису предметної області

Діаграми потоків даних (Data flow diagramming, DFD) в BPwin використовуються для опису потоків даних і інформаційних потоків для заданої предметної області. Для побудови концептуальної моделі буде використовуватися бізнес-процес, зв'язаний із проходженням митних формальностей при веденні зовнішньоекономічних операцій фірми, тому назва моделі буде: зовнішньоекономічна діяльність-митниця.

Для того, щоб побудувати діаграму DFD, потрібно запустити пакет BPwin, створити новий документ і у вкладці, що відкрилася, "кликнути" по радіо-кнопці **Data Flow (DFD)**, у поле **Name** занести назву моделі "Зовнішньо-економічна діяльність-митниця" (рис. 2.1), клацнути на кнопці "OK", що приведе до відкриття стартового вікна BPwin для моделі DFD (рис. 2.2).

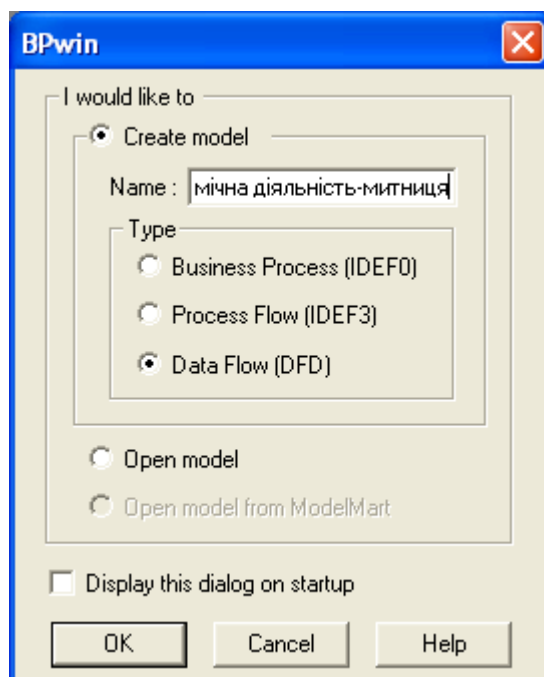




Рис. 2.1. Вибір типу моделі

У палітрі інструментів на новій діаграмі DFD знаходяться кнопки:

 – додати в діаграму зовнішнє посилання (**External Reference**). Зовнішнє посилання є джерелом або приймачем даних ззовні моделі.

 – додати в діаграму сховище даних (**Data store**). Сховище даних дозволяє описати дані, які необхідно зберегти в пам'яті перш ніж використовувати в роботах.

Стрілки діаграми DFD показують, як об'єкти, включаючи дані, рухаються від однієї роботи до іншої. Це представлення потоків спільно зі сховищами даних і зовнішньою сутністю робить моделі DFD більш схожими на фізичні характеристики системи – рух об'єктів (**data flow**), зберігання об'єктів (**data stores**), постачання і розповсюдження об'єктів (**external entities**).

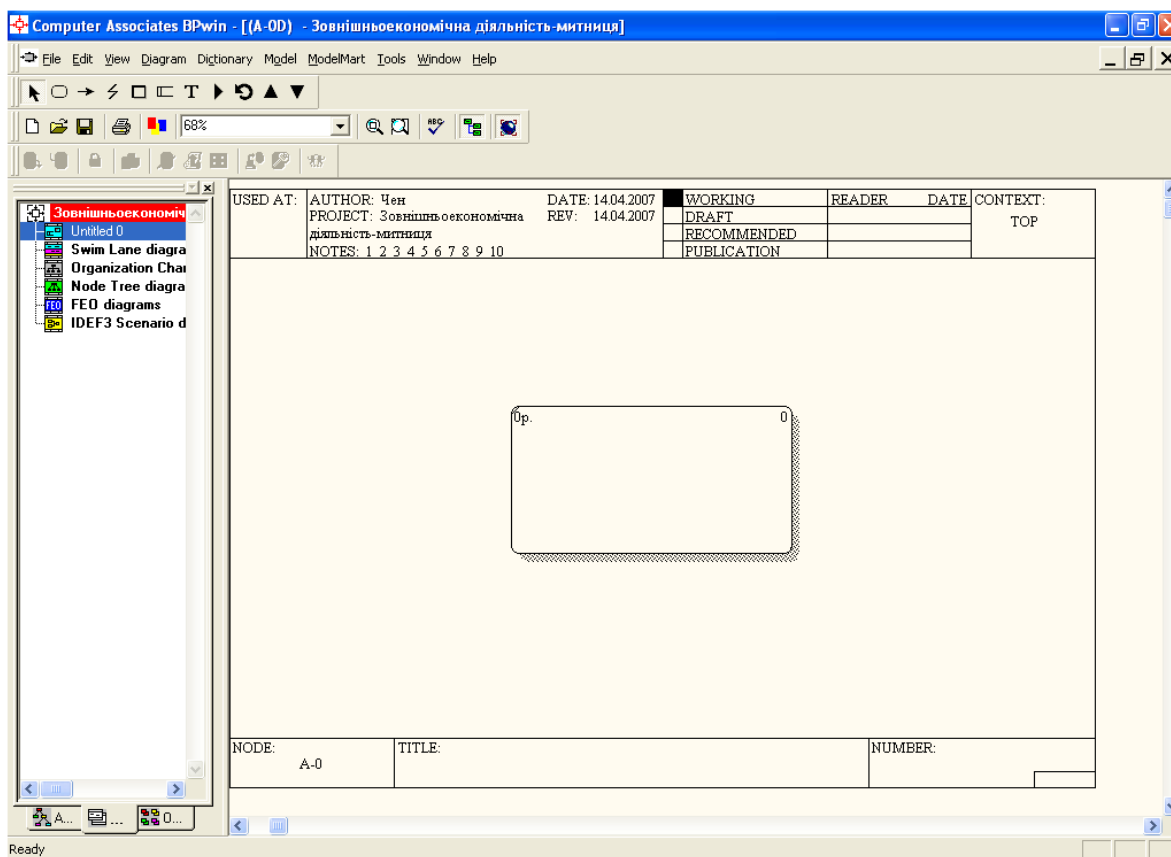


Рис. 2.2. Стартове вікно для побудови моделі DFD

DFD розглядає систему як сукупність предметів. Тому контекстна діаграма включає роботи і зовнішні посилання. Роботи звичайно іменуються за назвою системи, наприклад "**Система ведення довідників**". Включення зовнішніх посилань в контекстну діаграму не відмінняє вимоги методології чітко визначити мету, область і єдину точку зору на модельовану систему.

Роботи. У DFD роботи є функціями системи, що перетворюють входи у виходи. Роботи зображаються прямокутниками з округляючими кутами, вони мають входи і виходи, але не підтримують управління і механізми, як IDEF0.

Зовнішня сутність. Зовнішня сутність зображає входи в систему і/або виходи з системи. Зовнішня сутність зображається у вигляді прямокутника з тінню і звичайно розташовується по краях діаграми (рис. 2.3). Одна зовнішня сутність може бути використана багато разів на одній або декількох діаграмах. Звичайно такий прийом застосовують, щоб не малювати занадто довгих і заплутаних стрілок.

Стрілки (потоки даних). 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 і виходити з будь-якої грані прямокутника роботи. У DFD також застосовуються двонаправлені стрілки для опису діалогів типу команди-відповіді між роботами, між роботою і зовнішньою сутністю і між зовнішніми сутностями (рис. 2.3).

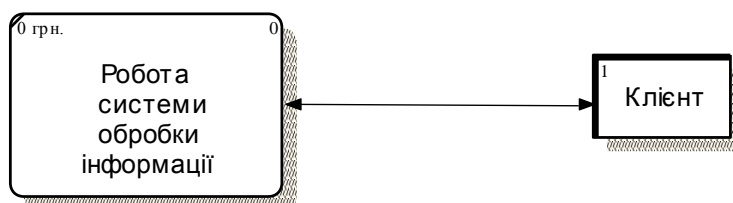


Рис. 2.3. Зовнішня сутність

Сховище даних. На відміну від стрілок, що описують об'єкти в русі, сховища даних зображають об'єкти у спокої (рис. 2.4).

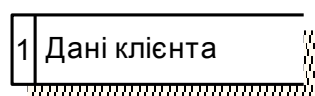


Рис. 2.4. Сховище даних

Злиття і розгалуження стрілок. У DFD стрілки можуть зливатися і розгалужуватися, що дозволяє описати декомпозицію стрілок. Кожен новий сегмент стрілки, що зливається або розгалужується, може мати власне ім'я.

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

Альтернативним підходом є підхід, популярний при створенні програмного забезпечення, котрий називається 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 зовнішніх сутностей. Модель оточення звичайно містить опис мети системи, одну контекстну діаграму і список подій. Контекстна діаграма містить один прямокутник роботи, що зображає систему в цілому, і зовнішні сутності, з якими система взаємодіє.

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

Одержані діаграми можуть бути перетворені з метою більш наочного представлення системи, зокрема роботи на діаграмах можуть бути декомповані.

Нумерація об'єктів. У DFD номер кожної роботи може включати префікс (A), номер батьківської роботи і номер об'єкта. Номер об'єкта – це унікальний номер роботи на діаграмі. Наприклад, робота може мати номер A.12.4. Унікальний номер мають сховища даних і зовнішні сутності незалежно від їх розташування на діаграмі. Кожне сховище даних має префікс D і унікальний номер, наприклад D5. Кожна зовнішня сутність має префікс E і унікальний номер, наприклад E5. Префікси можуть на діаграмах не відобразитися.

Хід виконання роботи. Діаграми будуються аналогічно вищезазначеним роботам. У нашому прикладі ми збудуємо модель "як має бути" в стандарті DFD.

Метою проекту є автоматизація рішення задачі "Ведення зовнішньоекономічних операцій фірми" з метою чинного проходження митних формальностей при експорті-імпорту товарів. При цьому дане завдання перетинається з іншими завданнями зовнішньоекономічної діяльності фірми.

Для завдання імені основної роботи моделюємого бізнес-процесу необхідно двічі клацнути по прямокутнику (поки ще "порожньому"), потім у вікні "**Activity Properties**", що відкрилося, записати назву задачі (рис. 2.5).

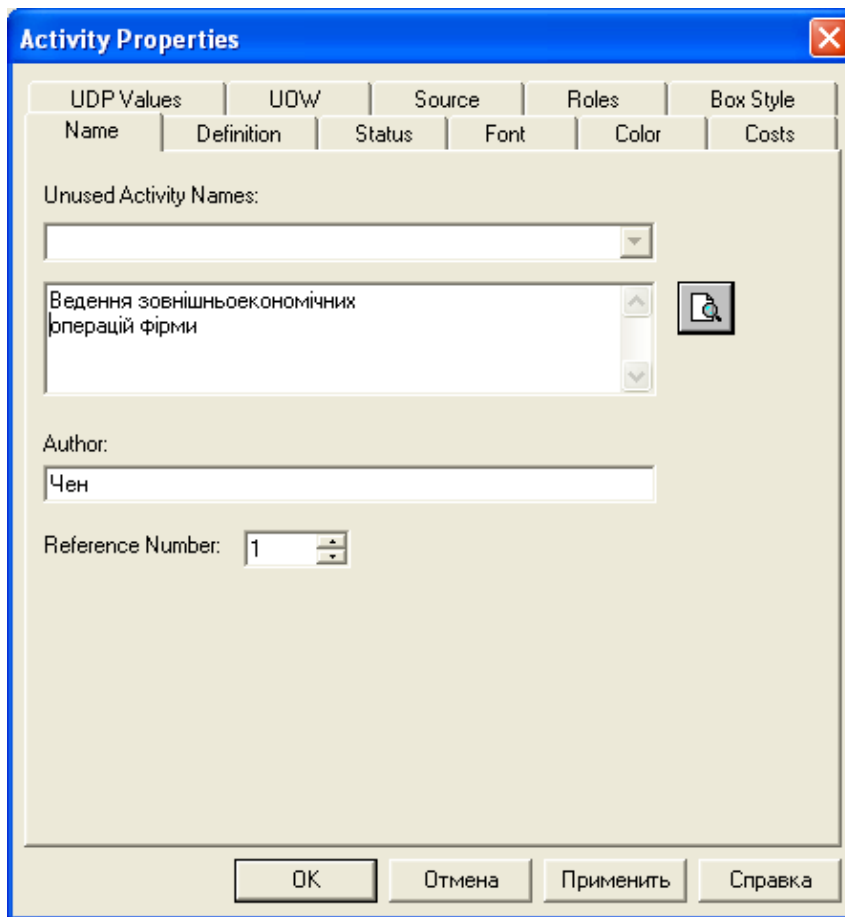


Рис. 2.5. Завдання назви роботи в вкладці Activity Properties

Для зміни розміру і типу шрифту необхідно скористатися вкладкою **Font** у цьому ж вікні (рис. 2.6).

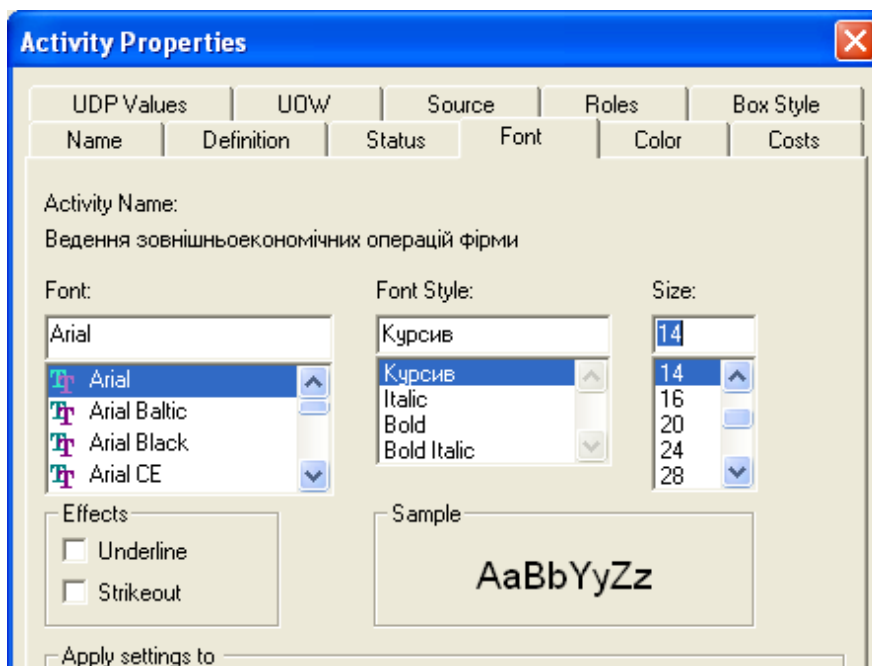


Рис. 2.6. Зміна розміру і типу шрифту

У результаті одержимо наступний вигляд діаграми (рис. 2.7)

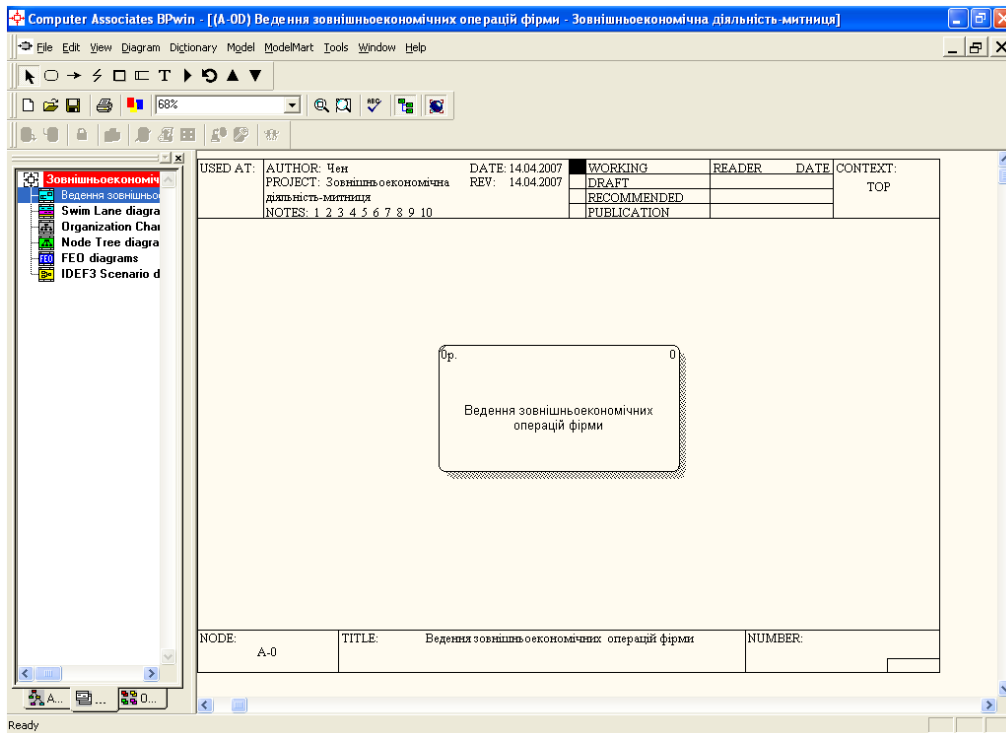


Рис. 2.7. Назва зовнішньої сутності

Послідовність подальших дій для побудови моделі показана на рис. 2.8 – 2.15.

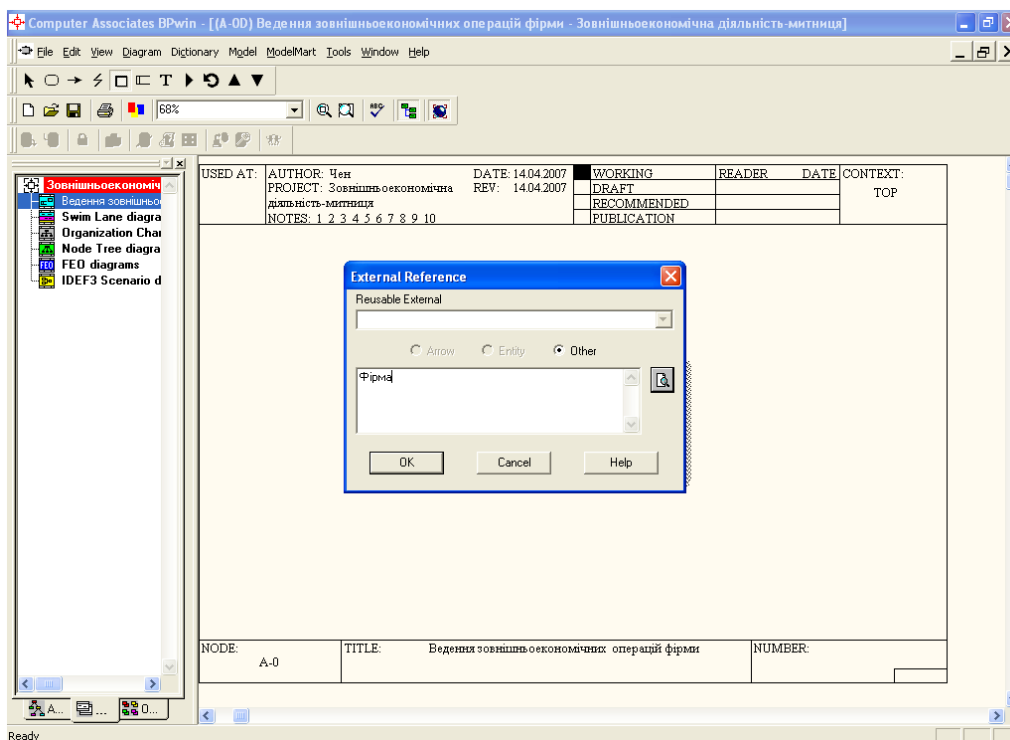


Рис. 2.8. Завдання зовнішньої сутності у вікні External Reference

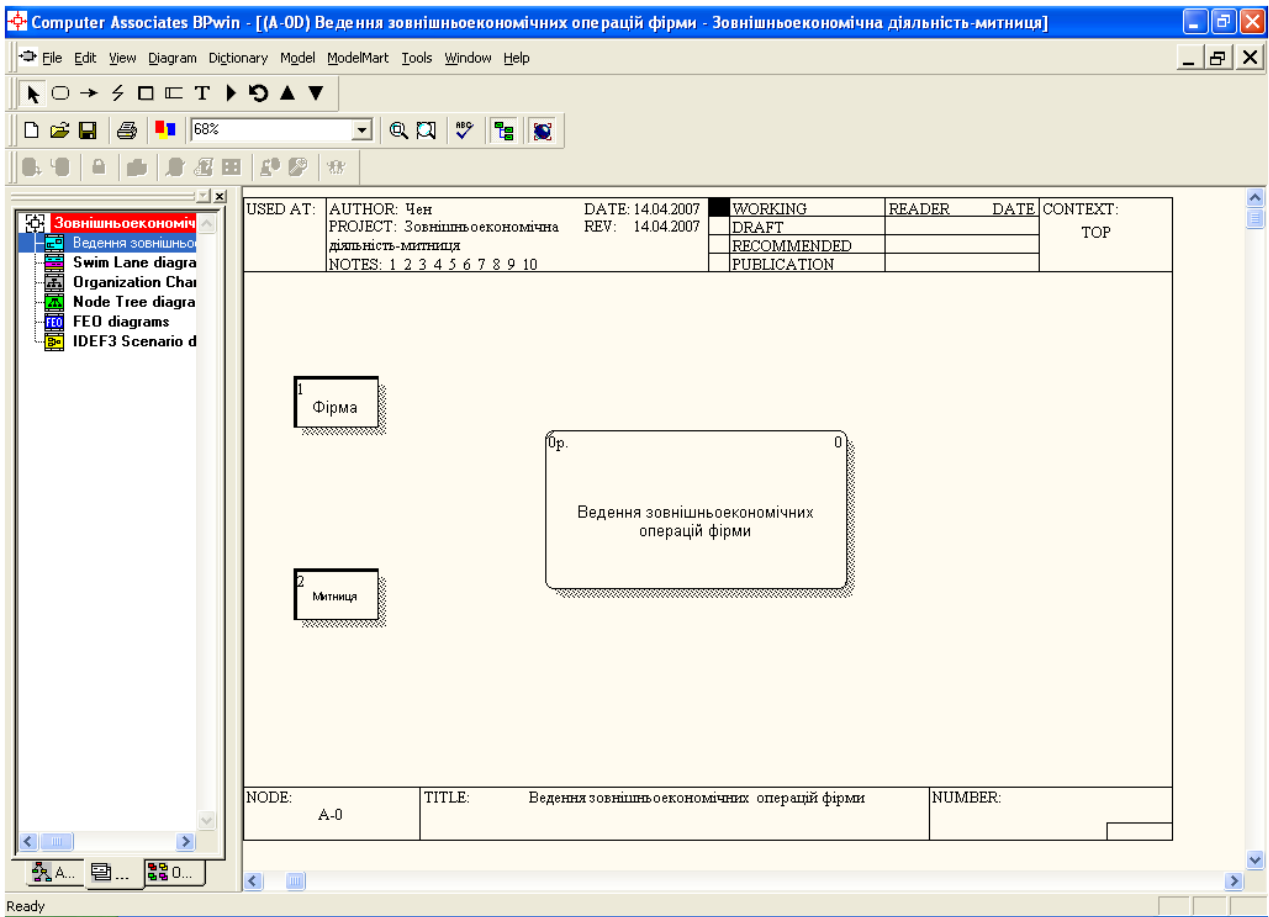




Рис. 2.9. Розміщення зовнішніх сутностей на діаграмі

Для побудови інтерфейсної дуги (стрілки) необхідно:

- Вибрати кнопку із символом стрілки  (**Precedence Arrow Tool**) у палітрі інструментів і перенести курсор до лівої сторони екрана, поки не з'явиться початкова темна смужка.
 - Клацнути один раз по смужці, відкіля виходить стрілка, і ще раз у лівій частині роботи з бокового входу, де закінчується стрілка.
 - Повернутися в палітру інструментів і вибрати опцію редагування стрілки .
 - Клацнути правою кнопкою миші на лінії стрілки й у спливаючий меню вибрати пункт **Name**, де і додати ім'я стрілки.
- Аналогічно будуються інші стрілки (рис.2.10).

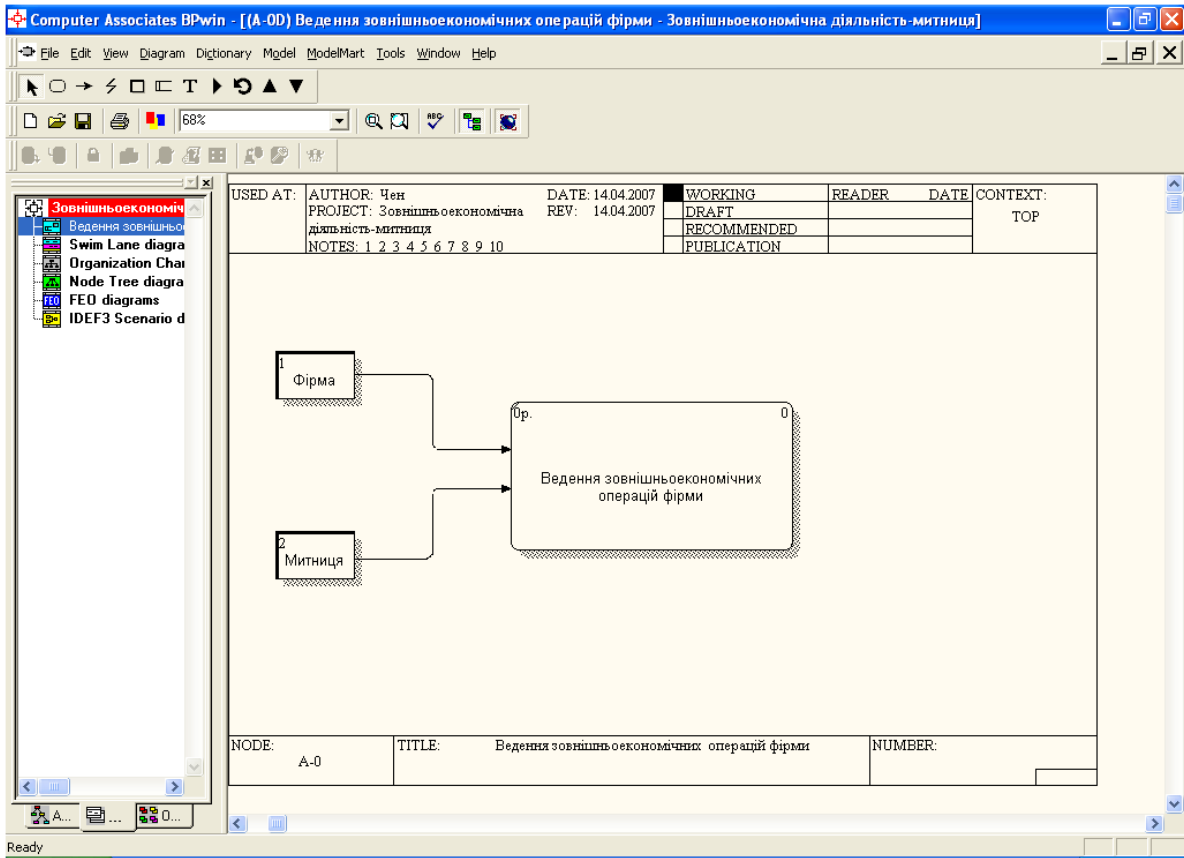


Рис. 2.10. Побудова інтерфейсних дуг

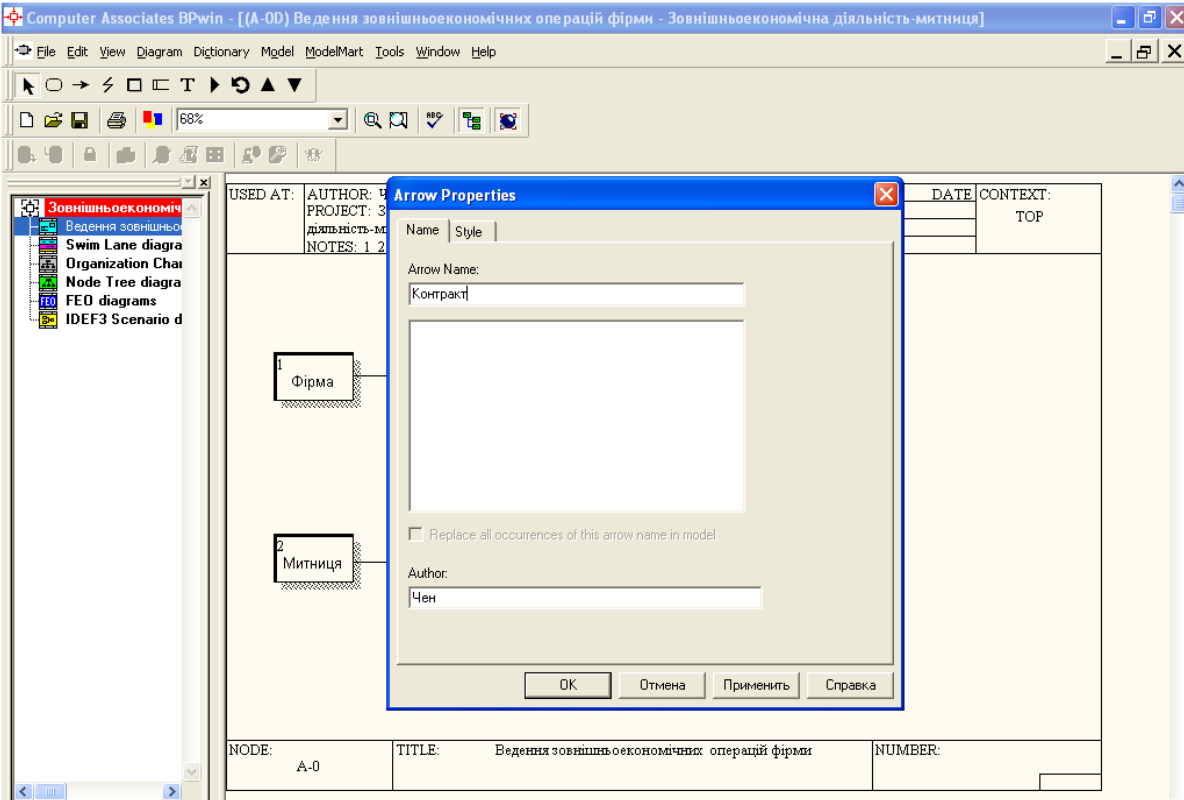


Рис. 2.11. Завдання імені стрілки Контракт у вікні Arrow Properties

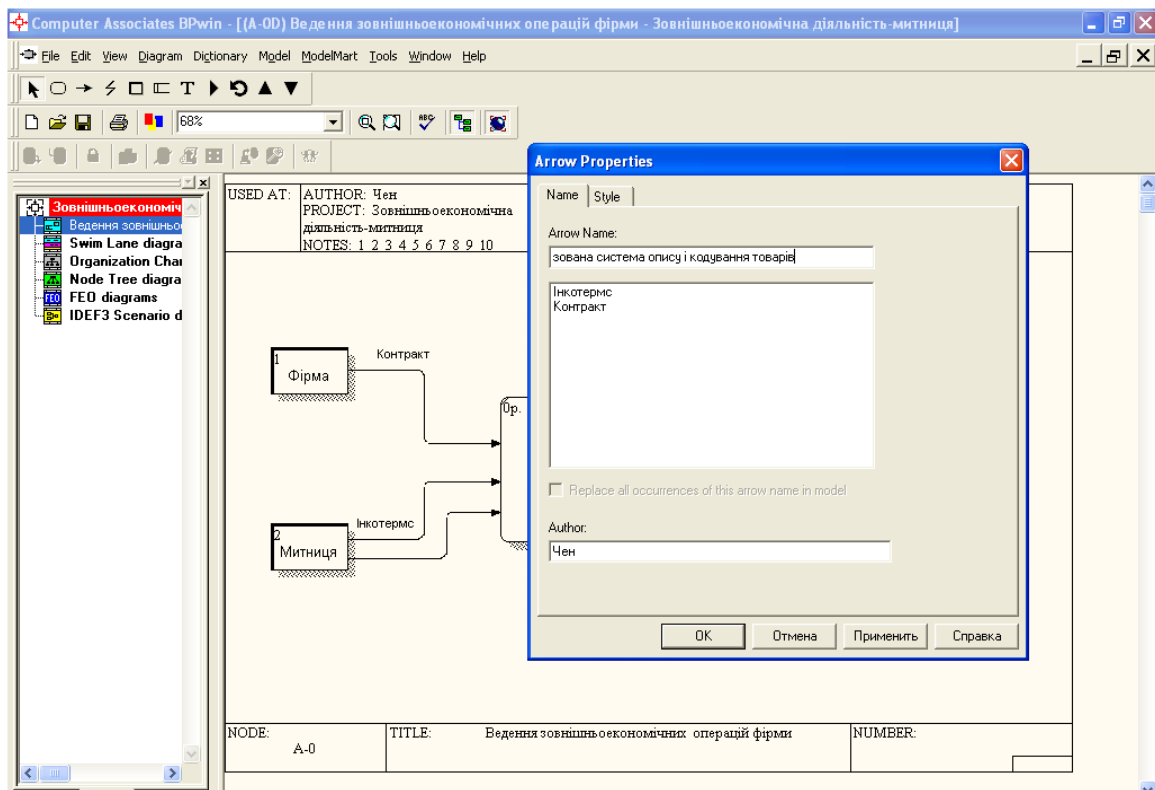


Рис. 2.12. Підпис імені другої стрілки Інкотермс

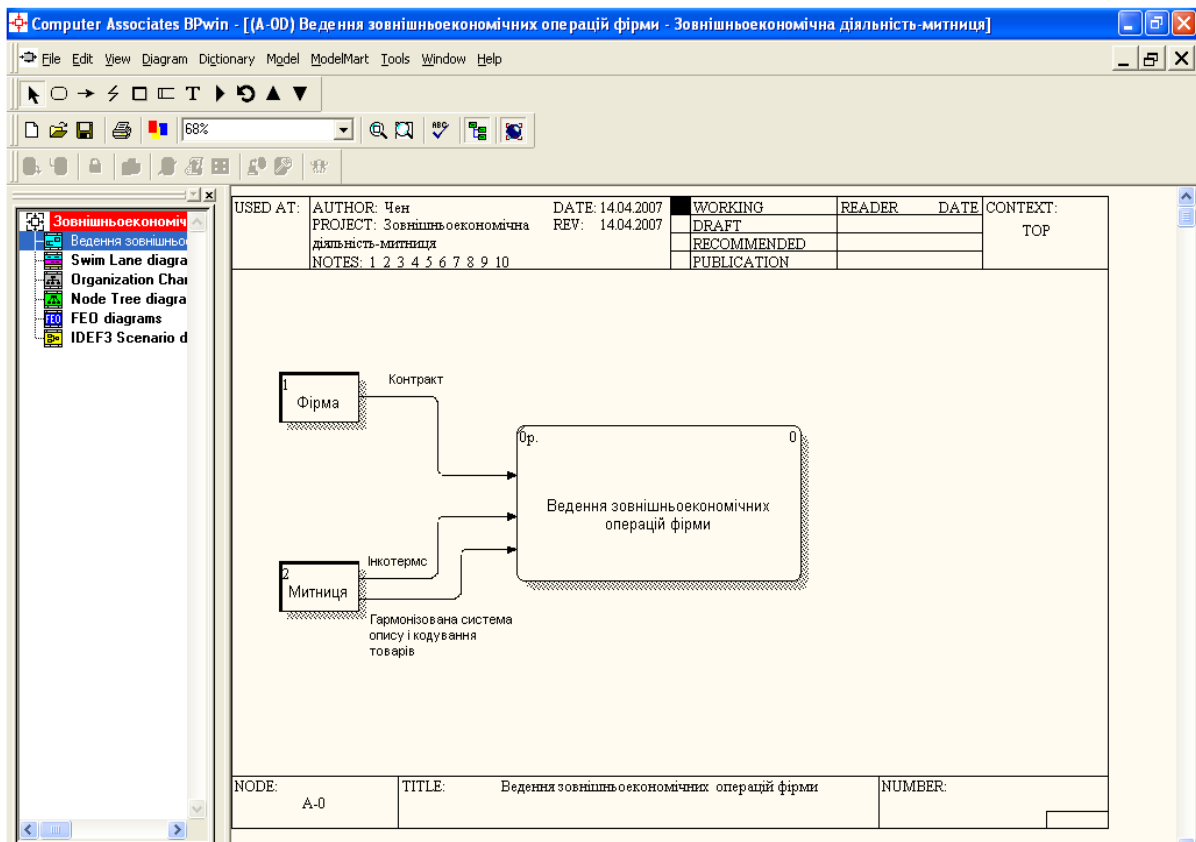


Рис. 2.13. Побудова вхідних потоків даних (стрілки та їх імена)

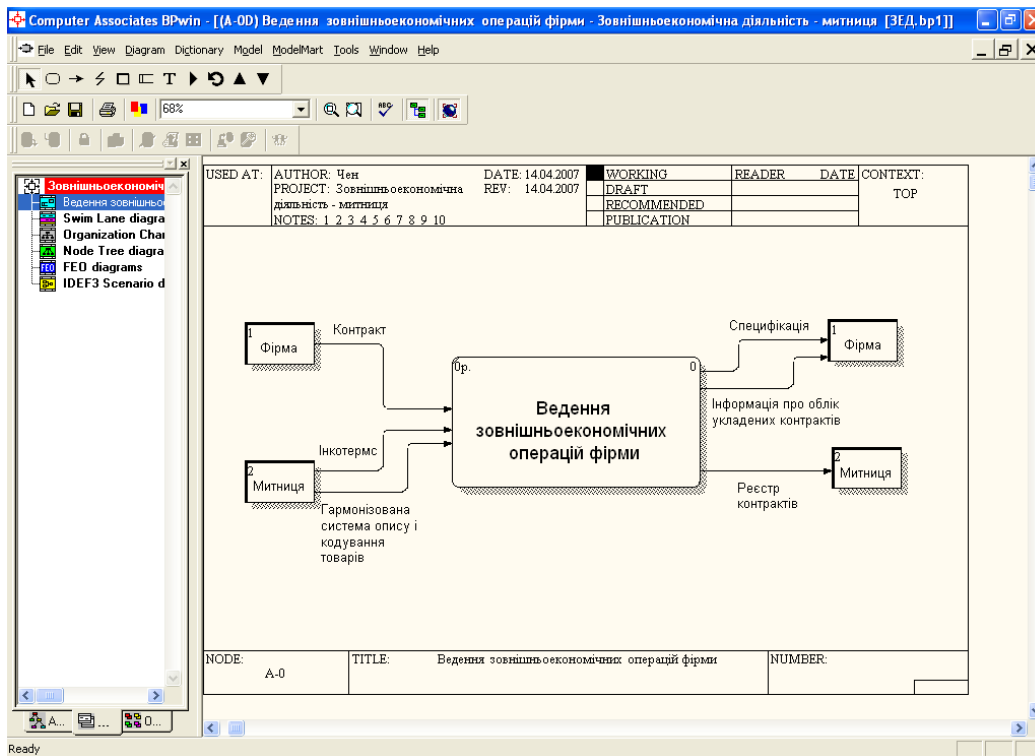


Рис. 2.14. Остаточний вигляд контекстної діаграми в стандарті DFD, побудований за допомогою CASE-засобу BPwin

Для збереження проектної версії DFD-діаграми в текстовому документі необхідно скористатися командою **Edit→Copy Picture** (рис. 2.15).

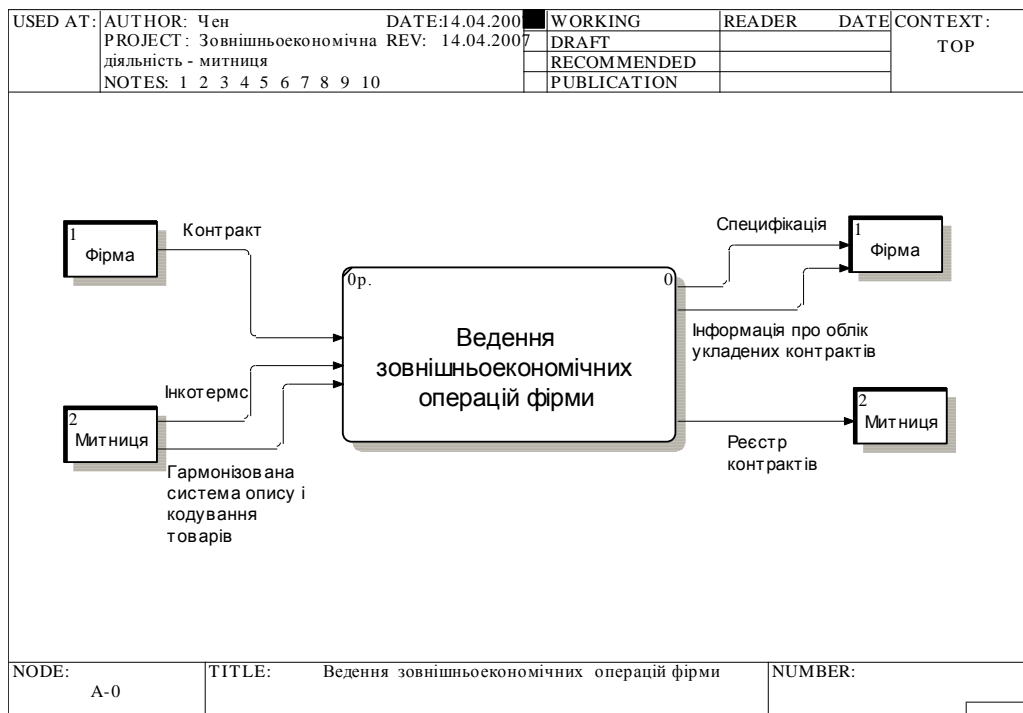


Рис. 2.15. Збереження проектної версії DFD-діаграми в текстовому документі

На рис. 2.15 показана контекстна діаграма рішення задачі "Ведення зовнішньоекономічних операцій фірми" й зовнішні об'єкти, з якими ця задача взаємодіє (ці взаємодії позначені за допомогою вхідних і вихідних інформаційних потоків).

Зовнішня сутність. Фірма моделює будь-яку фірму, яка проводить експортно-імпорتنу операцію через митницю, використовуючи при цьому раніше укладені контракти.

Специфікація – доповнення до контракту, яка складається на підставі "Контракту".


Інформація про облік укладених контрактів – служить для контролю за виконанням зобов'язань між постачальниками і покупцями.

Зовнішня сутність. Митниця моделює будь-які митниці, що контролюють законність експортно-імпорتنих операцій при проходженні митного кордону.

Інкотермс – правила й умови постачань INCOTERMS (на відвантаження продукції через митний кордон).

Гармонізована система опису і кодування товарів – дані про правила кодування товарів, які минають митний кордон.

Реєстр контрактів – перелік укладених контрактів на митниці.

Для деталізації (декомпозиції) діаграми потоків даних по основних роботах, які виконуються при вирішенні задачі "Ведення зовнішньоекономічних операцій фірми", необхідно клацнути на кнопці  – **Go to Child Diagram** на панелі інструментів (рис. 2.16) і вказати кількість робіт на ній (у нашому випадку 3), натиснути ОК. У результаті одержуємо перший етап декомпозиції, наведений на рис. 2.17.

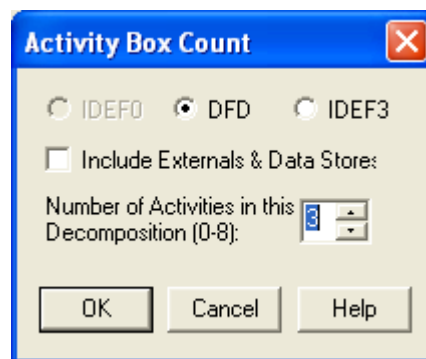


Рис. 2.16. Вибір кількості робіт для декомпозиції

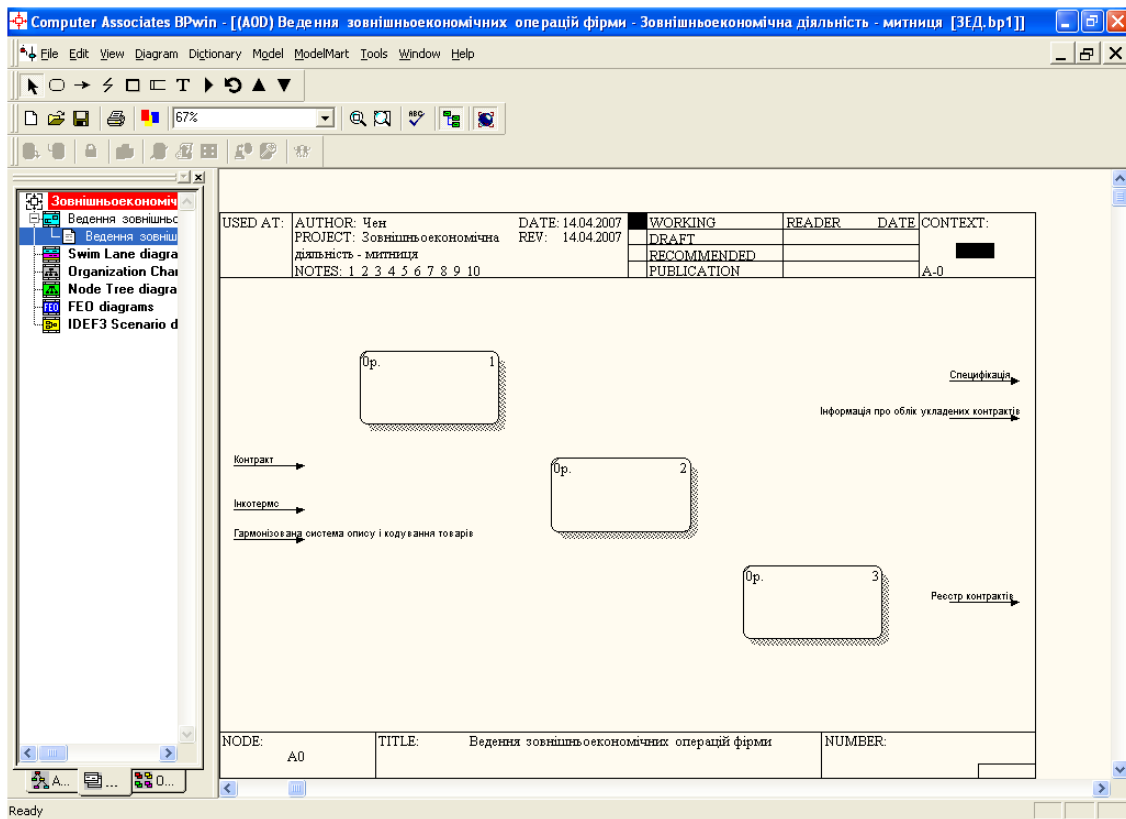


Рис. 2.17. Перший етап декомпозиції DFD-діаграми

Функціонально задача розбивається на наступні процеси: підготовка контракту, формування специфікації, ведення реєстрів договорів. Детальна послідовність виконання цієї декомпозиції наведена на рис. 2.18 – 2.20.

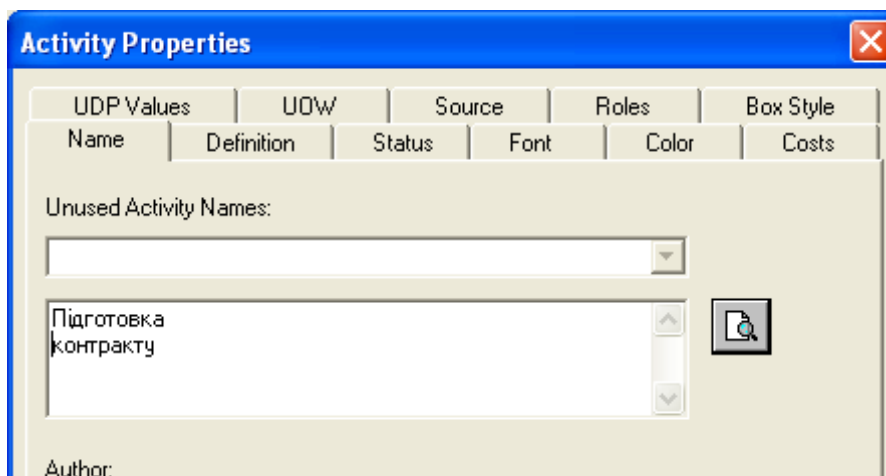


Рис. 2.18. Оформлення процесу Підготовка контракту у вікні Activity Properties

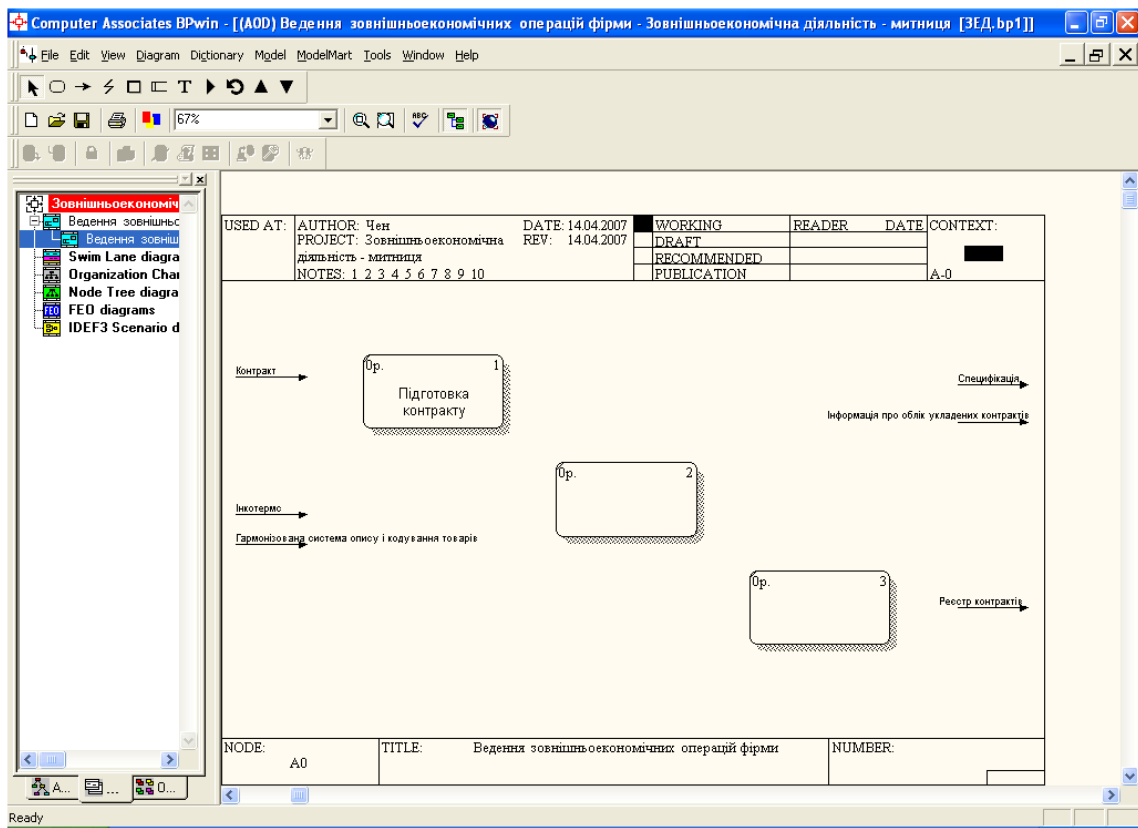


Рис. 2.19. Результат введення у поле імені процесу її найменування – Підготовка контракту

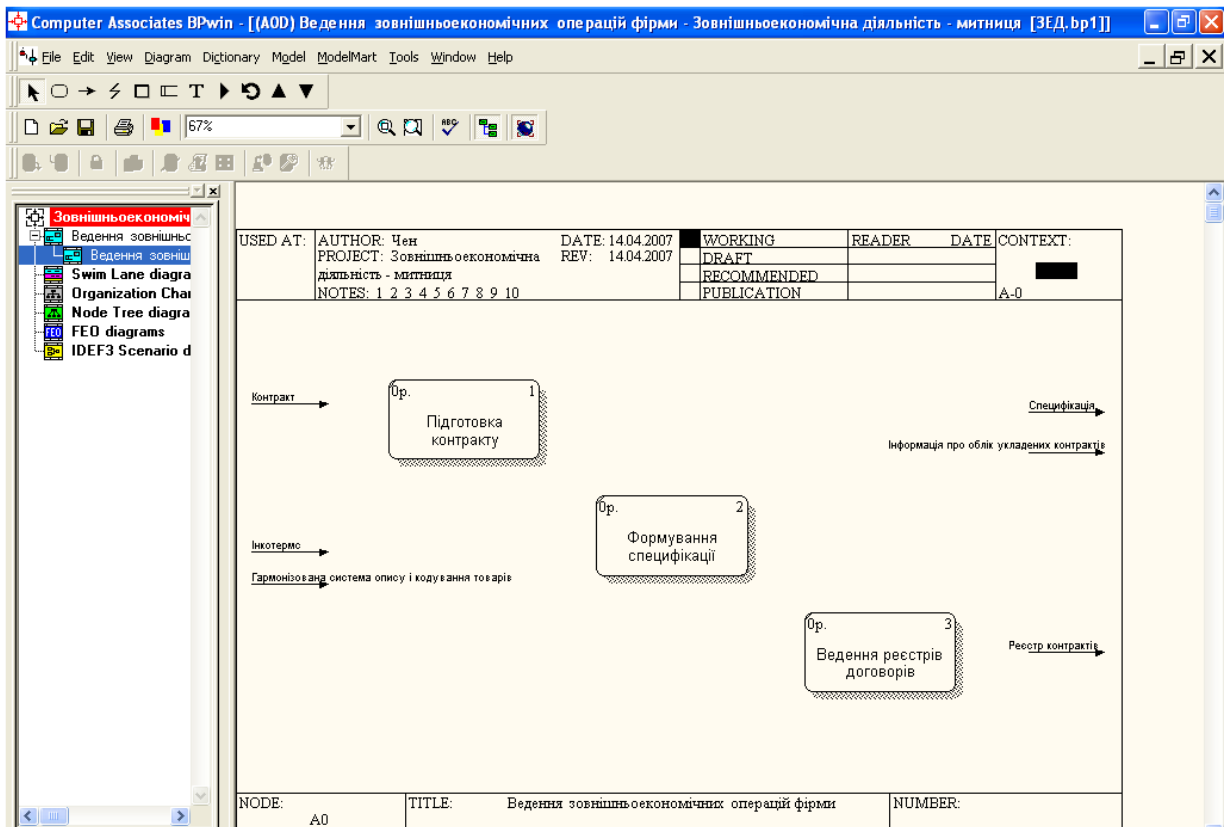


Рис. 2.20. Завдання імен усіх процесів

Далі необхідно вставити зовнішні сутності Фірма и Митниця й продовжити стрілки (рис.2.21).

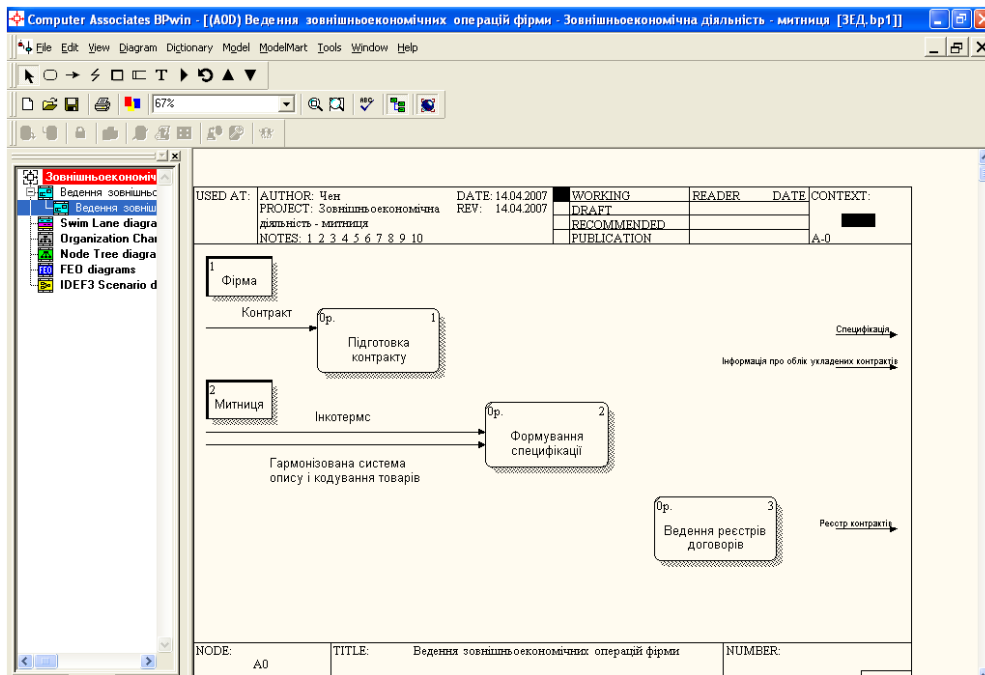


Рис. 2.21. Вставка зовнішніх сутностей Фірма и Митниця

Потім з'єднати зовнішні сутності Фірма і Митниця з відповідними роботами (рис.2.22).

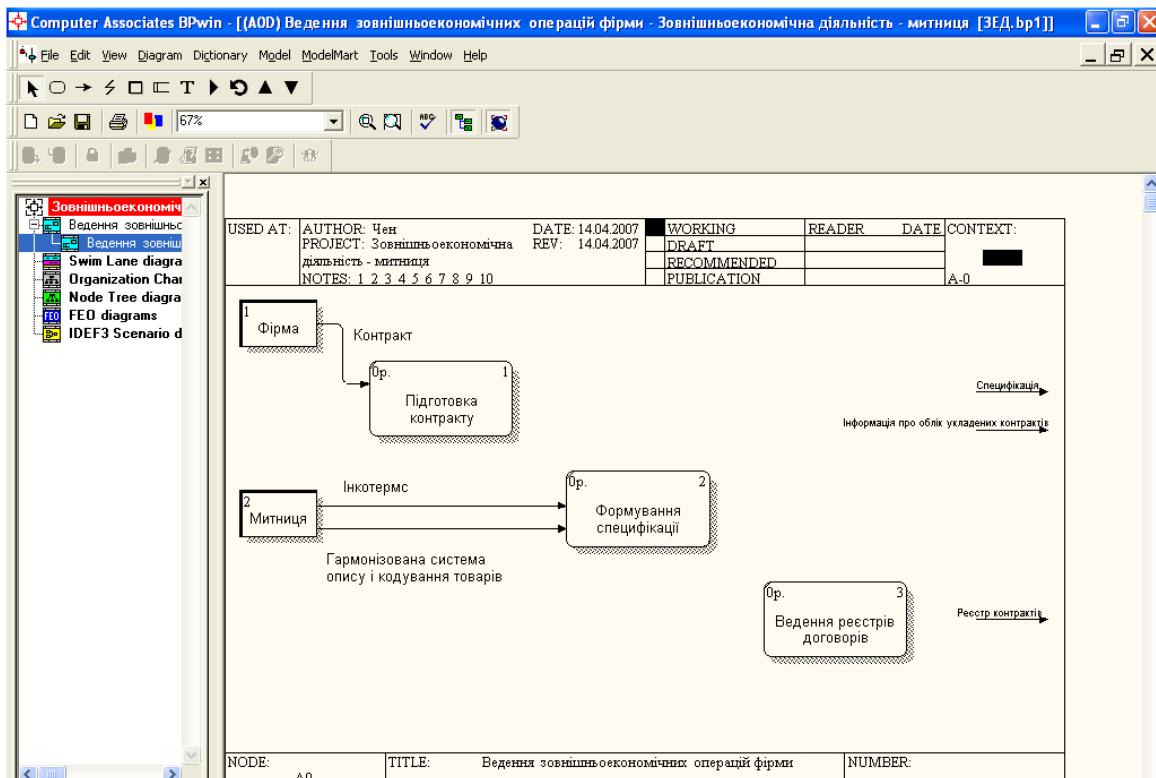


Рис. 2.22. Оформлення стрілок (потоків даних)

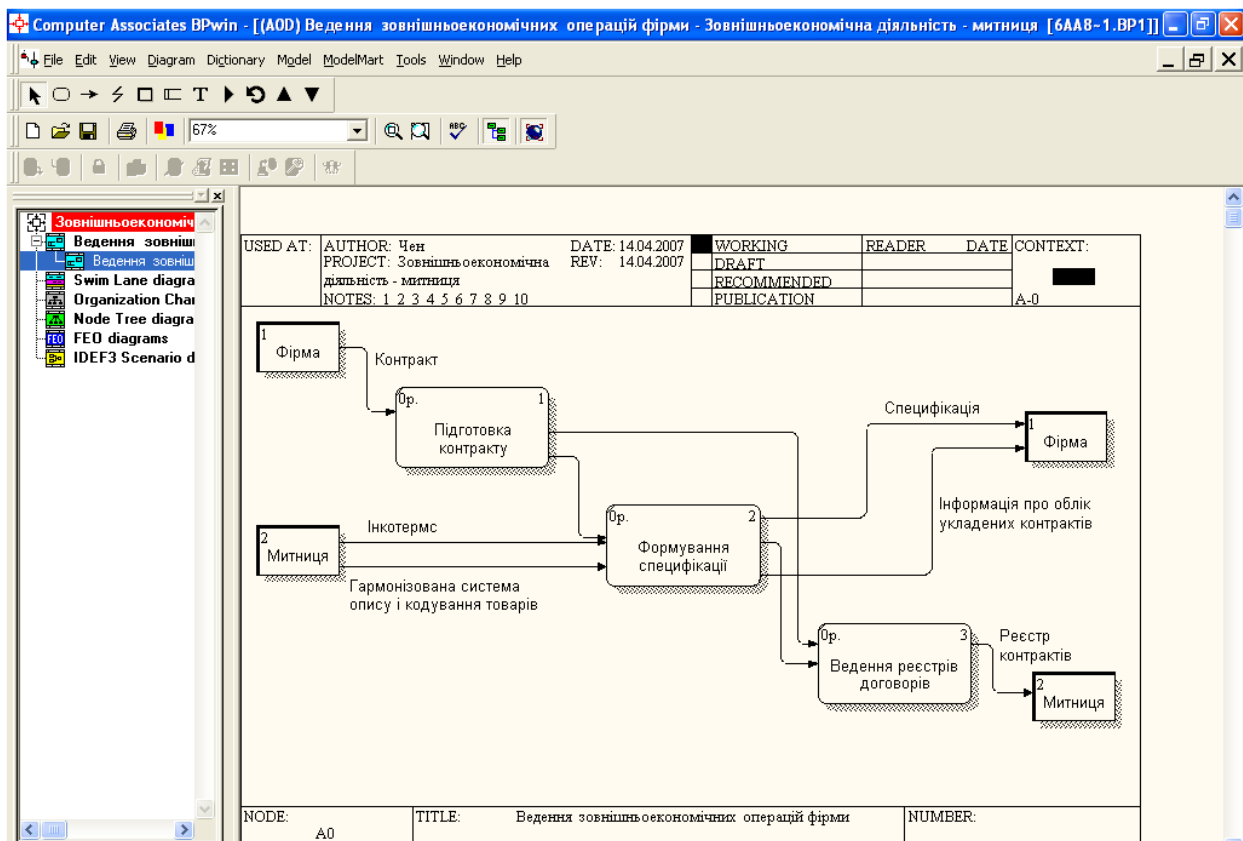


Рис. 2.23. З'єднання робіт і доробка зв'язків об'єктів з роботами

Аналогічним образом робимо з'єднання робіт і доробку зв'язків об'єктів з роботами (рис. 2.23).

Після цього необхідно ввести в діаграму накопичувачі даних, які використовуються при вирішенні задачі. Однак перед тим, як вставити сховища даних на діаграму, розробник повинен визначити загальний набір сутностей, а також основні їх атрибути, які використовуватимуться ним надалі.

Для цього необхідно зайти в пункт меню Dictionary / Entity і внести в Entity Dictionary всі сутності, які використовуватимуться для сховищ даних (рис. 2.24).

Наступний крок проектувальника – внесення для кожної з сутностей набору атрибутів. Для цього необхідно зайти в пункт Dictionary / Attribute (рис. 2.25).

Name	Definition	Exchange with ERwin
Довідник Валюта		<input checked="" type="checkbox"/> Exchange with ERwin
Довідник груп товарів		<input checked="" type="checkbox"/> Exchange with ERwin
Довідник Інкотермс		<input checked="" type="checkbox"/> Exchange with ERwin
Довідник країн		<input checked="" type="checkbox"/> Exchange with ERwin
Довідник міст		<input checked="" type="checkbox"/> Exchange with ERwin
Довідник товарів		<input checked="" type="checkbox"/> Exchange with ERwin
Довідник фірм		<input checked="" type="checkbox"/> Exchange with ERwin

Рис. 2.24. Внесення всіх сутностей у вікні Entity Dictionary

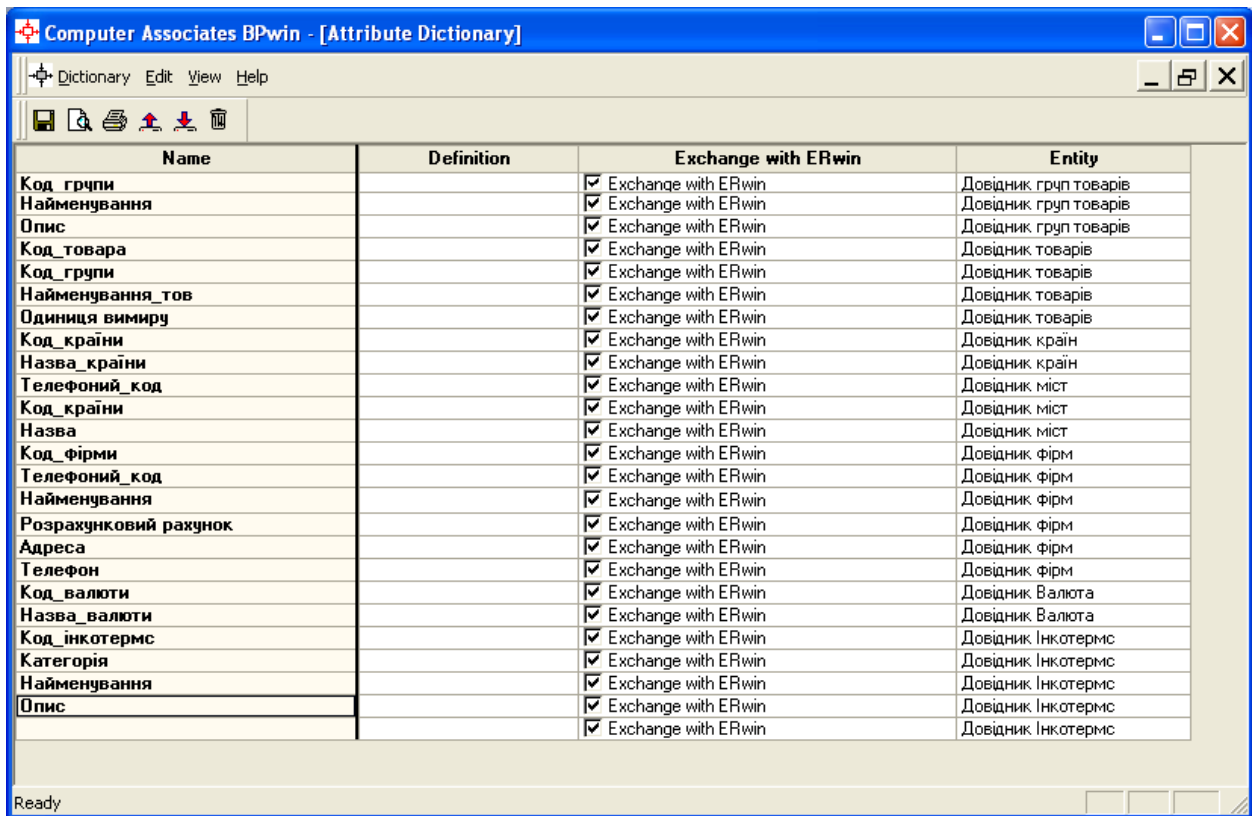



Рис. 2.25. Внесення для кожної з сутностей набору атрибутів за допомогою команди Dictionary / Attribute

Тільки після вищезазначених дій можна вносити сховища даних у діаграму. При створенні сховища даних необхідно натиснути на кнопку  на панелі інструментів, а потім – на чистому полі робочої області. З'явиться вікно введення інформації у вибір сховища даних (рис. 2.26). Необхідно вибрати радіокнопку Entity і в зміненому вікні (рис. 2.27) вибрати найменування сутності з спадаючого списку, які були внесені в словник сутностей заздалегідь. Після цього натиснути на кнопку ОК.

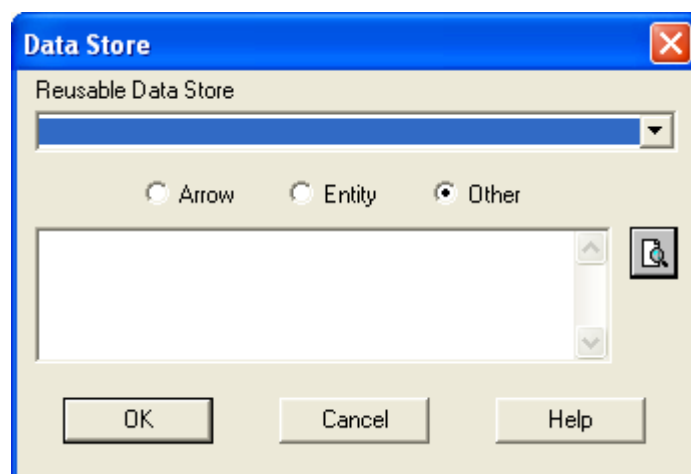


Рис. 2.26. Вікно вибору сховища даних

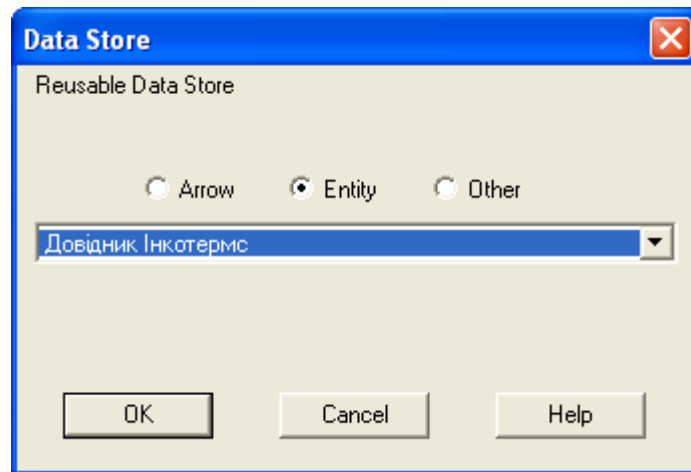


Рис. 2.27. Вікно вибору сутності зі словника сутностей

Таким чином створюються всі сховища даних (рис.2.28). До них відносяться наступні:

довідник Валюта – призначений для зберігання даних за всіма валютами використовуваних у зовнішньоекономічних операціях;

довідник фірм – призначений для зберігання даних за всіма фірмами, з якими співпрацюють постачальники і покупці;

довідник груп товарів – призначений для зберігання даних за всіма товарними позиціями і субпозиціями Гармонізованої системи опису і кодування товарів;

довідник товарів – призначений для зберігання даних за номенклатурою митних тарифів, розроблених відповідно до законодавства договірних сторін для стягування мита при імпорті товарів;

довідник Інкотермс – призначений для зберігання даних за всіма правилами і умовами постачань товарів за митний кордон країни;

довідник міст – призначений для зберігання даних у всіх містах, у відповідності до адміністративно-територіального ділення України;

довідник країн – призначений для зберігання даних у всіх країнах світу.

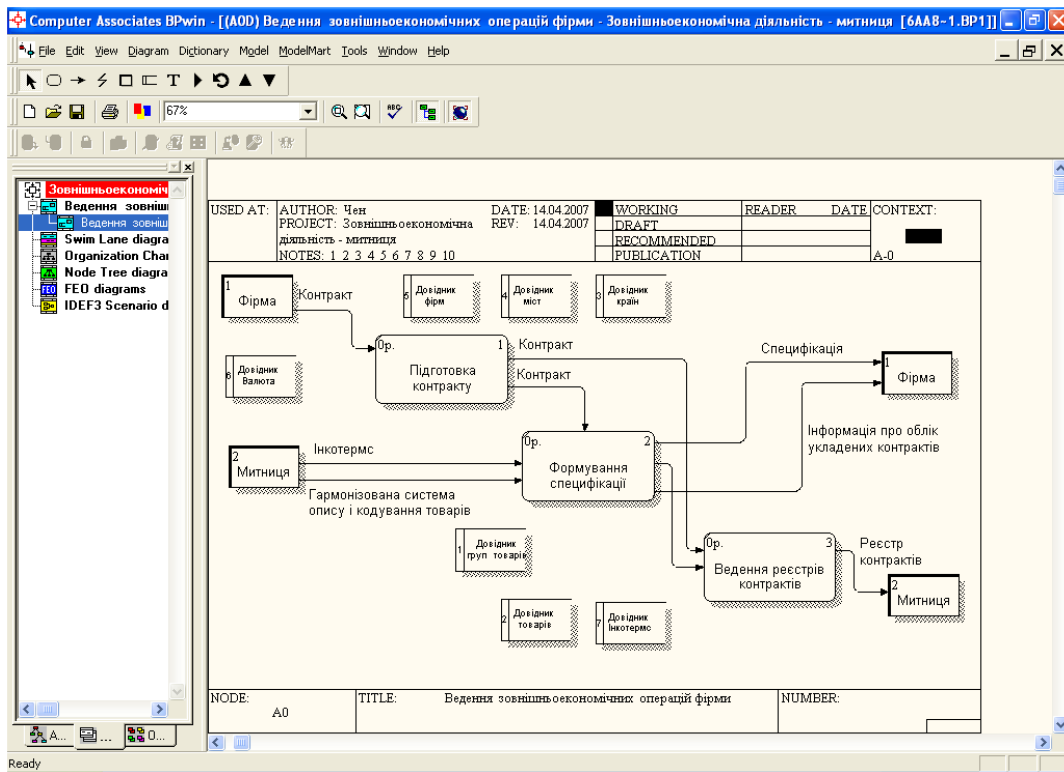


Рис. 2.28. Створення всіх сховищ даних

Потім провести стрілки до тих робіт, у яких вони використовуються (рис. 2.29)

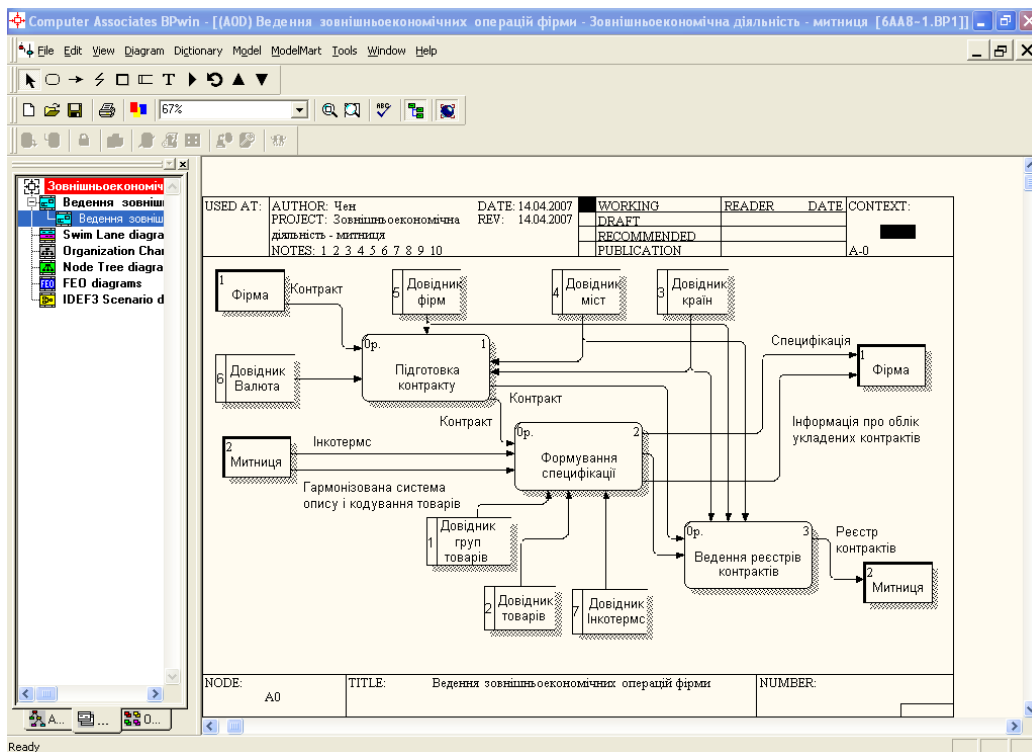


Рис. 2.29. Остаточний від діаграми DFD для рішення задачі "Ведення зовнішньоекономічних операцій фірми"

3. Експорт даних з bpwin у erwin

Мета навчитися експортувати дані з BPwin у ERwin і будувати логічну та фізичну моделі даних на їх основі.

Хід виконання роботи

Для експорту одержаних сутностей і атрибутів в ERwin необхідно після створення всіх сутностей з атрибутами використати пункт меню **File / Export / ERwin 4.0 (BPX)** (рис. 3.1). При цьому на екрані з'явиться вікно, в котрому треба ввести ім'я файлу, в якому будуть збережені дані, і вказати, де буде знаходитися файл даних (рис. 3.2).

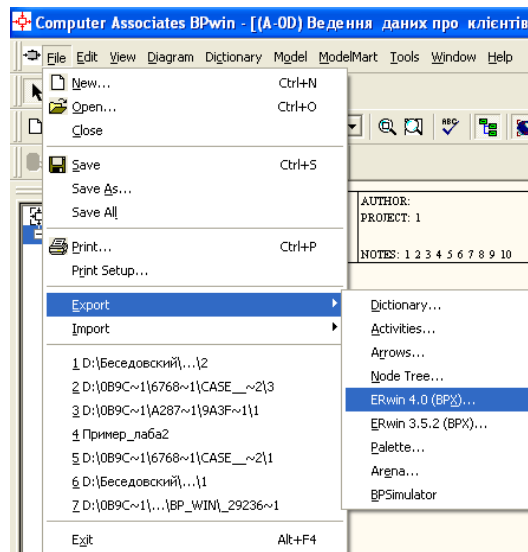


Рис. 3.1. Експорт сутностей і атрибутів з BPwin в ERwin

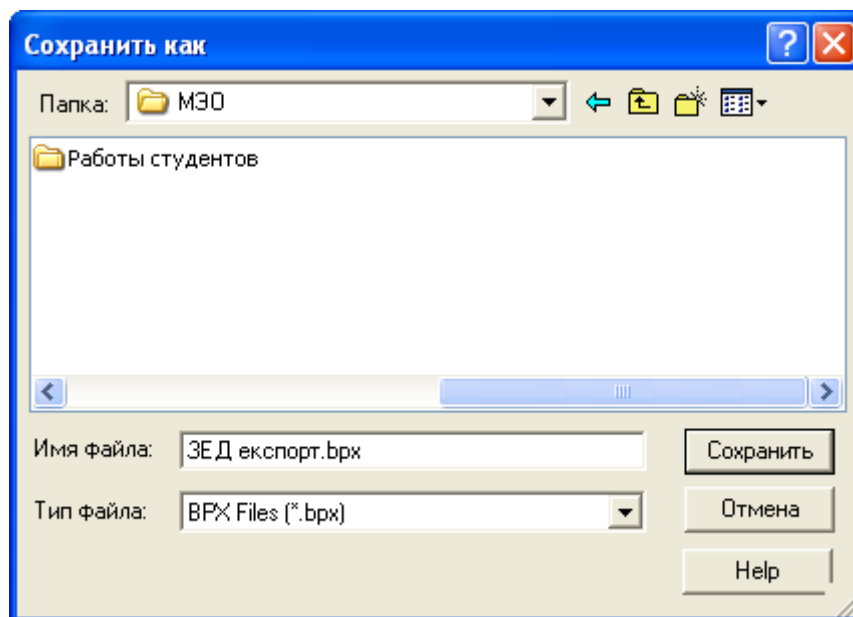


Рис. 3.2. Перший етап експорту даних

Після цього необхідно завантажити ERwin, створити нову модель (логічну та фізичну) (рис. 3.3). Для імпорту даних необхідно використати команду **File / Import / BPwin** (рис. 3.4) та вибрати файл даних, з котрого буде проведений імпорт (рис. 3.5).

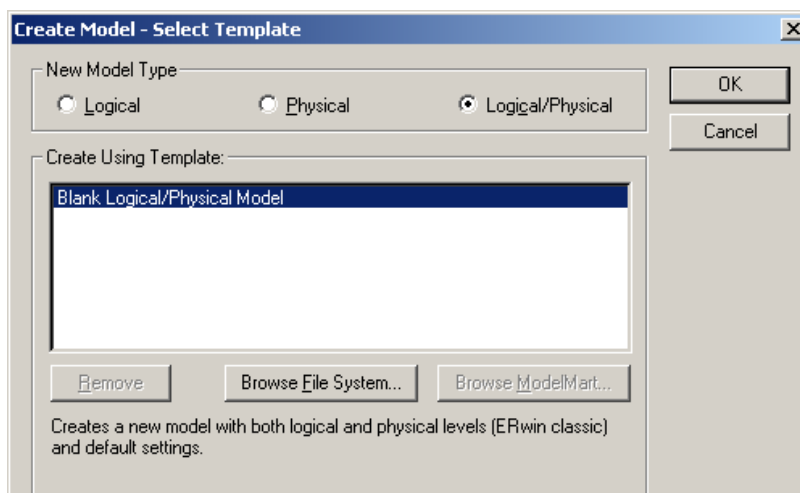


Рис. 3.3. Вікно створення нової моделі

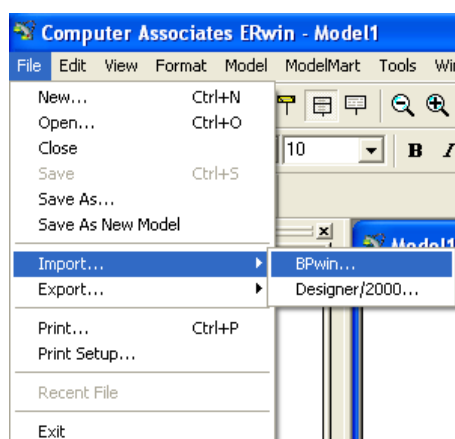


Рис. 3.4. Вибір пункту меню імпорту даних з BPwin

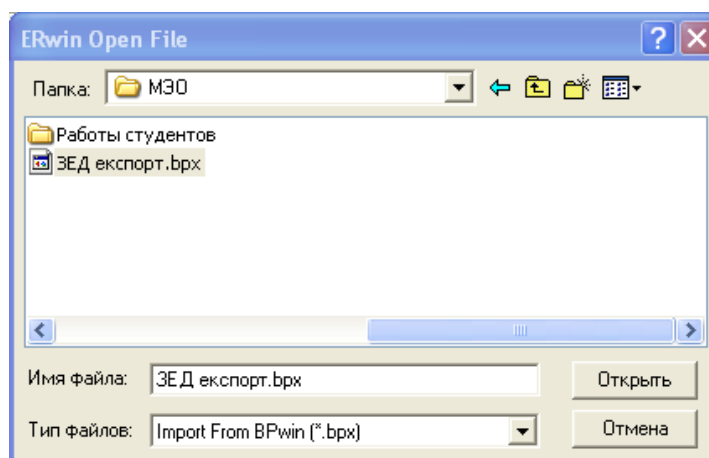


Рис. 3.5. Вибір файлу для імпорту даних

У вікні, що відкрилося (рис. 3.6), необхідно вказати сутності та атрибути, котрі повинні бути імпортовані в модель даних (за умовчужанням усі сутності з атрибутами імпортуються). Не рекомендується здійснювати експорт і імпорт декілька разів, тому що при наступному імпорті до раніше імпортованих даних буде заблокований доступ.

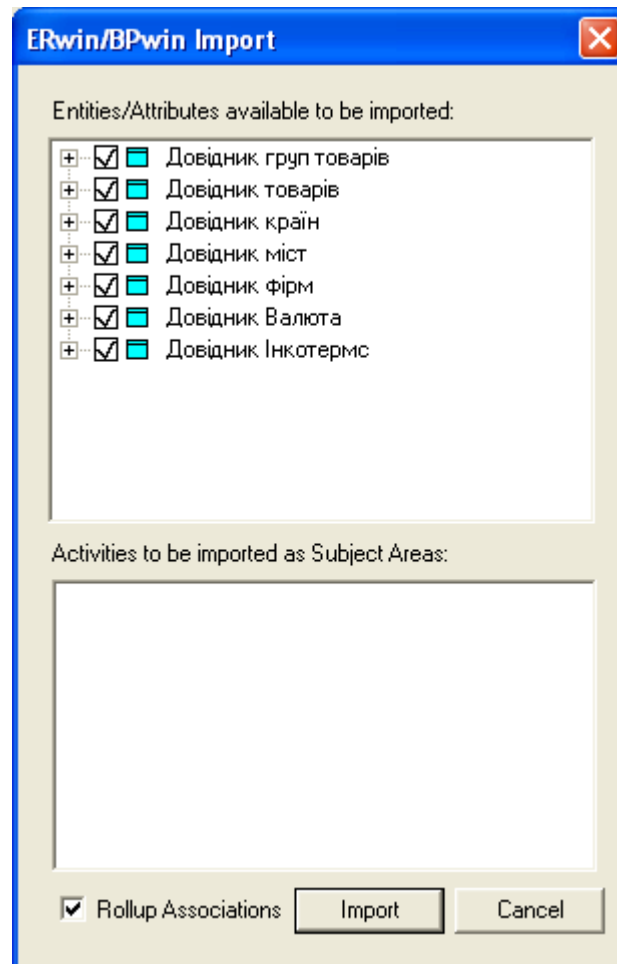


Рис. 3.6. Вікно вибору сутностей та атрибутів для імпорту

Результатом імпорту буде вікно з сутностями, зображеними на рис. 3.7. Найчастіше імпортовані дані спочатку будуть мати незрозумілий вигляд. Для нормального їх відображення необхідно використати команду **Format / Default Fonts & Colors** (рис. 3.8) й змінити для сутностей та атрибутів значення поле **Font** на **Arial Cyr** або **Times New Roman Cyr**, у пункті **Apply To** вибрати радіо-кнопку **All Objects** (рис. 3.9).

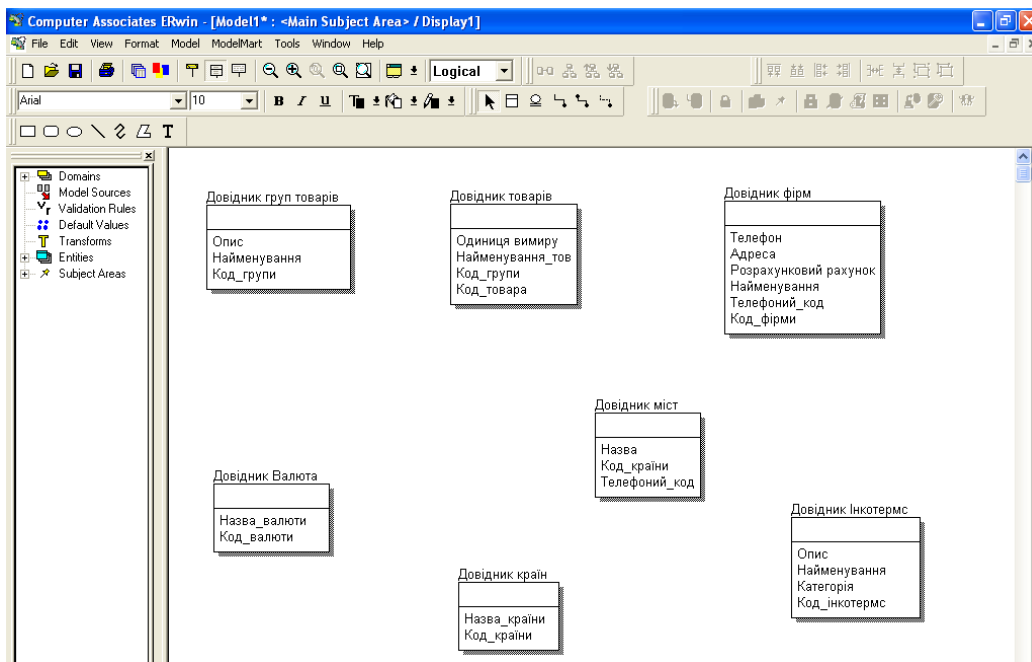


Рис. 3.7. Вікно з імпортованими даними

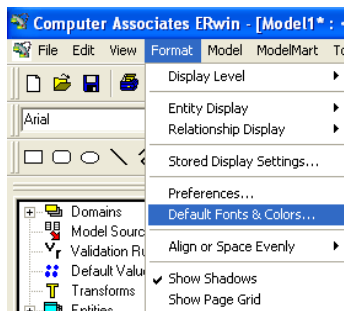


Рис. 3.8. Меню вибору шрифту

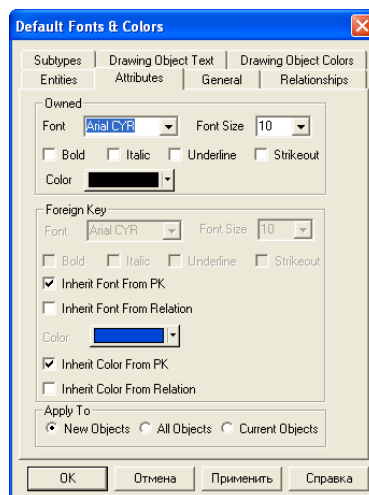


Рис. 3.9. Вікно зміни шрифту для сутностей та атрибутів

Подальша робота з побудови логічної та фізичної моделей даних здійснюється відповідно етапів побудови бази даних.

4. Створення моделі даних за допомогою allfusion erwin data modeler (erwin)

4.1. Основи методології IDEF1X

Метод IDEF1, розроблений Т. Рэмей (Т. Ramey) и заснований на підході П. Чена, дозволяє побудувати модель даних, еквівалентну реляційній моделі в третій нормальній формі. На основі вдосконалювання методології IDEF1 створена її нова версія – методологія IDEF1X. IDEF1X розроблена з урахуванням таких вимог, як простота вивчення і можливість автоматизації. IDEF1X-діаграми використовують розповсюджені CASE-засоби (зокрема ERwin, Design/IDEF).

Сутність у методології IDEF1X є незалежність від ідентифікаторів або просто незалежність, якщо кожен екземпляр сутності може бути однозначно ідентифікований без визначення його відносин з іншими сутностями. Сутність називається залежною від ідентифікаторів або просто залежною, якщо однозначна ідентифікація екземпляра сутності залежить від його відношення до іншої сутності (рис. 4.1).

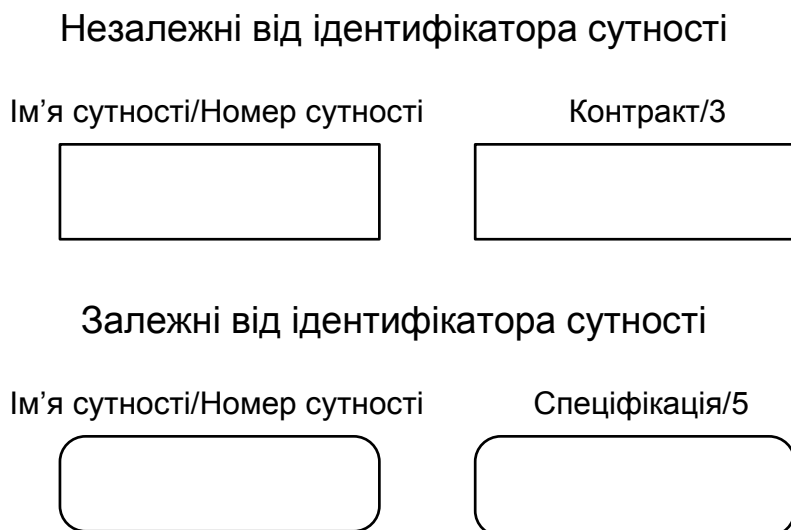


Рис. 4.1. Сутності

Кожній сутності привласнюється унікальне ім'я і номер, розділені косою рисою "/" і поміщаються над блоком.

Зв'язок може додатково визначатися за допомогою вказівки ступеня або потужності (кількості екземплярів сутності-нащадка, що може існувати для екземпляра сутності-батька). У IDEF1X можуть бути виражені наступні потужності зв'язків:

кожен екземпляр сутності-батька може мати нуль, один або більш зв'язаних з ним екземплярів сутності-нащадка;

кожен екземпляр сутності-батька повинен мати не менш одного зв'язаного з ним екземпляра сутності-нащадка;

кожен екземпляр сутності-батька повинен мати не більш одного зв'язаного з ним екземпляра сутності-нащадка;

кожен екземпляр сутності-батька зв'язаний з деяким фіксованим числом екземплярів сутності-нащадка.

Якщо екземпляр сутності-нащадка однозначно визначається своїм зв'язком із сутністю-батьком, то зв'язок називається ідентифікуючим, у протилежному випадку – неідентифікуючим.

Зв'язок зображується лінією, проведеною між сутністю-батьком і сутністю-нащадком із крапкою на кінці лінії в сутності-нащадка.

Потужність зв'язку становить відношення кількості екземплярів батьківської сутності до відповідної кількості екземплярів дочірньої сутності. Для будь-якого зв'язку, крім неспецифічного, цей зв'язок записується як 1:n.

ERwin, відповідно до методології IDEF1X, надає 4 варіанти для n, які зображуються додатковим символом у дочірній сутності:

- – нуль, один або більше;
- P – один або більше;
- Z – нуль або один;
- N – рівно N, де N – конкретне число.

Допустимість порожніх (NULL) значень у неідентифікуючих зв'язках зображується порожнім ромбиком на дузі з боку батьківської сутності.

Ідентифікуючий зв'язок між сутністю-батьком і сутністю-нащадком зображується суцільною лінією (рис. 4.2). Сутність-нащадок в ідентифікуючому зв'язку є залежною від ідентифікатора сутністю. Сутність-батько в ідентифікуючому зв'язку може бути як незалежною, так і залежною від ідентифікатора сутністю (це визначається її зв'язками з іншими сутностями).

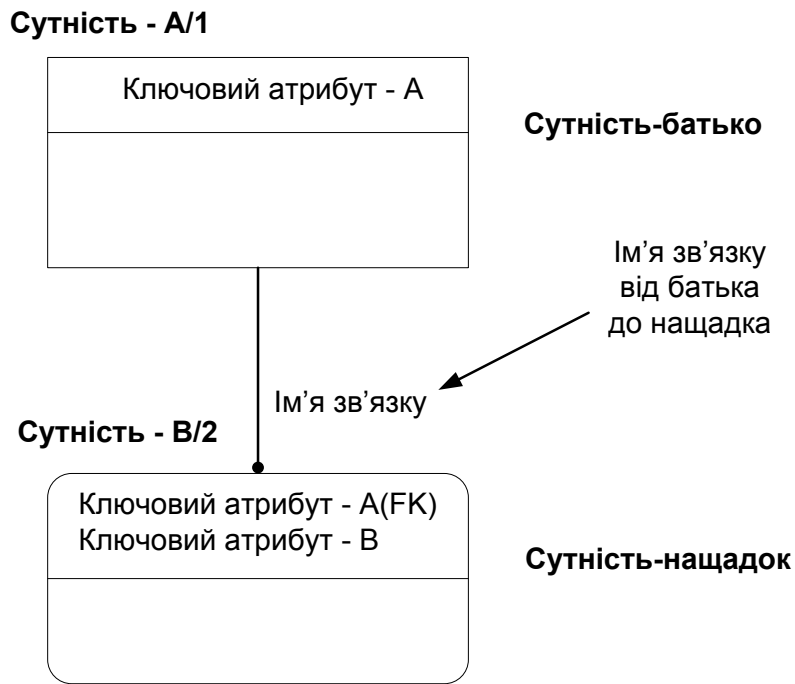


Рис. 4.2. Ідентифікуючий зв'язок

Пунктирна лінія зображує неідентифікуючий зв'язок (рис. 4.3). Сутність-нащадок у неідентифікуючому зв'язку буде незалежною від ідентифікатора, якщо вона не є також сутністю-нащадком у якому-небудь ідентифікуючому зв'язку.

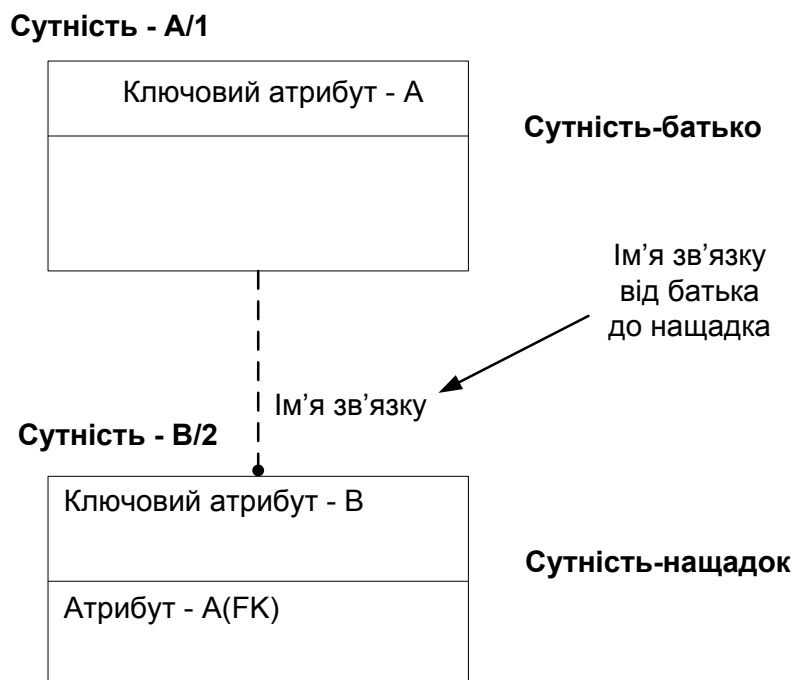


Рис. 4.3. Неідентифікуючий зв'язок

Атрибути зображуються у вигляді списків імен усередині блоків сутності. Атрибути, що визначають первинний ключ, розміщуються нагорі списків і відокремлюються від інших атрибутів горизонтальною рисою (рис. 4.4).

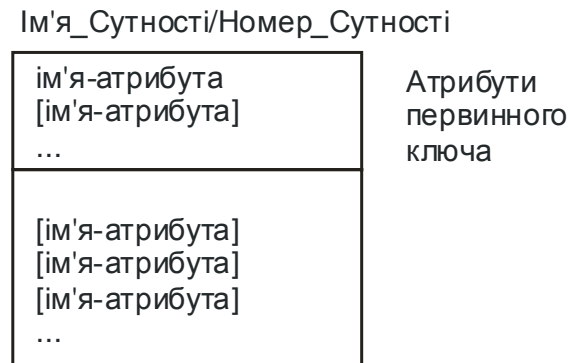
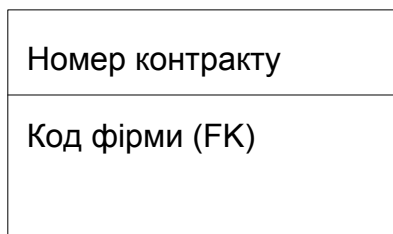


Рис. 4.4. Атрибути і первинні ключі

Сутності можуть мати також зовнішні ключі (Foreign Key, FK), що можуть використовуватися як частина або цілий первинний ключ або неключовий атрибут. Зовнішній ключ зображується за допомогою поміщення усередині блоку сутності імен атрибутів, після яких впливають букви FK у дужках (рис.4.5).

Приклад зовнішнього ключа – неключового атрибуту

Контракт/3



Приклад зовнішнього ключа – атрибута первинного ключа

Товар/7

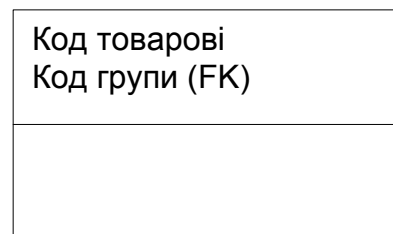


Рис. 4.5. Приклади зовнішніх ключів

4.2. Процес побудови інформаційної моделі

Побудова моделі БД інформаційної системи починається з аналізу документів, які описують її функціональні можливості. Після вивчення документів необхідно сформулювати мету моделювання та точку зору на модель. Мета і точка зору моделі визначаються на самій ранній стадії створення моделі. Вибір мети здійснюється з урахуванням питань, на які повинна відповісти модель, а вибір точки зору – відповідно до вибору позиції, за якою описується система. Іноді мету і точку зору можна вибрати до того, як буде зроблена перша діаграма.

Процес побудови інформаційної моделі складається з наступних кроків:

- визначення сутностей;
- визначення залежностей між сутностями;
- завдання первинних і альтернативних ключів;
- визначення атрибутів сутностей;
- приведення моделі до необхідного рівня нормальної форми;
- перехід до фізичного опису моделі: призначення відповідностей ім'я сутності – ім'я таблиці, атрибут сутності – атрибут таблиці; завдання тригерів, процедур і обмежень;
- генерація бази даних.

ERwin створює візуальне представлення (модель даних) для розв'язуваної задачі. Це представлення може використовуватися для детального аналізу, уточнення і поширення як частини документації, необхідної в циклі розробки. Однак ERwin далеко не тільки інструмент для малювання. ERwin автоматично створює базу даних (таблиці, індекси, збережені процедури, тригери для забезпечення посилальної цілісності й інші об'єкти, необхідні для керування даними).

Діаграма ERwin будується з трьох основних блоків – сутностей, атрибутів і зв'язків. Якщо розглядати діаграму як графічне представлення правил предметної області, то сутності й атрибути є іменниками, а зв'язки – дієсловами.

Звичайно моделі ERwin зберігаються на диску у вигляді файлу. Мається можливість зберегти модель у цільовий СУБД. Для цього за допомогою самого ERwin у цільовий СУБД створюється метабаза ERwin. У цій базі даних зберігається інформація моделі. В окремому випадку базою даних можуть бути і dBase-файли, з якими ERwin працює через ODBC.

4.3. Опис роботи з пакетом

При запуску ERwin за замовчуванням з'являється основна панель інструментів, палітра інструментів (рис.4.6). При створенні нової моделі обов'язково необхідно вказати тип моделі **Logical/Physical** (див. рис. 3.3), а у властивостях моделі відзначити птичкою автоматичний розрив зв'язку багато-до-багатьох асоціативною таблицею, використовуючи команду **Model → Model Properties → Many to Many Relationships with Association Table** (рис.4.7).

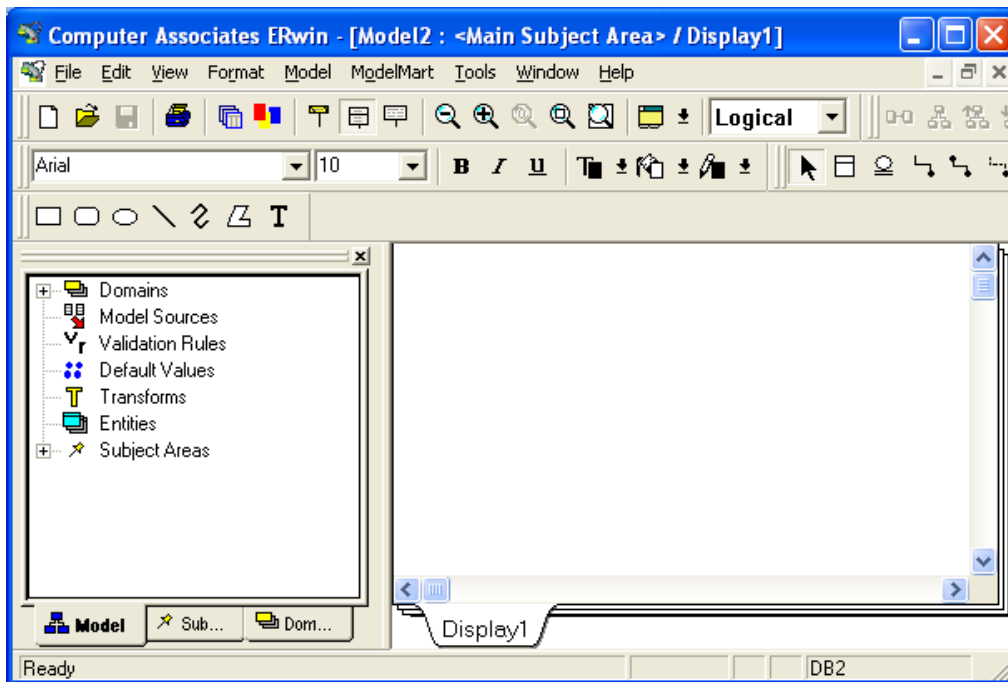


Рис.4.6. Інтегрована середовище розробки ERwin

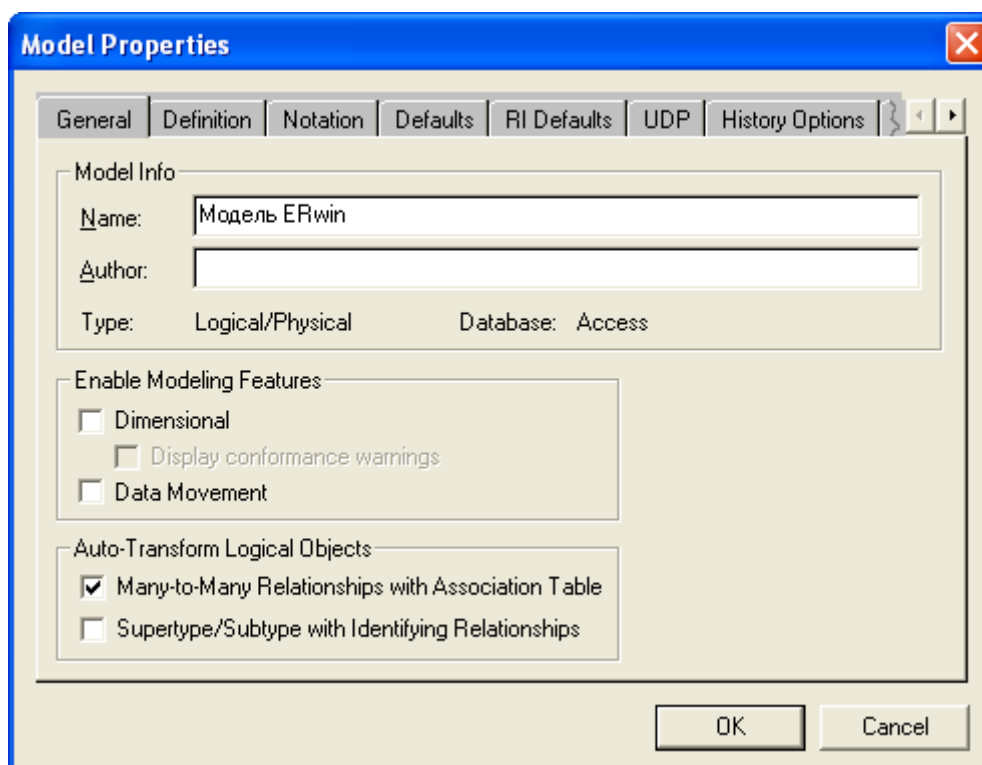








Рис. 4.7. Вибір Auto-Transform Logical Objects

Основні функції ERwin щодо відображення моделі, а також панель і палітра інструментів наведені в табл. 4.1.

Призначення кнопок палітри інструментів

Кнопки	Призначення кнопок
	Створення, відкриття, збереження і друк моделі
	Виклик діалогу Report Browser для генерації звітів
	Зміна рівня перегляду моделі: рівень сутностей, рівень атрибутів, рівень визначень
	Зміна масштабу перегляду моделі
	Генерація схеми БД, вирівнювання схеми з моделлю і вибір сервера (доступні тільки на рівні фізичної моделі)
	Переключення між областями моделі Subject Area

Палітра інструментів виглядає по-різному на різних рівнях відображення моделі. На логічному рівні палітра інструментів (рис. 4.8) має:

ліворуч на право, верхній ряд –

кнопку покажчика (режим миші)

кнопку внесення сутності

кнопку категорії

Кнопки створення зв'язків: ідентифікуючу, багато-до-багатьох, неідентифікуючу.

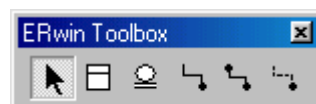



Рис. 4.8. Палітра інструментів

4.4. Створення логічної й фізичної моделі БД

Для внесення сутності в модель необхідно "клацнути" по кнопці сутності на панелі інструментів (ERwin Toolbox) , потім – по тому місцю на діаграмі, де необхідно розташувати нову сутність (рис. 4.9, 4.10). Клацнувши правою кнопкою миші по сутності і вибравши зі спливаючого меню пункт **Entity Properties**, можна викликати діалог **Entities**, у якому визначаються ім'я, коментарі й опис сутності.

Кожна сутність повинна бути цілком визначена за допомогою текстового опису в закладці **Definition**. Закладки **Note**, **Note 2**, **Note 3**, **UDP** служать для внесення додаткових коментарів і визначень до сутності.

Наступним кроком у процесі створення логічної моделі повинне стати визначення зв'язків між сутностями.

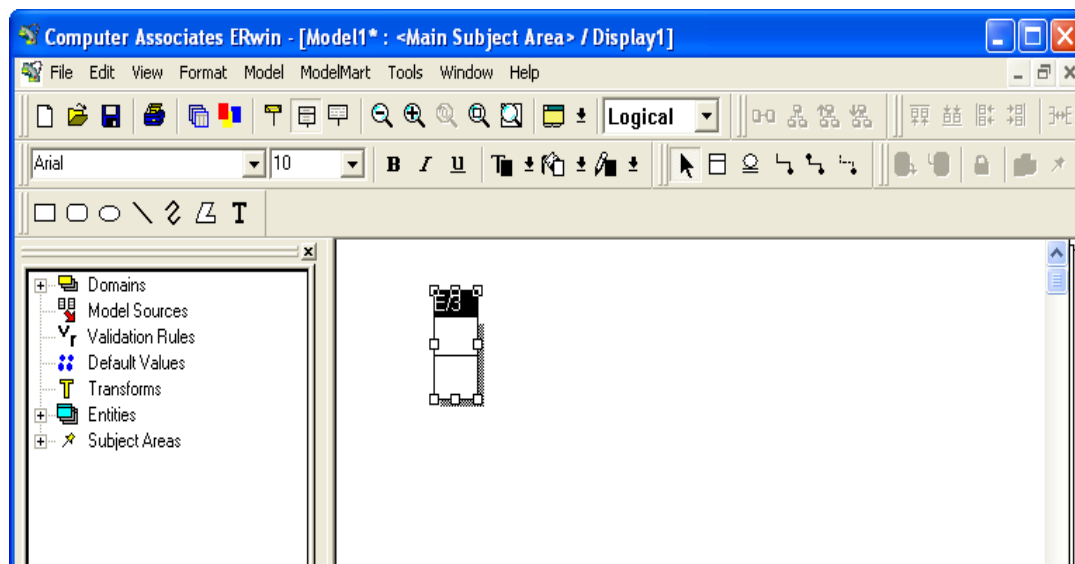


Рис. 4.9. Розташування нової сутності

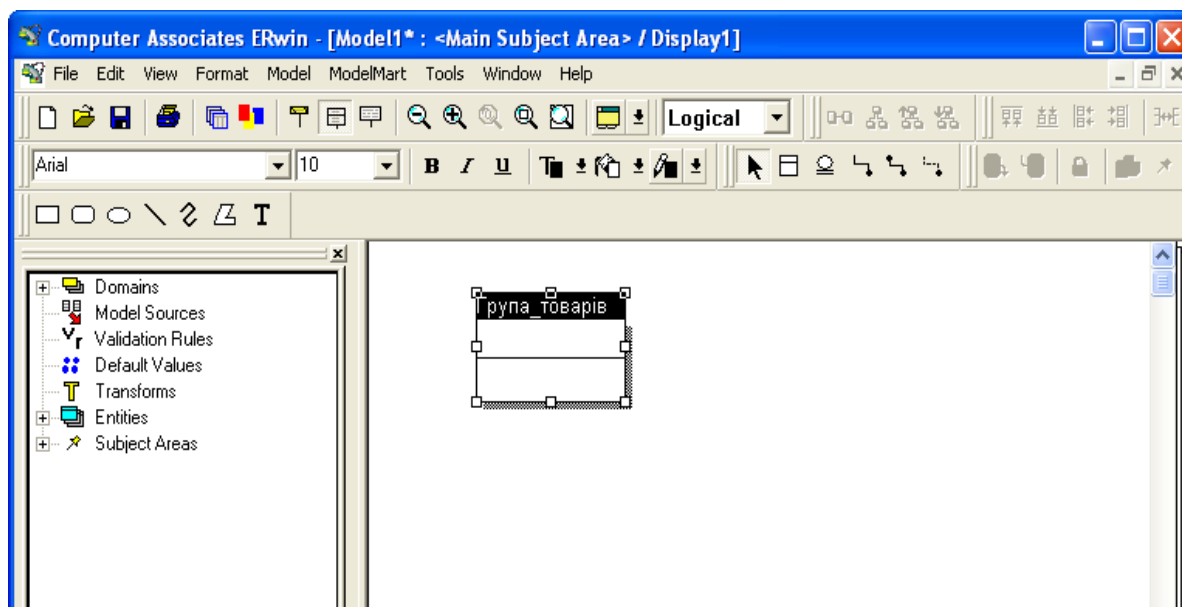


Рис. 4.10. Визначення імені сутності

Тепер для даної сутності необхідно вказати первинні ключі і неключові атрибути. Крім того, для деяких, можливо, знадобиться завдання альтернативних ключів і інверсних входів.

Для завдання первинних ключів і атрибутів використовується редактор атрибутів. Для його виклику клікніть правою кнопкою миші на сутності і виберіть пункт **Attribute Editor** (рис. 4.11).

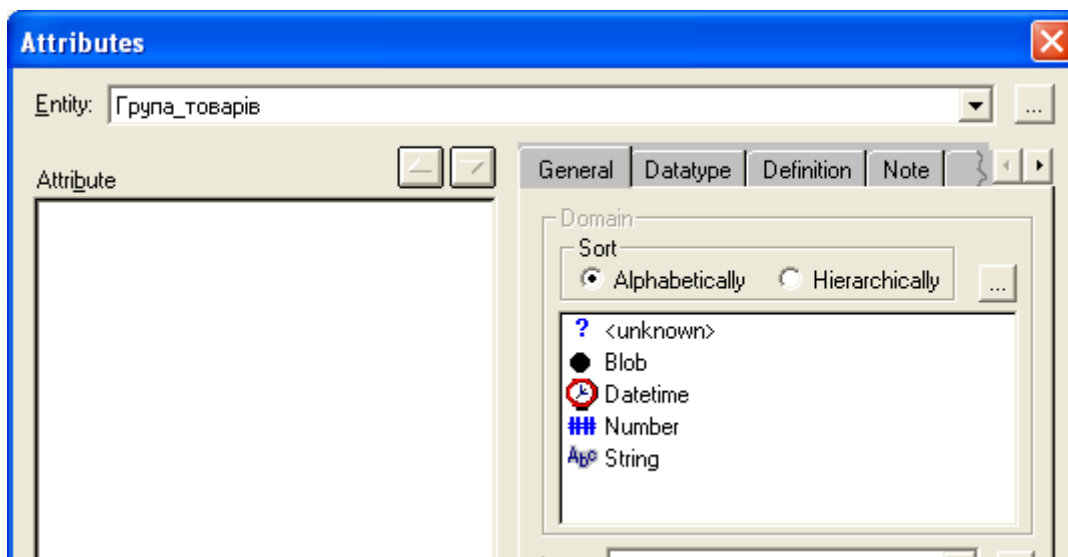


Рис. 4.11. Вікно редактору атрибутів для завдання первинних ключів і атрибутів сутності

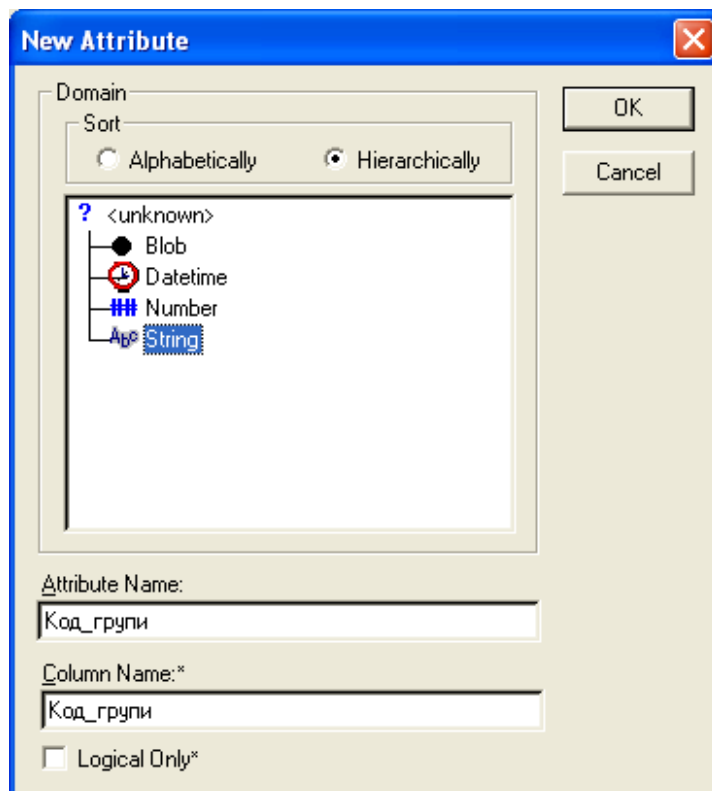


Рис. 4.12. Вибір типу даних для атрибутів

Для кожної сутності задаємо перелік атрибутів, серед яких обов'язково повинен бути ключовий (рис. 4.12 – 4.14)

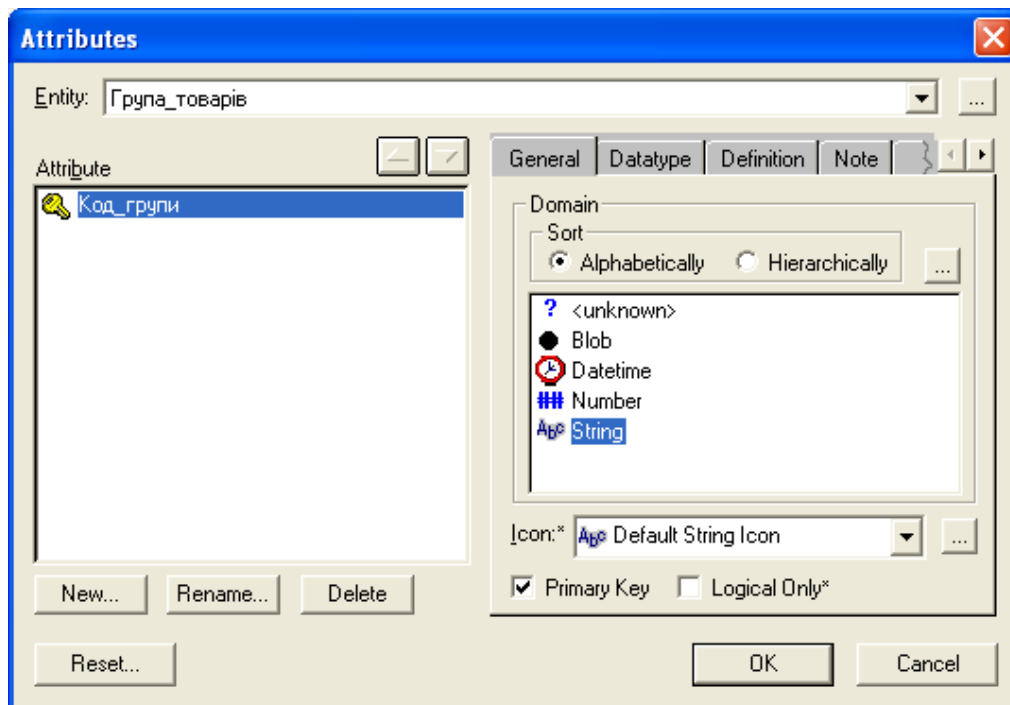


Рис. 4.13. Завдання первинного ключа

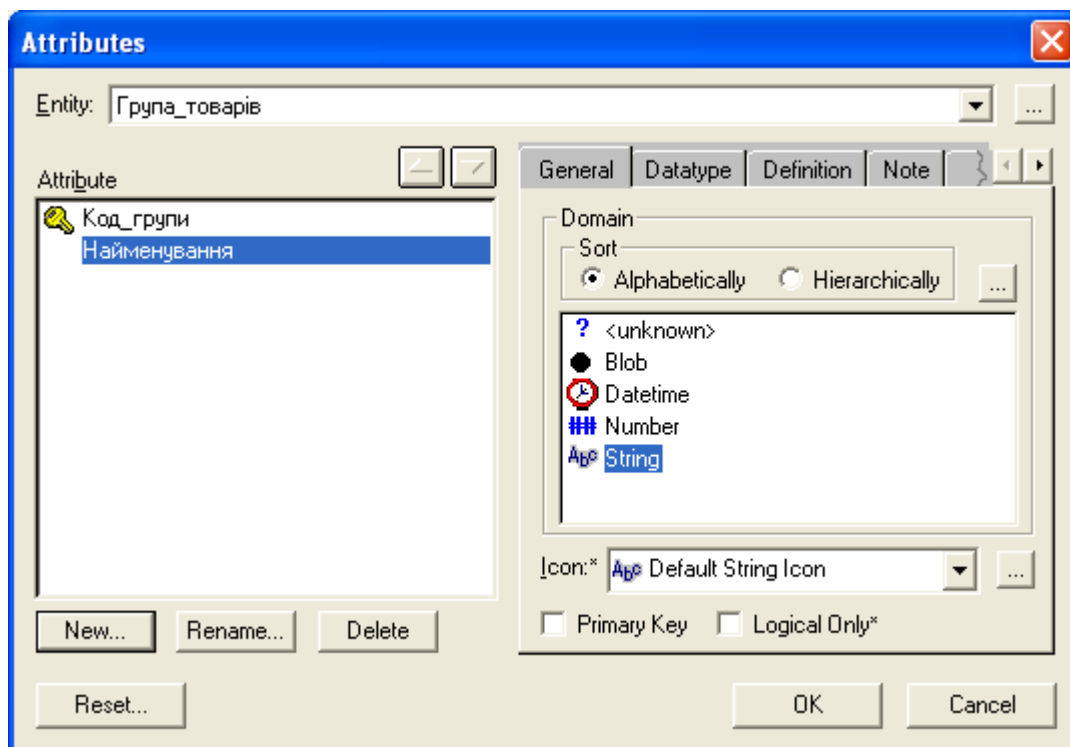


Рис. 4.14. Додавання нових атрибутів для сутності

Аналогічним образом створюємо сутності для заданої предметної області й одержуємо наступний стан вікна логічної моделі у ERwin (рис. 4.15).

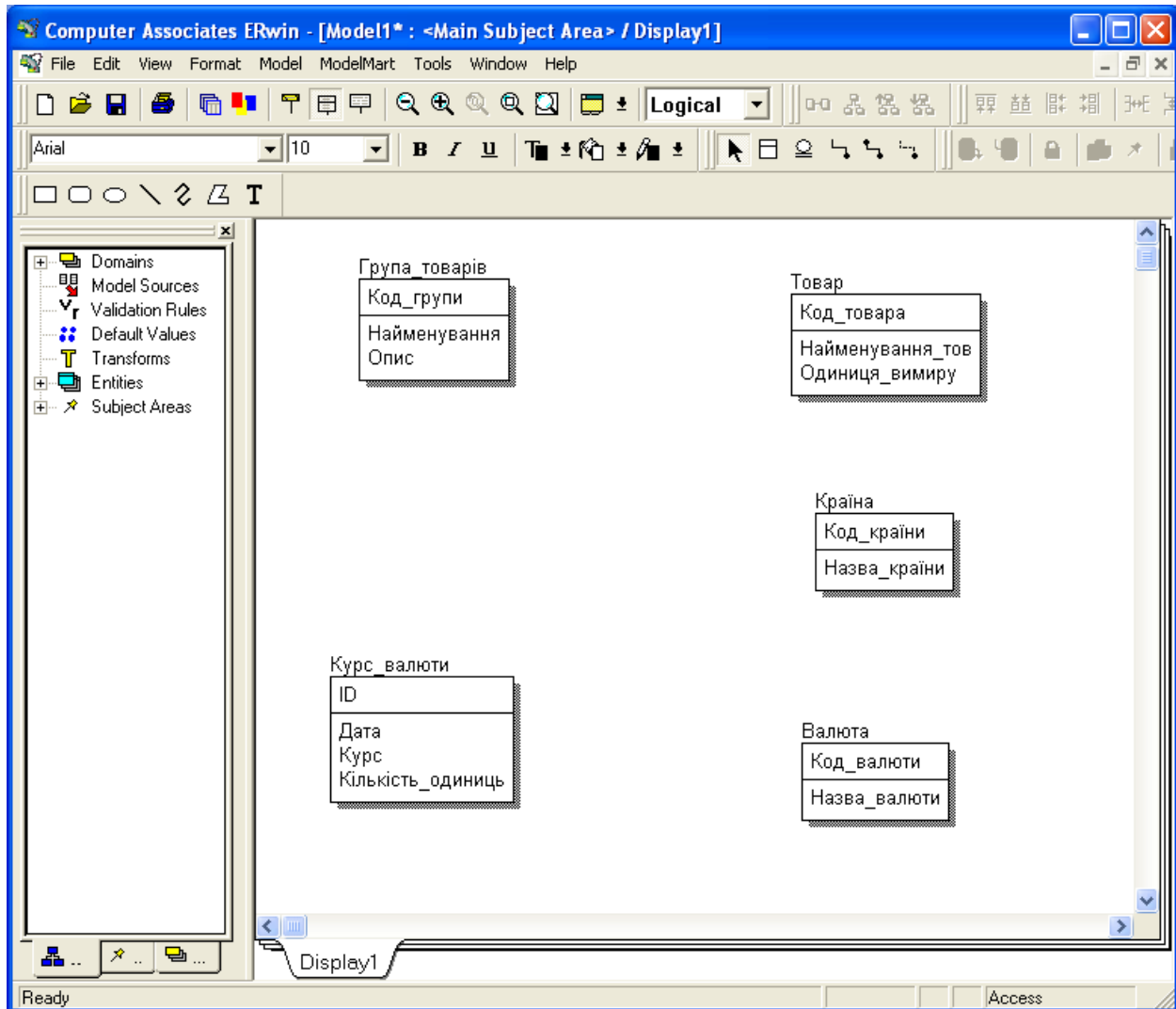


Рис. 4.15. Стан вікна логічної моделі у Erwin після завдання всіх сутностей

Створення зв'язків між сутностями вимагає знання опису й семантики (тобто смислу) даної предметної області.

Для встановлення зв'язку необхідно клацнути по кнопці зв'язок у панелі інструментів, потім указати мишею спочатку на батьківську сутність, а потім на дочірню. Задати параметри зв'язку можна за допомогою редактора зв'язків **Relationships** (рис. 4.16).

Після завдання зв'язків між сутностями діаграма буде виглядати в такий спосіб (рис. 4.17).

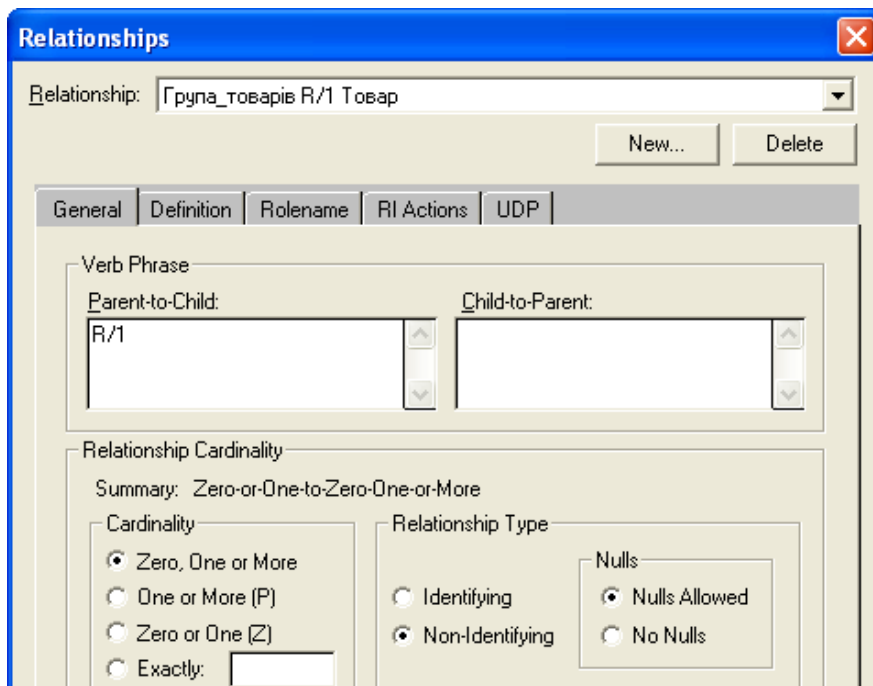


Рис. 4.16. Діалогове вікно Relationship для завдання параметрів зв'язку

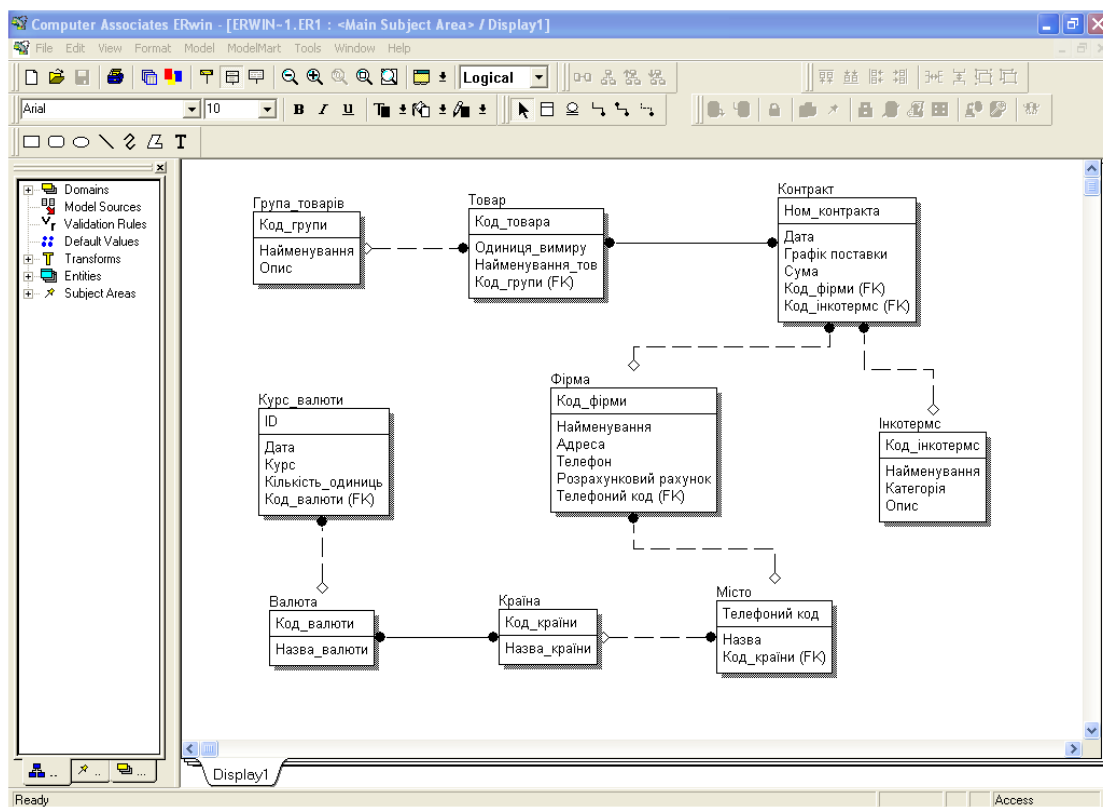


Рис. 4.17. Вид логічної моделі після завдання зв'язків

Після переходу на фізичний рівень (просто переключивши **Logical** на **Physical**) модель придбає наступний вид (рис. 4.18).

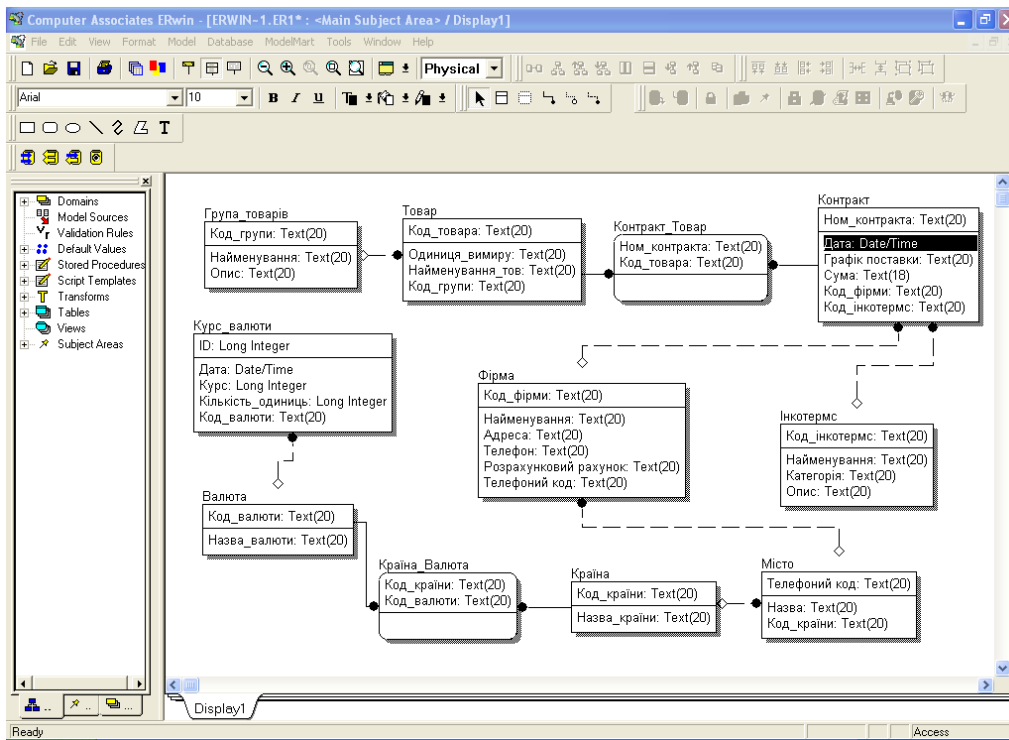


Рис. 4.18. Загальний вигляд фізичної моделі

Отримана фізична модель найчастіше вимагає редагування, тому що таблиці зв'язку (або віртуальні таблиці) мають зчеплені первинні ключі, що бажано перетворити за допомогою редактора зв'язків. Для цього необхідно клацнути правою кнопкою миші на зв'язку й у вікні вкладки, що відкрилося, вибрати тип зв'язку **Non-Identifying** (рис. 4.19). Потім для кожної таблиці зв'язку необхідно встановити первинні ключі.

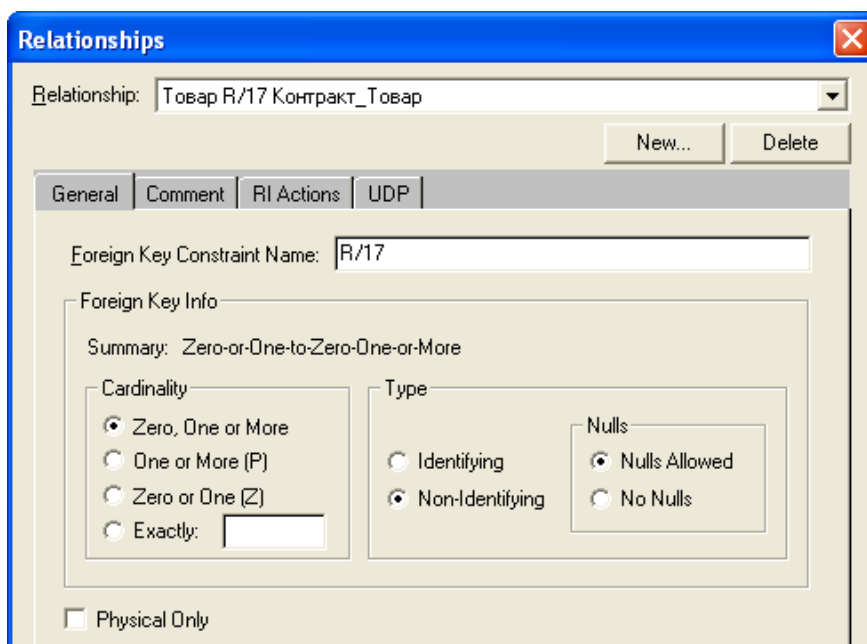


Рис. 4.19. Вибір неідентифікуючого зв'язку

Остаточно фізична модель буде мати вигляд, показаний на рис. 4.20.

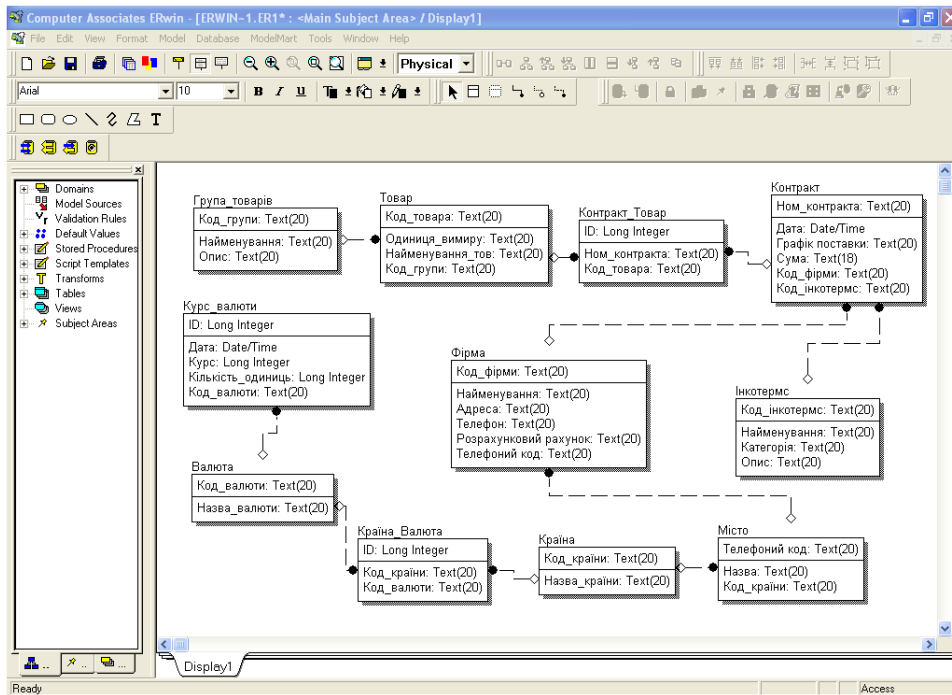


Рис. 4.20. Фізична модель БД після редагування

4.5. Генерація схеми БД у СУБД Ms Access

Для автоматичної генерації з ERwin схеми бази даних у СУБД Access необхідно попередньо створити нову базу даних (рис. 4.21, 4.22), привласнити ім'я й закрити її.

Далі відкрити пакет ERwin с розробленою фізичною моделлю (див. рис. 3.10), для цього необхідно переключитися на фізичну модель (**Model → Physical Model**), потім вибрати пункт меню **Database → Choose Database...** і в діалоговому вікні **Target Server**, що відкрилося, вибрати необхідний сервер СУБД, в яку буде генеруватися модель – Access, і натиснути кнопку ОК (рис. 4.23).

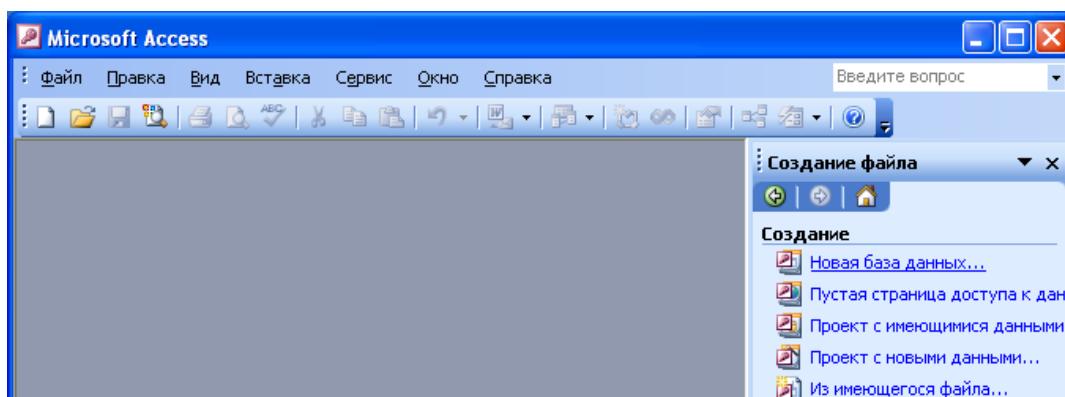


Рис. 4.21. Створення нової бази даних

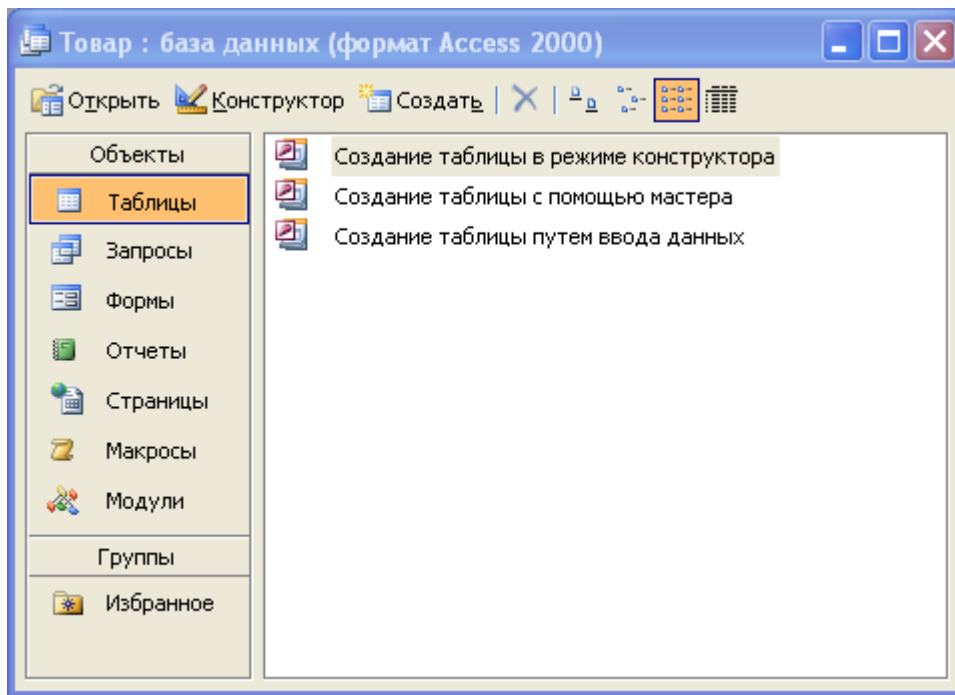


Рис. 4.22. Нова база даних Товар

Діалог **Target Server** дозволяє задати тип даних і опцію NOT NULL для нових стовпчиків, а також правила посилальної цілісності, прийняті за замовчуванням.

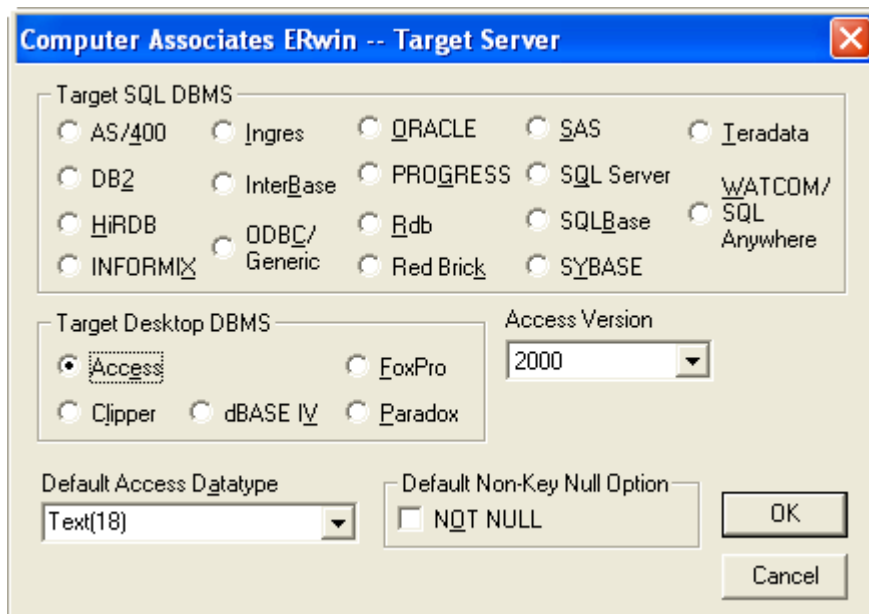


Рис. 4.23. Діалогове вікно Target Server

Потім необхідно підтвердити конвертацію (рис. 4.24)

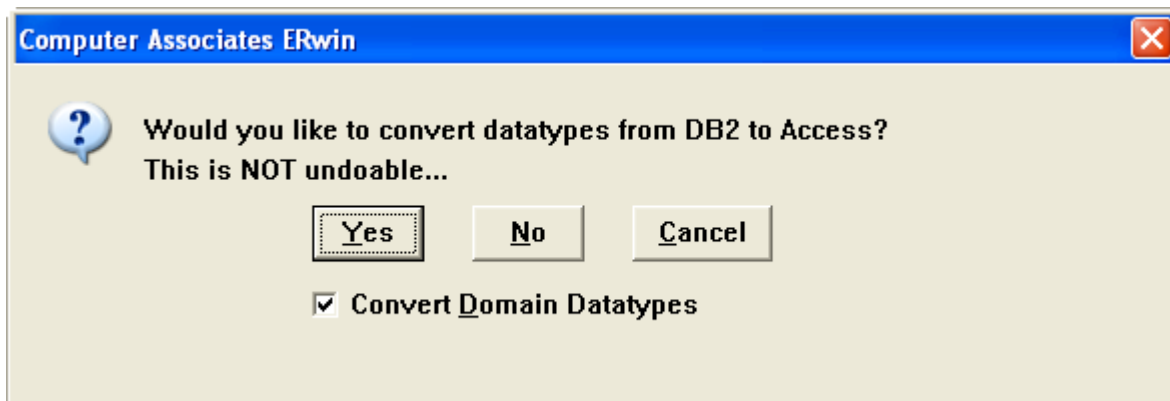


Рис. 4.24. Підтвердження конвертації (генерації схеми БД)

Після підтвердження вибрати пункт меню **Database → Database Connection...** і у вікні, що відкрилося, задати в поле **User Name** ім'я користувача **admin** (рис.4.25).

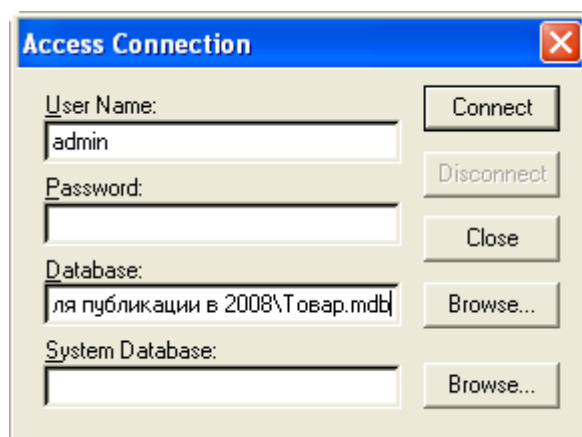


Рис. 4.25. Завдання імені користувача

Нагадаємо, що на рівні фізичної моделі сутності відповідають таблицям в реальній СУБД, атрибути – колонкам таблиці, зв'язки – зовнішнім ключам, первинним і альтернативним ключам – унікальні індекси, а інверсним входам не унікальні.

Останнім кроком є генерація схеми БД. Для цього в необхідно вибрати пункт меню **Tools → Forward Engineer → Schema Generation...** Усі необхідні параметри генерації схеми БД можна задати на призначеній для цього панелі діалогу Access **Schema Generation** (рис. 4.26). Натискання кнопки **Preview** дозволяє подивитися код, що буде створений автоматично ERwin (рис. 4.27). Генерація схеми БД запускається натисканням кнопки **Generate**. Двічі нажати кнопку ОК.

Потім відкрити заздалегідь створену БД (рис.4.28), у ній перебувають розроблені в ERwin таблиці. Для перегляду згенерованої схеми бази даних необхідно вибрати пункт меню **Сервис → Схема данных** (рис. 4.29).

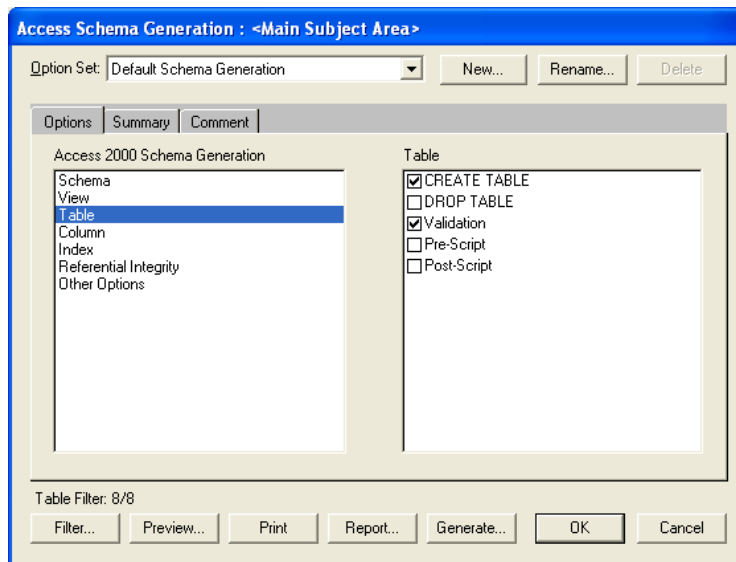


Рис. 4.26. **Діалогове вікно Access Schema Generation**

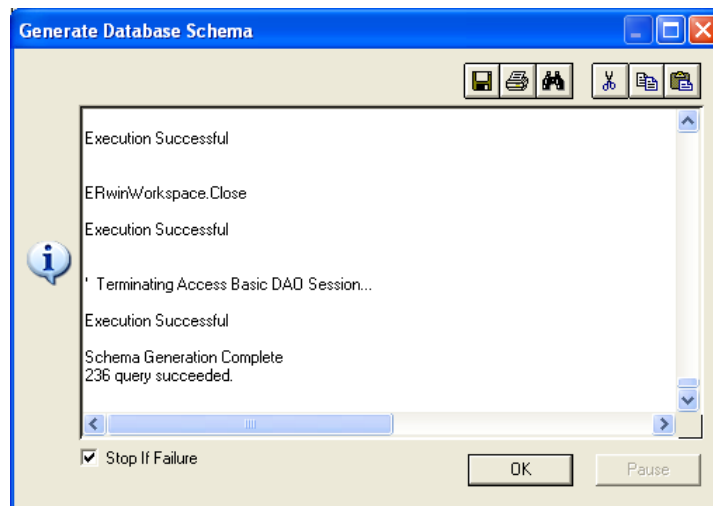


Рис. 4.27. **Вікно перегляду програмного коду**

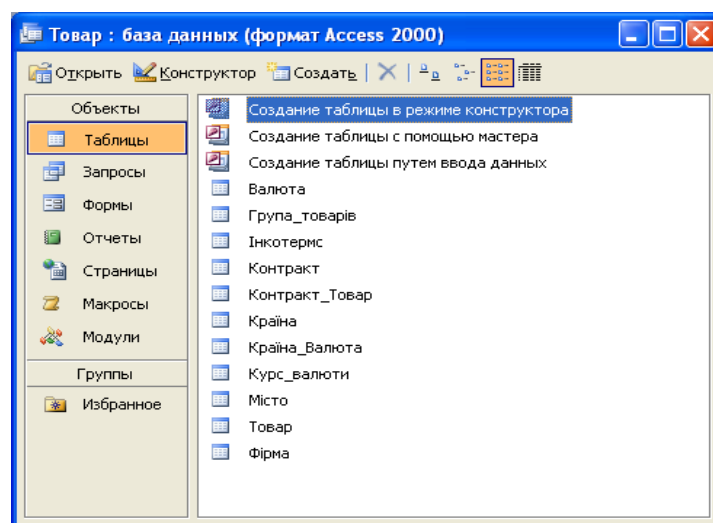


Рис. 4.28. **БД Товар в Ms Access з новими таблицями, згенерованими з ERwin**

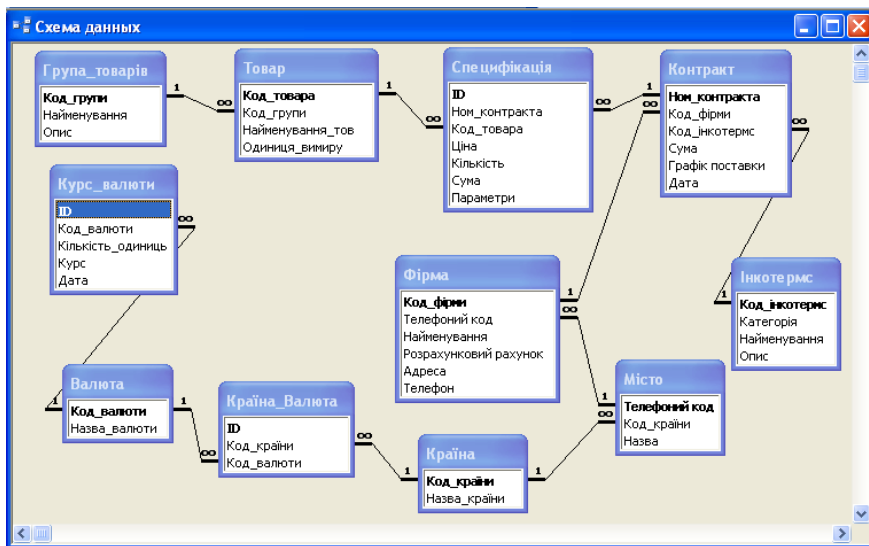


Рис.4.29. Схема (модель) даних БД .db1, отримана з Erwin

Звіт по лабораторній роботі

У звіт по лабораторній роботі необхідно включити:

1. Вхідні дані виконання роботи (запропонований варіант).
2. Контекстну діаграму в стандарті DFD та діаграми декомпозиції вузлів відповідного варіанту.
3. Результати експорту-імпорту даних з DFD-моделі в Erwin.
4. Логічну і відредаговану фізичну моделі в Erwin.
5. Результат автоматичної генерації схеми бази даних у СУБД Ms Access з ERwin.

Контрольні запитання

1. Що зображує діаграма DFD?
2. Яка нотація використовується в ERwin для побудови діаграм DFD?
3. Перерахуйте складові частини діаграми DFD.
4. Що називається зовнішньою сутністю?
5. Що описують сховища даних?
6. Що необхідно зробити, щоб додати сховище даних, котре буде характеризувати сутності об'єктів?
7. Дайте визначення потоку даних.
8. Що зображує діаграма IDEF1X?
9. Що необхідно для експорту даних з ERwin у ERwin?
10. Що таке сутність?
11. Що таке атрибут?
12. Які відмінності між ідентифікуючим і неідентифікуючим зв'язками між сутностями?
13. Що таке модель даних і які існують способи її представлення?
14. Для чого використовуються CASE-інструменти?

Рекомендована література

1. Вейскас Дж. Эффективная работа: Microsoft Office Access 2003. – СПб.: "Питер", 2005. – 1168 с.
2. Журавльова І. В. Інформаційні системи міжнародного бізнесу. Навчальний посібник / І. В. Журавльова, Р. М. Чен. – Харків: Вид. ХНЕУ. – 2006. – 452 с.
3. Журавльова І. В. Інформаційні системи та технології. Лабораторний практикум / І. В. Журавльова, Р. М. Чен, О. В. Ананьїна, О. В. Цурихин. – Харків: Вид. ХНЕУ, 2005. – 304 с.
4. Маклаков С. В. ВРwin и ERwin. CASE-средства разработки информационных систем. – М.: Диалог МИФИ, 2000. – 256 с.
5. Маклаков С. В. Моделирование бизнес-процессов с ВРwin 4.0. – М.: Диалог МИФИ, 2002. – 224 с.
6. Маклаков С. В. Создание информационных систем з AllFusion Modeling Suite. – М.: Диалог-мифи, 2003. – 432 с.
7. Федотова Д. Э. CASE-технологии: Практикум / Д. Э. Федотова, Ю. Д. Семенов, К. Н. Чижик. – М.: Горячая линия – Телеком, 2003. – 160 с.
8. Чен Р. М. Робоча програма з навчальної дисципліни "Інформаційні системи міжнародного бізнесу" для студентів спеціальності 8.050103 "Міжнародна економіка" всіх форм навчання. – Харків: Вид. ХНЕУ, 2007. – 52 с.
9. Чен Петер Пин-Шен. Модель "Сущность-связь" – шаг к единому представлению о данных. // Журнал "Системы управления базами данных". – 1995. – №3. С.137 – 158.
10. Чубук В. В. Бази даних у питаннях в відповідях. Навчальний посібник / В. В. Чубук, Р. М. Чен, Л. А. Павленко, В. І. Клименко. – Харків: Вид. ХНЕУ, 2004. – 288 с.

Теми, що рекомендуються, для виконання лабораторної роботи

1. Облік міжміських телефонних розмов.
2. Облік кадрів на підприємстві.
3. Складський облік.
4. Бібліотека.
5. Деканат.
6. Оптова база.
7. Управління гуртожитком.
8. Школа.
9. ВНЗ м. Харкова.
10. Оптовий магазин.
11. Супермаркет.
12. Фірма нерухомості.
13. Банк обслуговування клієнтів.
14. Ощадна каса.
15. Залізничний розклад.
16. Розклад аеропорту "Харків".
17. Їдальня.
18. Маркетингове дослідження місткості ринку.
19. Облік основних засобів підприємства.
20. Розклад занять.
21. Поліклініка.
22. Рекламне агентство.
23. Нотаріальна контора.
24. Страхова компанія.
25. Житлово-експлуатаційна контора.

НАВЧАЛЬНЕ ВИДАННЯ

**Методичні рекомендації
до виконання лабораторної роботи
з теми "Використання CASE-технологій у проектуванні
інформаційних систем та БД"
навчальної дисципліни
"ІНФОРМАЦІЙНІ СИСТЕМИ
МІЖНАРОДНОГО БІЗНЕСУ"
для студентів напрямку підготовки
"Міжнародна економіка"
всіх форм навчання**

Укладач **Чен Роза Миколаївна**

Відповідальний за випуск **Пономаренко В. С.**

Редактор **Демченко Н. І.**

Коректор **Бриль В. О.**

План 2008 р. Поз. №214.

Підп. до друку Формат 60 × 90 1/16. Папір MultiCopy. Друк Riso.

Ум.-друк. арк. 3,25. Обл.-вид. арк. 3,74. Тираж прим. Зам. №

*Свідоцтво про внесення до Державного реєстру суб'єктів видавничої справи
Дк №481 від 13.06.2001 р.*

Видавець і виготівник — видавництво ХНЕУ, 61001, м. Харків, пр. Леніна, 9а

Методичні рекомендації
до виконання лабораторної роботи
з теми "Використання CASE-технологій у проектуванні інформаційних систем та БД"
навчальної дисципліни
"ІНФОРМАЦІЙНІ СИСТЕМИ МІЖНАРОДНОГО БІЗНЕСУ"
для студентів напряму підготовки
"Міжнародна економіка"
всіх форм навчання