

ЛЕКЦІЯ. КОНЦЕПТУАЛЬНЕ ПРОЕКТУВАННЯ БАЗ ДАНИХ

План:

1. Концептуальні моделі.
2. Модель «сутність-зв'язок».
3. Розширена модель "сутність – зв'язок".
4. Проблеми побудови моделей "сутність – зв'язок".
5. Приклад побудови моделі "сутність – зв'язок".

Література

1. Гайна Г.А. Основи проектування баз даних: Навчальний посібник. К.: КНУБА, 2005. – 204 с.
2. Гайна Г.А. Організація баз даних і знань. Мови баз даних: Конспект лекцій.–К.:КНУБА, 2002. – 64 с.
3. Гайна Г.А., Попович Н.Л. Організація баз даних і знань. Організація реляційних баз даних: Конспект лекцій.–К.:КНУБА, 2000. – 76 с.

1. Концептуальні моделі.

З концептуального проектування починається створення концептуальної схеми БД, в основі якої лежить концептуальна модель даних. Концептуальна модель представляє загальний погляд на дані. Розрізняють два головних підходи до моделювання даних при концептуальному проектуванні:

- семантичні моделі;
- об'єктні моделі.

Семантичні моделі головну увагу приділяють структурі даних. Найбільш поширеною семантичною моделлю є модель "сутність – зв'язок" (Entity Relationship model, ER-модель). ER-модель складається із сутностей, зв'язків, атрибутів, доменів атрибутів, ключів. Моделювання даних відображає логічну структуру даних, так само, як блок-схеми алгоритмів відображають логічну структуру програми.

Об'єктні моделі головну увагу приділяють поведінці об'єктів даних і засобам маніпуляції даними. Головне поняття таких моделей – об'єкт, тобто сутність, яка має стан і поведінку. Стан об'єкта визначається сукупністю його атрибутів, а поведінка об'єкта визначається сукупністю операцій специфікованих для нього.

Зближення цих моделей реалізується в розширеному ER-моделюванні (Extended Entity Relationship model, EER-модель).

2. Модель сутність-зв'язок.

ER-моделювання являє собою низхідний підхід до проектування БД, який починається з визначення найбільш важливих даних, які називаються сутностями (entities), і зв'язків (relationships) між даними, які повинні бути представлені в моделі. Потім в модель заноситься інформація про властивості сутностей і зв'язків, яка називається атрибутами (attributes), а також всі обмеження, які відносяться до сутностей, зв'язків і атрибутів. ER-модель дає графічне представлення логічних об'єктів і їх відношень в структурі БД. Послідовність проведення ER-моделювання показана на рис. 1.

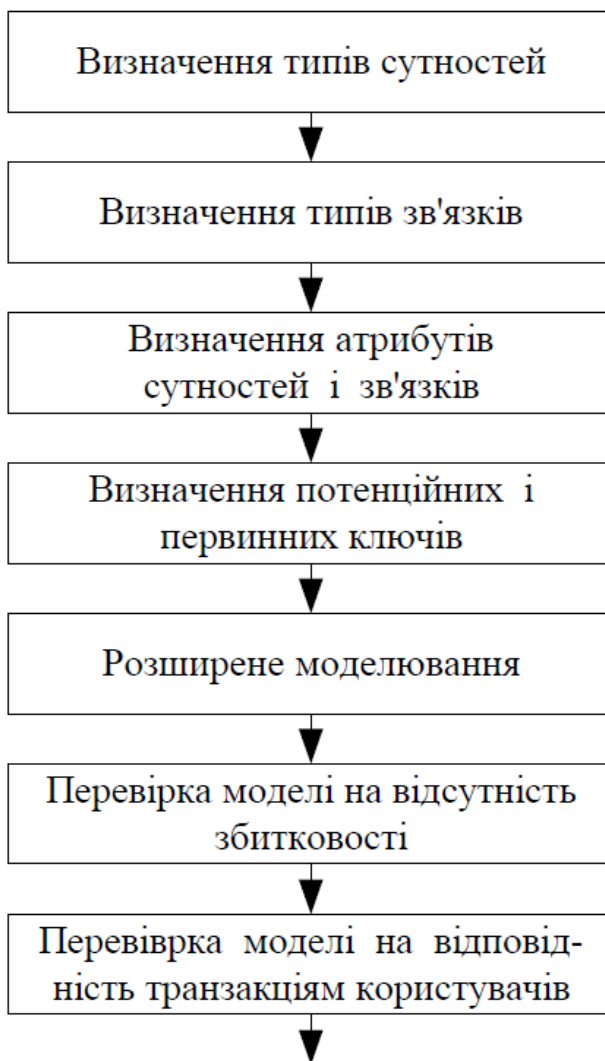


Рис. 1. Етапи побудови моделі "сутність – зв'язок".

Вперше поняття ER-моделі запровадив П.Чен. Підхід П.Чена дозволив концептуальне моделювання перевести в практичну площину проектування БД. У подальшому діаграми Чена набули розвитку у багатьох інших роботах з ER-моделювання. До них належать такі моделі:

- "пташина лапка", розроблена К.М. Бахманом;
- IDEF1X, розроблена Т.Ремеєм;
- на основі UML;
- модель Баркера і багато інших моделей.

Сутності

Сутність дозволяє моделювати клас однотипних об'єктів. Сутність має унікальне ім'я у межах системи, що моделюється. Оскільки сутність відповідає деякому класу однотипних об'єктів, то передбачається, що в системі існує багато екземплярів даної сутності. Об'єкт, якому відповідає сутність, має набір атрибутів, які характеризують його властивості. При цьому набір атрибутів повинен бути таким, щоби можна було розрізняти конкретні екземпляри сутності.

Приклад. Сутність Викладач може мати такі атрибути:

Табельний номер, Прізвище, Ім'я, По батькові, Посада, Вчений ступінь.

Набір атрибутів, що однозначно ідентифікує конкретний екземпляр сутності, називають ключовим. У наведеному прикладі для сутності Викладач ключем буде Табельний номер, оскільки для всіх викладачів табельні номери різні. Екземпляром сутності Викладач буде опис конкретного викладача. Загальноприйняте позначення сутності – прямокутник (рис. 2).

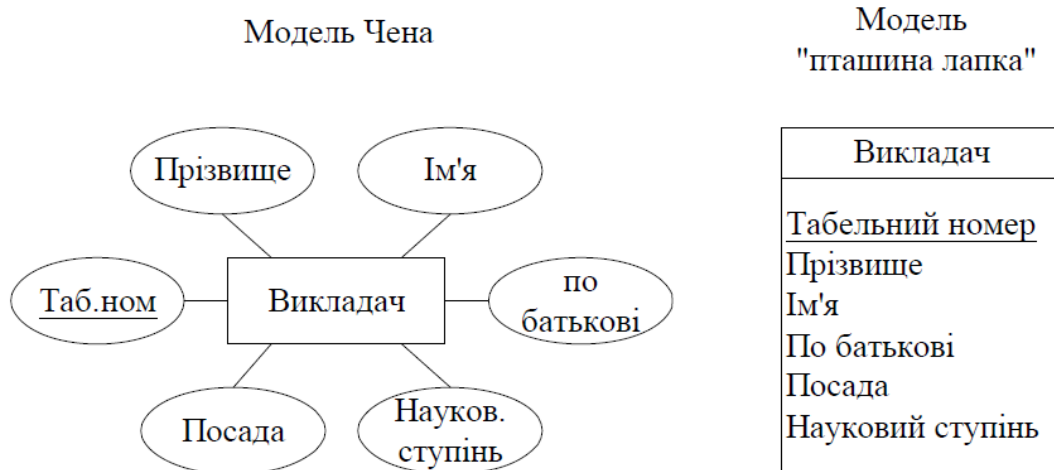


Рис. 2. Представлення сутностей і атрибутів у ER-діаграмах П. Чена і ER-діаграмах "пташина лапка"

Зв'язки

Між сутностями встановлюються зв'язки, які вказують яким чином сутності співвідносяться або взаємодіють між собою. Розрізняють такі зв'язки:

- між двома сутностями (бінарний зв'язок);
- між трьома сутностями (тернарний зв'язок);
- між N сутностями (N-арний зв'язок);
- між однією сутністю (рекурсивний зв'язок).

Найбільш поширеними є бінарні зв'язки. Зв'язок показує яким чином екземпляри сутностей зв'язані між собою. Бінарні зв'язки бувають:

- 1:1 (один до одного);
- 1:M (один до багатьох);
- N:M (багато до багатьох).

На рис. 3, 4 показані відображення цих зв'язків у різних ER-моделях.

Зв'язок "один до одного" (1:1): завідуючий кафедрою може керувати тільки однією кафедрою, а кожною кафедрою керує тільки один завідуючий



а

Зв'язок "один до багатьох" (1:М): на кафедрі працює багато викладачів, а кожен викладач працює тільки на одній кафедрі



б

Рис. 3. Представлення зв'язків між відношеннями на діаграмі Чена: а – 1:1;

б – 1:М

Зв'язок "багато до багатьох" (N:M): студент займається у багатьох викладачів, а кожен викладач навчає багатьох студентів

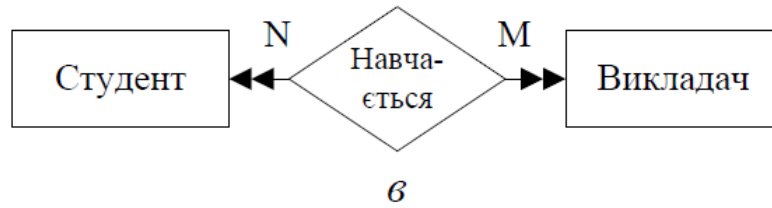


Рис. 5.3. Закінчення: *в* – N:M

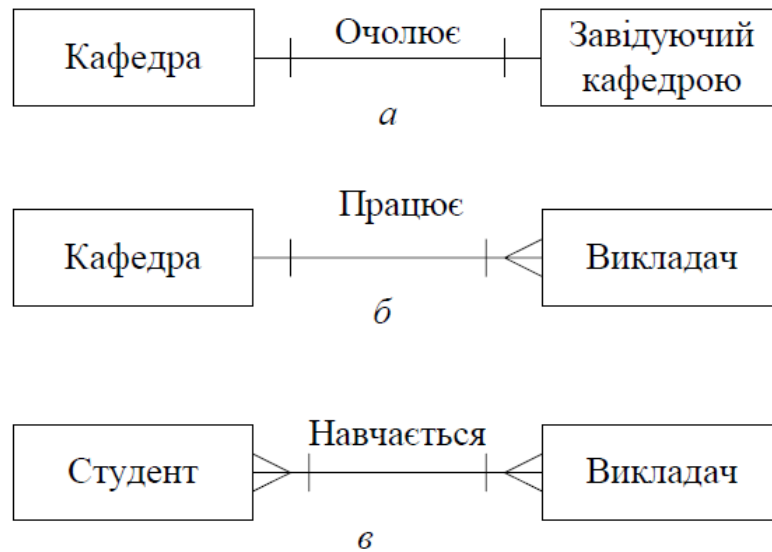


Рис. 4. Представлення зв'язків між відношеннями на діаграмі "пташина лапка":

а – 1:1; б – 1:M; в – N:M

Атрибути

Атрибути являють собою властивості сутності. Значення кожного атрибута вибирають з відповідної множини значень, яка включає всі потенційні значення, які можуть бути присвоєні атрибуту. Ця множина значень називається доменом.

Приклад. Атрибут Оцінка може приймати чотири значення: 2, 3, 4, 5. Ці значення і складають домен цього атрибута. Атрибути залежно від складності значень, які вони можуть приймати поділяються на певні категорії (табл. 1).

Таблиця 1

Типи атрибутів

Тип	Властивість
Простий	Атрибут, який не може бути поділений

	на інші атрибути. Приклад. Прізвище; посада
Складовий	Атрибут, який може бути поділений на інші атрибути. Приклад. Адреса; прізвище, ім'я, по батькові
Однозначний	Атрибут, який може приймати тільки одне значення. Приклад. Табельний номер, номер залікової книжки
Багатозначний	Атрибут, який може приймати багато значень. Приклад. Телефон, Адреса (постійне місце проживання і гуртожиток)
Похідний	Атрибут, який не зберігається в БД, а обчислюється за допомогою певного алгоритма Приклад. Вік (обчислюється по даті народження), кількість студентів в групі

Приклад. Розглянемо сутність *Студент* (рис. 5).

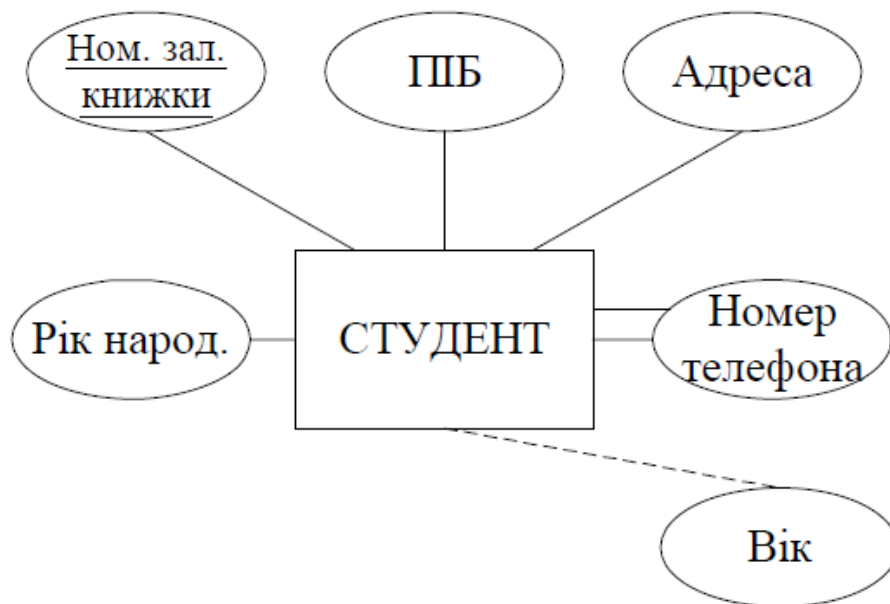


Рис. 5. ER-діаграма сутності Студент

Атрибути номер залікової книжки, рік народження є простими.

Атрибути ПІБ і Адреса є складовими. ПІБ може бути поділений на атрибути: прізвище, ім'я, по батькові, а Адреса – на індекс, місто, вулиця, будівля, квартира.

Атрибут Вік є похідним: він обчислюється за значенням атрибута Рік народження (зображається пунктирною лінією).

Атрибут Номер залікової книжки є однозначним: він не може приймати два значення для одного студента.

Атрибут Номер телефону є багатозначним: він може приймати декілька значень для одного студента (зображається подвійною лінією).

Атрибут або набір атрибутів сутності, які застосовуються для ідентифікації екземпляра сутності, називаються потенційним ключем. Сутність може містити декілька потенційних ключів. В прикладі в якості потенційних ключів можуть бути такі атрибути: Номер залікової книжки, Прізвище Ім'я по Батькові.

Потенційний ключ, який вибрано для однозначної ідентифікації кожного екземпляра сутності певного типу, називається первинним ключем. Первинний ключ вибирається за умови гарантії унікальності його значень, а також мінімальної довжини атрибутів, які входять в його склад. В прикладі в якості первинного ключа служить Номер залікової книжки.

Потужність зв'язків.

Потужність зв'язку (кардинальність) відображає певне число екземплярів сутностей, які зв'язані з одним екземпляром зв'язаної сутності. В моделі Чена потужність зв'язку відображається вказівкою відповідних чисел поруч з сутностями у форматі (x, y). Перше число визначає мінімальне значення потужності зв'язку, а друге – його максимальне значення. Потужність вказує на число екземплярів у зв'язаній сутності.

Відомості про максимальне і мінімальне значення потужності зв'язку може застосовуватися у прикладному програмному забезпеченні або за допомогою тригерів; на рівні таблиць СУБД не може оперувати з потужністю зв'язків.

Приклад. Розглянемо зв'язок Група-Студент (рис. 6).

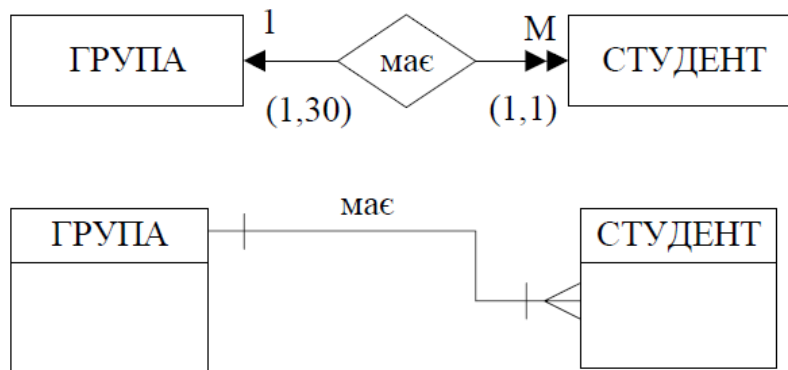


Рис. 6. Зв'язок і потужність в ER-діаграмах

Потужність (1,30) поруч із сутністю Група вказує, що в групі може займатися від 1 до 30 студентів. Потужність (1,1) поруч із сутністю Студент вказує, що студент може займатися тільки в одній групі (мінімум 1, максимум 1). У моделі "пташина лапка" числовий діапазон значень потужності не відображається в ER-діаграмах.

Сильні і слабкі зв'язки.

Якщо сутність може існувати незалежно від інших сутностей, то вона є незалежною від існування. Якщо сутність залежить від існування інших сутностей, то вона є залежною від існування. Наприклад, сутності Студент і Група можуть існувати незалежно одна від одної, а сутність Нагорода студента є залежною від сутності Студент й існувати без неї не може.

Якщо одна сутність незалежна від існування іншої сутності, то зв'язок між ними називається не ідентифікаційним зв'язком або слабким зв'язком. На ER-діаграмах "пташина лапка" слабкий зв'язок відображається штриховою лінією.

Ідентифікаційний зв'язок або сильний зв'язок має місце у тому випадку, коли одна зв'язана сутність залежить від існування іншої. На ER-діаграмах "пташина лапка" сильний зв'язок відображається суцільною лінією.

Приклад. Розглянемо зв'язки Група-Студент і Студент-Нагорода (рис. 7).

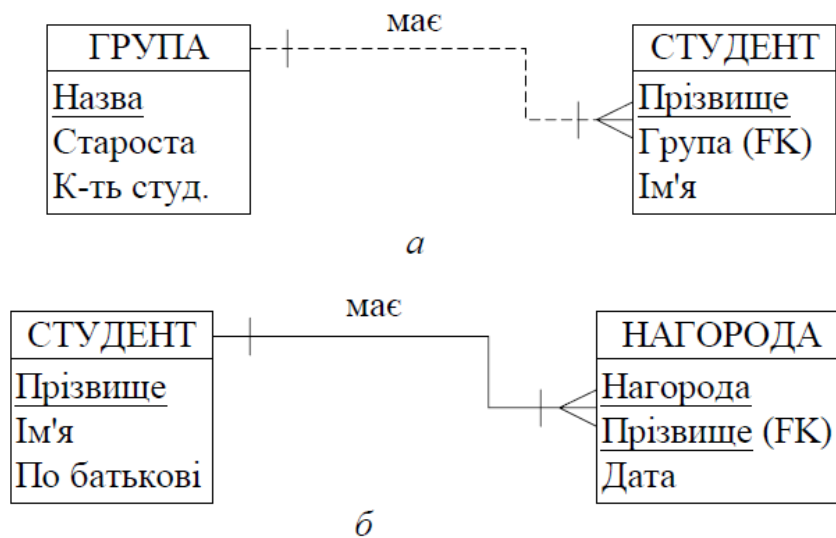


Рис. 7. Неідентифікаційний (а) та ідентифікаційний (б) зв'язки

Сутність Студент не залежить від сутності Група, в цьому випадку зв'язок між сутностями відображається штриховою лінією, а атрибут Назва групи в сутності Студент є зовнішнім ключем (Foreign Key, FK).

Сутність Нагорода залежить від сутності Студент, в цьому випадку зв'язок між сутностями відображається суцільною лінією, а атрибут Прізвище студента в сутності Нагорода є частиною первинного ключа (Primary Key, PK) і одночасно зовнішнім ключем (FK).

Атрибути зв'язків.

Так само як і сутності зв'язки можуть мати атрибути.

Приклад. Атрибути День і Аудиторія належать до зв'язку між сутностями Студент і Дисципліна (рис. 8).

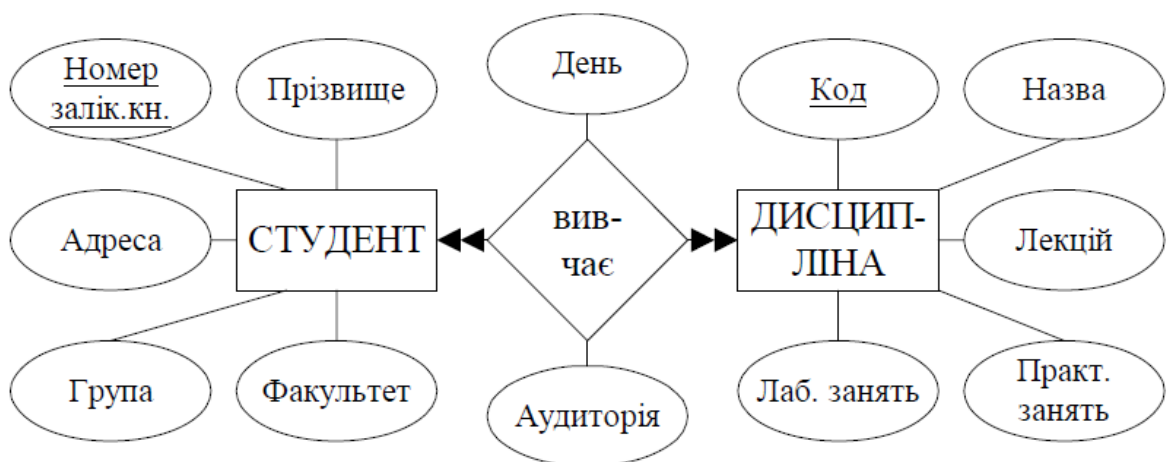


Рис. 8. ER-діаграма з атрибутами зв'язку День і Аудиторія

Обов'язкові і необов'язкові зв'язки.

Участь сутності у зв'язку може бути обов'язковою або необов'язковою. Якщо один екземпляр сутності не потребує наявності відповідного екземпляра сутності в окремому зв'язку, то участь сутності у зв'язку є необов'язковою. Наприклад, у зв'язку сутностей Студент-Нагорода – не кожен студент має нагороди. Тобто не кожен екземпляр в таблиці Студент потребує обов'язкової наявності екземпляра сутності в таблиці Нагорода. Сутність Нагорода розглядається як необов'язкова по відношенню до сутності Студент. Необов'язкова сутність позначається невеликим колом з боку необов'язкової сутності. Існування необов'язковості вказує на те, що для необов'язкової сутності мінімальне значення потужності зв'язку дорівнює 0.

Участь сутності у зв'язку буде обов'язковою, якщо кожен екземпляр сутності обов'язково потребує відповідного екземпляра сутності в окремому зв'язку. При обов'язковому зв'язку для обов'язкової сутності мінімальна потужність зв'язку дорівнює 1.

Приклад. Зв'язок між сутностями Студент і Нагорода є необов'язковим (рис. 9). Необов'язково кожен студент має нагороду, але якщо є нагорода, то вона обов'язково зв'язана з певним студентом.

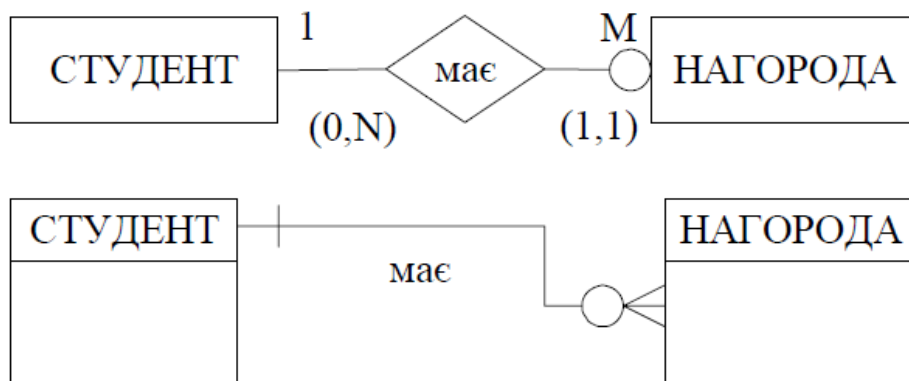


Рис. 9. Необов'язкова сутність Нагорода в ER-діаграмах П. Чена і "пташина лапка"

Слабкі сутності.

Слабкою сутністю називається сутність, яка задовольняє таким умовам:

- залежності від існування сутності з якою вона зв'язана;
- первинний ключ цієї сутності частково або повністю отриман з іншої сутності.

Слабка сутність на діаграмі Чена позначається подвійним прямокутником, а на діаграмі "пташина лапка" невеликими сегментами в кожному з кутів прямокутника.

Приклад. Сутність Нагорода є слабкою по відношенню до сутності Студент: вона залежить від існування цієї сутності і в її первинний ключ входить первинний ключ сутності Студент (рис. 10).

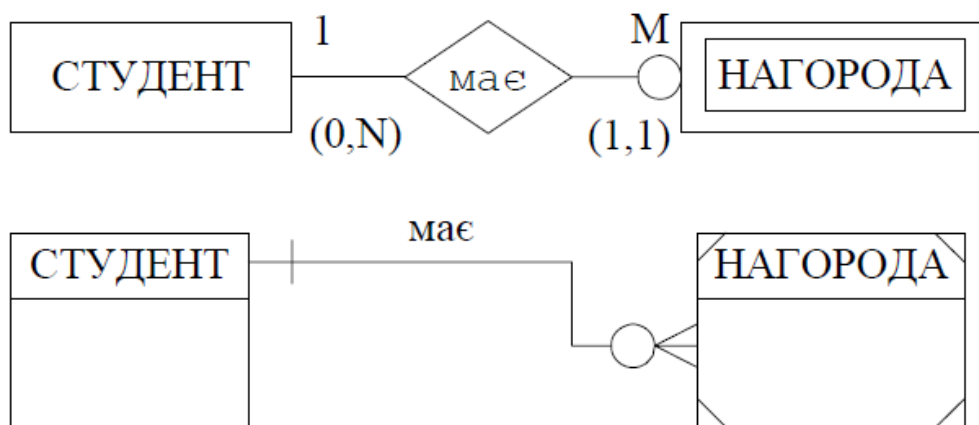


Рис. 10. Слабка сутність на діаграмах П. Чена і "пташина лапка"

Складні зв'язки.

Використання зв'язків більш високого порядку дає можливість у багатьох випадках краще відобразити семантику проблемної області.

Приклад. Сутності Викладач, Дисципліна і Екзамен утворюють тернарний зв'язок (рис. 11).

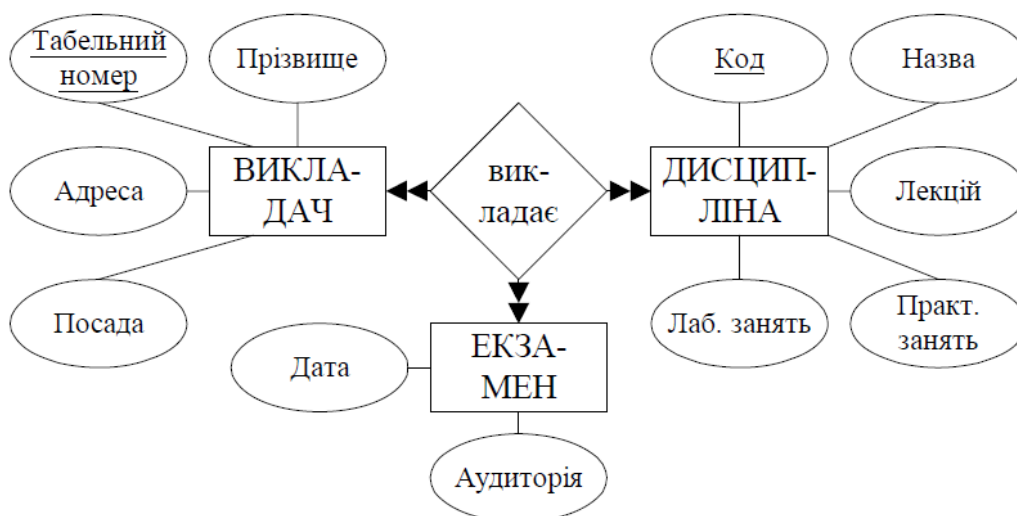


Рис. 11. Тернарний зв'язок між трьома сутностями на діаграмі П. Чена

Рекурсивні зв'язки/

Рекурсивний зв'язок має місце, коли є зв'язок між екземплярами одного і того ж набору сутностей.

Приклад. Розглянемо можливі варіанти рекурсивних зв'язків (рис. 12).

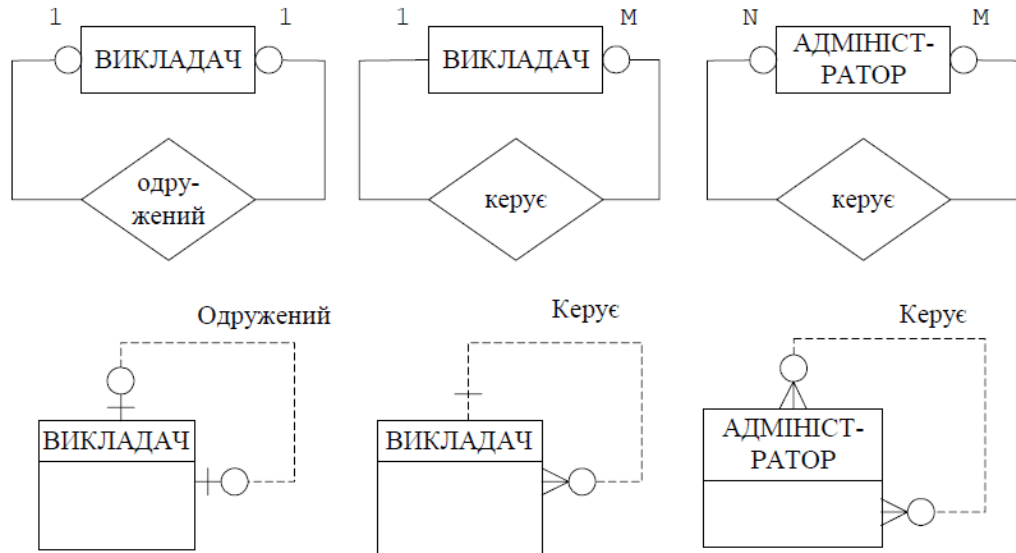


Рис. 12. Представлення рекурсивних зв'язків на діаграмі П. Чена і моделі "пташина лапка"

Зв'язок 1:1 представляє висловлювання: "викладач може бути одружений тільки з одним співробітником". Зв'язок 1:M представляє таке висловлювання: "викладач, якщо він є завідуючим кафедрою, керує декількома викладачами, а викладачі мають тільки одного керівника – завідуючого кафедрою". Зв'язок M:N представляє висловлювання: "адміністратор має декілька підлеглих-адміністраторів, і в свою чергу адміністратор має декілька керівників-адміністраторів".

Між двома сутностями може бути декілька зв'язків з різними змістовними навантаженнями.

Приклад. Між сутностями Викладач і Студент можна встановити такі змістовні зв'язки: Викладає і Керівництво дипломним проектуванням. Викладач викладає для багатьох студентів, для кожного студента викладає багато викладачів.

Кожен студент обов'язково повинен мати одного керівника дипломного проекту, але необов'язково кожен викладач керує дипломниками (рис. 13).

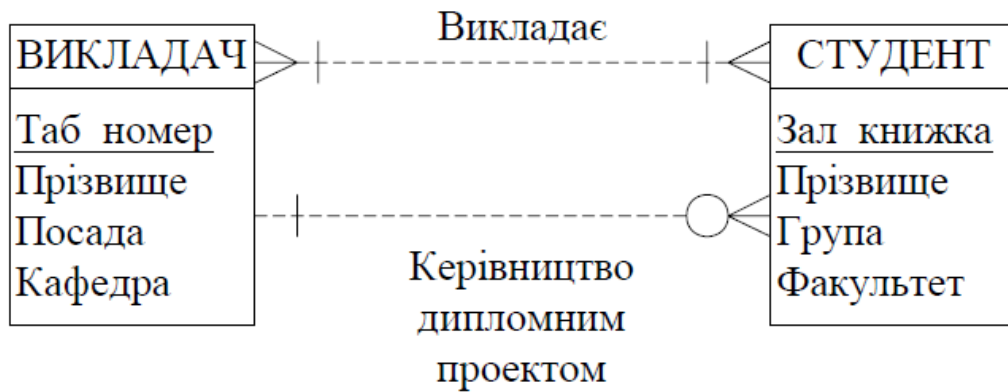


Рис. 13. Різні зв'язки між двома сутностями

3. Розширена модель "сутність – зв'язок".

Для задоволення нових потреб, що висуваються більш складними застосуваннями, в семантичне моделювання були введені додаткові концепції, що розширюють його можливості. Така модель має назву розширеної ER-моделі (Enhanced Entity Relationship, EER-модель). Вона включає всі концепції ER-моделі плюс концепції *уточнення, узагальнення, агрегування і композиції*. Додаткові концепції базуються на таких поняттях, як суперклас і підклас. Суперклас може мати декілька підкласів. Наприклад підкласи *Викладач, Керівник, Лаборант* є членами суперкласу *Співробітник*. Це означає, що кожен екземпляр підкласу є в той же час і екземпляром суперкласу. Зв'язок між суперкласом і підкласом відноситься до типу 1:1.

Використання понять суперклас і підклас дозволяє визначити для підкласів власні атрибути і атрибути, що наслідуються від суперкласу. Так, наприклад, підклас *Викладач* повинен мати ті ж атрибути, що і всі *Співробітники*. Однак він має і свої власні атрибути, які не визначені для інших категорій працівників університету. До цих атрибутів можна віднести вчене звання, номер диплому про вчене звання, кількість навчально-методичних праць тощо. При відсутності підкласів для об'єкту *Співробітник* слід було б вводити атрибути, які б мали невизначене значення для інших співробітників (наприклад для лаборантів). Підклас може мати свої власні зв'язки, які не підходять для всіх екземплярів суперкласу. Наприклад, *Викладач* може мати підкласи *Професор, Доцент, Асистент*. Підклас наслідує не тільки атрибути, але і всі зв'язки суперкласу.

Уточнення це процес збільшення різниці між окремими екземплярами об'єкта за рахунок визначення їхніх відмінних характеристик. Цей процес є низхідним. Наприклад, перехід від об'єкта *Співробітник* до об'єктів *Викладач* і *Керівник*.

Узагальнення це процес зведення відмінностей між об'єктами до мінімуму шляхом виділення їх спільних характеристик. Цей процес є висхідним. Наприклад, перехід від об'єктів *Викладач* і *Керівник* до об'єкта *Співробітник*.

У процесі проведення уточнення або узагальнення можуть застосовуватися обмеження:

- ступеня участі;
- неперетинання.

Підкласи набору сутностей можуть перетинатися і не перетинатися. Якщо підкласи суперкласу *не перетинаються*, то це означає, що кожен екземпляр сутності може бути елементом тільки одного з підкласів (позначається *Or*). Зв'язки, які не перетинаються позначаються символом "*G*". Наприклад, співробітник може працювати або на посаді доцента, або на посаді професора, і не може бути одночасно і професором, і доцентом.

Якщо підкласи суперкласу *перетинаються*, то це означає, що будь-який екземпляр сутності може бути елементом декількох з підкласів (позначається *And*). Зв'язки які перетинаються позначаються символом "*Gs*". Наприклад, завідуючий кафедрою проводить заняття, і одночасно виконує обов'язки викладача і керівника. На рис. 14 показана ієрархія сутностей з підкласами, що перетинаються і не перетинаються.

Приклад. Відношення суперклас – підклас.

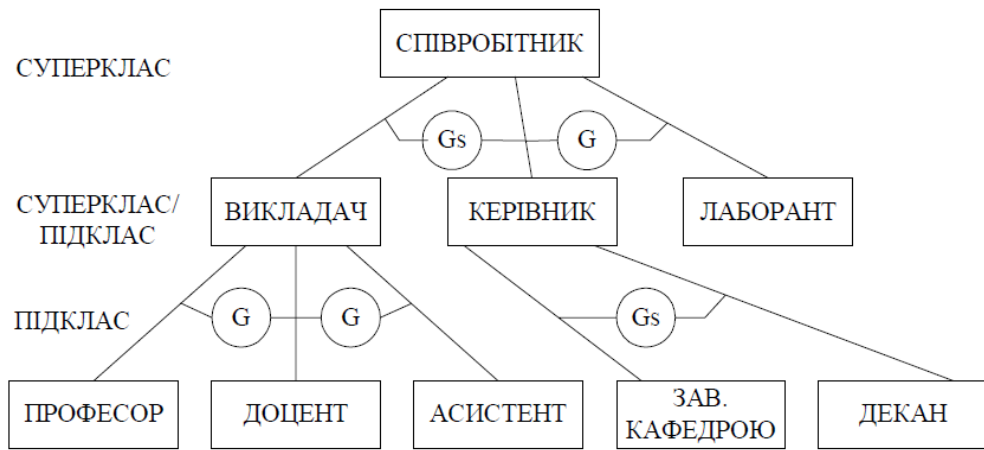


Рис. 14. Діаграма з використанням понять суперклас і підклас

Не кожен елемент суперкласу повинен бути елементом одного з підкласів.

Зв'язок суперкласу з підкласом з *обов'язковою участю*, вказує на те, що кожен елемент суперкласу повинен бути також елементом підкласу (*Mandatory*).

Приклад. Студент обов'язково займається або на денній, або на заочній, або на вечірній формі навчання, або екстернатом (рис. 15).

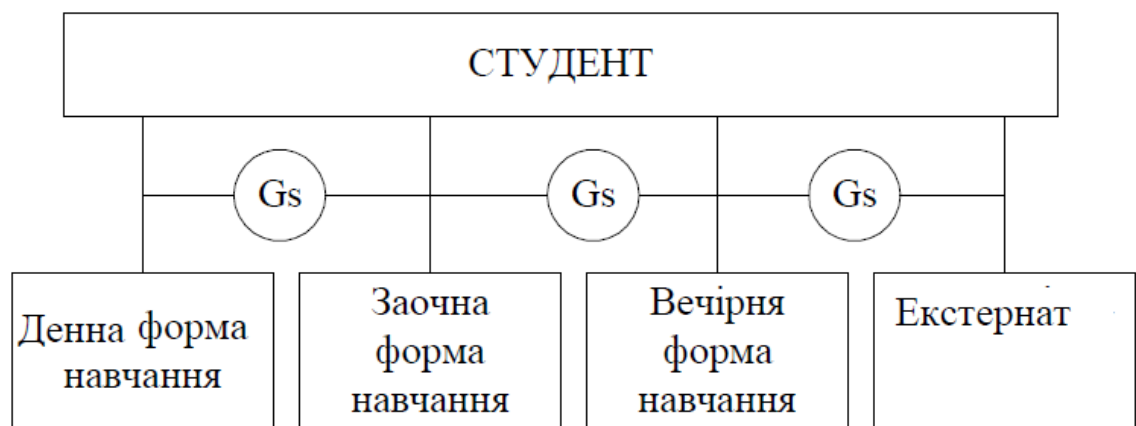


Рис. 15. Діаграма зв'язку суперкласу з підкласом з обов'язковою участю

Зв'язок суперкласу з підкласом з *необов'язковою участю*, вказує на те, що деякі елементи суперкласу можуть не належати жодному з підкласів (*Optional*). Наприклад, можуть бути викладачі, які не займають посади професора, доцента або асистента (наприклад старший викладач).

Введення понять суперкласів і підкласів дозволяє уникнути опису різних екземплярів сутності, які можуть мати різні атрибути. Це може привести до появи великої кількості незаповнених атрибутів. До того ж індивідуальні

атрибути можуть показати зв'язки, які притаманні тільки для конкретних екземплярів, а не всім екземплярам сутностей.

Приклад. Екземпляри сутності *Викладач*, які належать до викладачів на посаді професора, у порівнянні з викладачами на посаді асистента, мають такі додаткові атрибути: вчене звання, вчений ступінь і т.ін. Крім того, вони можуть бути зв'язані індивідуальними атрибутами із сутністю *Аспірант*.

Один підклас може бути зв'язаний з декількома суперкласами, які в свою чергу є підкласами одного спільного для них суперкласу.

Приклад. Підкласи *Викладач* і *Керівник* наслідують суперклас *Співробітник*. Підклас *Завідуючий кафедрою* є одночасно екземпляром суперкласу *Викладач* і суперкласу *Керівник* (рис. 16).

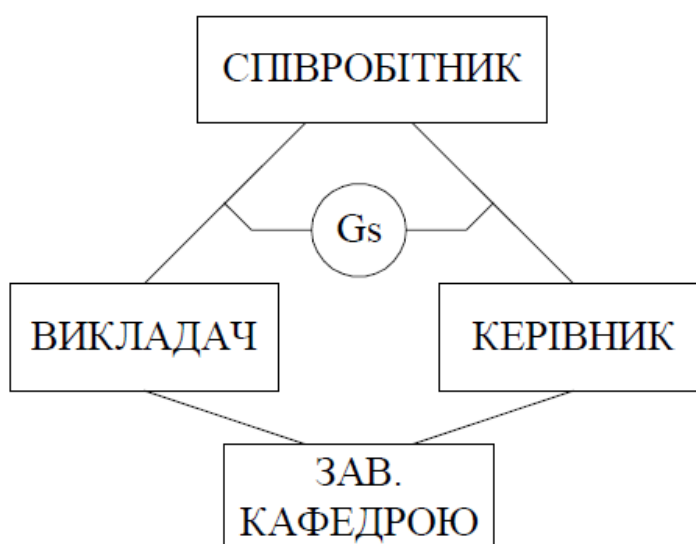


Рис. 16. Діаграма з підкласом, що сумісно використовується

Підклас є *категорією*, якщо він зв'язаний одразу з декількома суперкласами різних типів. Категорія може бути з повною участю і з частковою участю екземплярів. Повна участь передбачає, що кожен екземпляр всіх суперкласі повинен бути представлений у даній категорії. При частковій участі присутність в категорії всіх екземплярів всіх суперкласі необов'язкова.

Введення понять суперкласів і підкласів дозволяє ввести в проект більший обсяг семантичної інформації. Наприклад, твердження "Асистент є викладачем" дозволяє встановити певні ієрархічні відношення між об'єктами *Асистент* і *Викладач*. EER-діаграми повинні використовуватися, якщо структура даних є

занадто складною, для того щоби її можна було легко представити з використанням ER-діаграм.

Для моделювання інших видів зв'язків вводиться поняття агрегування і композиції.

Агрегування являє собою зв'язок типу "входить в склад" або "включає" між двома сутностями, одна з яких представляє "ціле", а інша – "частину".

Композиція – це особлива форма агрегування, яка представляє залежність між сутностями, що характеризуються повною приналежністю і співпаданням термінів існування між "цілим" і "частиною".

4. Проблеми побудови моделей "сутність – зв'язок".

При недостатньому розумінні суті встановлених зв'язків може бути створена модель, яка не буде повною мірою відображати зв'язки між реальними об'єктами. Визначають *дефекти з'єднання*, які виникають при невірній інтерпретації змісту деяких зв'язків: дефекти розгалуження і дефекти розриву.

Дефекти розгалуження мають місце, коли модель вірно відображає зв'язки між сутностями, але шлях між окремими сутностями визначений неоднозначно. Цей дефект виникає в тому випадку, коли два або більше зв'язків типу 1:М виходять з однієї сутності.

Приклад. Розглянемо такі зв'язки: на факультеті займається багато студентів, у склад факультету входить багато груп (рис. 17). Ці зв'язки вірно відображають зміст предметної області, але при спробі з'ясувати, в яких групах займаються конкретні студенти, виникають проблеми. Із сутності *Факультет* виходять два зв'язки 1:М.

Усунути цю проблему можна шляхом перебудови моделі для представлення вірної взаємодії цих сутностей (рис. 18).

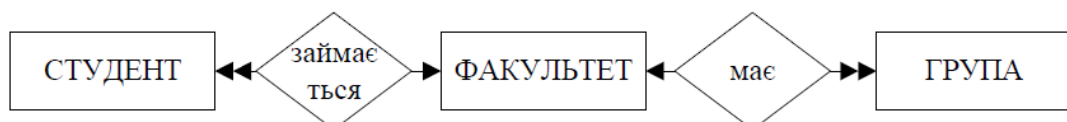


Рис. 17. Приклад дефекту розгалуження в ER-моделі

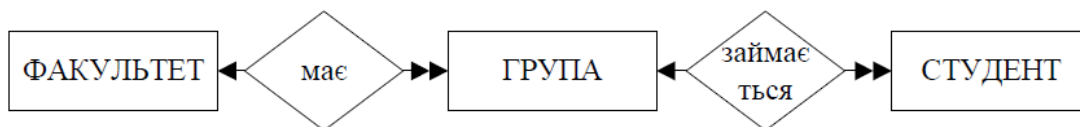


Рис. 18. Усунення дефекту розгалуження в прикладі ER-моделі

Отже, тепер відповідь на попереднє питання не є проблемою.

Дефекти розриву виникають у тому випадку, коли в моделі передбачається наявність зв'язку між декількома сутностями. Цей дефект виникає у разі, коли існує один або декілька зв'язків з мінімальною потужністю рівною 0, яка визначає необов'язкову участь, і ці зв'язки складають частину шляху між взаємозв'язаними сутностями.

Приклад. Розглянемо такі зв'язки: в склад факультету входить багато кафедр, кожна кафедра може відповідати за декілька комп'ютерних класів (від 0 до M) (рис. 19). Тобто деякі кафедри можуть не бути відповідальними за комп'ютерний клас. У свою чергу комп'ютерний клас може підпорядковуватися певній кафедрі, а може підпорядковуватися безпосередньо факультету (факультетський комп'ютерний клас). Ці зв'язки вірно відображають зміст предметної області, але при спробі з'ясувати, які комп'ютерні класи підпорядковані певному факультету, виникають проблеми. Зв'язок між сутностями *Кафедра* і *Комп'ютерний клас* передбачає необов'язкову участь сутностей, і він є частиною шляху між сутностями *Факультет* і *Кафедра*.

Усунути цю проблему можна шляхом введення додаткового зв'язку між сутностями *Факультет* і *Комп'ютерний клас* (рис. 20).

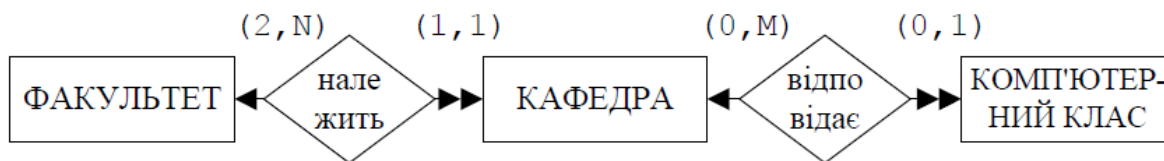


Рис. 19. Приклад дефекту розриву в ER-моделі

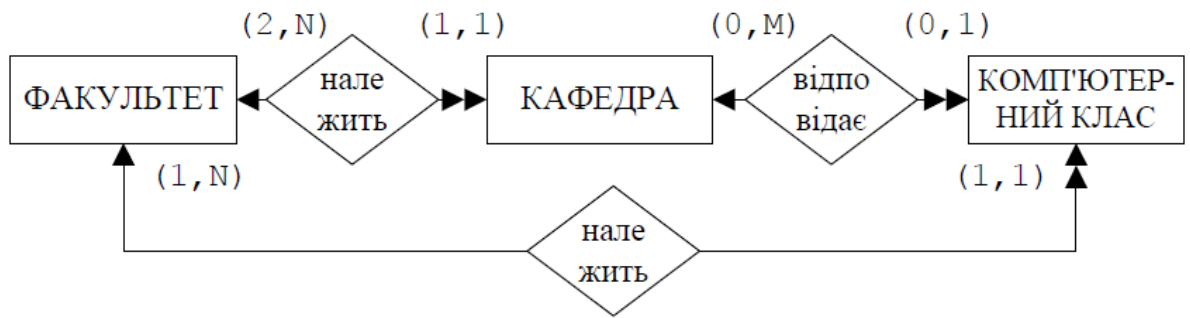


Рис. 20. Усунення дефекту розриву в прикладі ER-моделі

5. Приклад побудови моделі "сутність – зв'язок".

Процес концептуального проектування БД є ітеративний й заснований на операціях і процедурах, що повторюються. Спочатку створюється базова ER-модель певної предметної області. При дослідженні цієї моделі як правило з'являться додаткові сутності, атрибути і зв'язки. Після цього ER-модель буде змінюватися. Процес змін повторюється до тих пір, поки концептуальна модель не буде відображати предметну область.

Розробники БД на основі опитування фахівців визначають сутності, атрибути, зв'язки у предметній області, що розглядається, дослідженні документів, які застосовуються в роботі на підприємстві.

Розглянемо приклад створення ER-моделі предметної області ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД (ВНЗ).

Задачі інформаційної системи.

Мета створення бази даних полягає у такому:

- забезпечення адміністрації університету довідковою інформацією по факультетах, кафедрах, спеціальностях, викладачах, студентах і дисциплінах, що викладаються у ВНЗ;
- контроль успішності студентів.

Аналіз предметної області.

У результаті дослідження й аналізу предметної області були визначені такі сутності, атрибути і первинні ключі (табл. 2).

Зв'язки сутностей визначаються на основі бізнес-правил, які побудовані з урахуванням організаційної структури та операцій, що виконуються в системі:

- в університеті існує декілька факультетів;

- на факультеті навчаються групи студентів за певними спеціальностями;
- у склад групи входять студенти;
- факультет об'єднує декілька кафедр;
- на кожній кафедрі працює декілька викладачів;
- на кожній спеціальності викладається ряд дисциплін, які проводять викладачі з різних кафедр;
- з кожної дисципліни своєї спеціальності студенти складають іспит або залік;
- не кожен викладач читає дисципліни (наприклад, асистент) і не з кожної дисципліни є викладач (наприклад, нова дисципліна, по якій викладача ще не призначено).

Таблиця 2

Сутності, атрибути і первинні ключі предметної області ВНЗ

Сутність	Атрибути	Первинний ключ
ФАКУЛЬТЕТ	<i>Код Назва Декан</i>	<i>Код факультету</i>
СПЕЦІАЛЬНІСТЬ	<i>Код Назва Вимоги</i>	<i>Код спеціальності</i>
ГРУПА	<i>Код Назва Кількість студентів Староста</i>	<i>Код групи</i>
СТУДЕНТ	<i>Номер залікової книжки Прізвище Адреса Рік народження</i>	<i>Номер залікової книжки</i>
КАФЕДРА	<i>Код Назва Завідуючий кафедрою Кількість викладачів</i>	<i>Код кафедри</i>
ВИКЛАДАЧ	<i>Табельний номер Прізвище Посада Науковий ступінь</i>	<i>Табельний Номер викладача</i>
ДИСЦИПЛІНА	<i>Код Назва Кількість годин Семестр</i>	<i>Код дисципліни</i>

Для спрощення концептуальної моделі бази даних цілий ряд об'єктів і бізнес-правил ВНЗ залишився нерозглянутим.

Дослідження предметної області виявило, що всі сутності є сильними, а зв'язки між сутностями – неідентифікуючими. Зв'язок між сутностями *Студент* і *Дисципліна* має атрибут *Оцінка*.

Побудова ER-діаграми.

На основі задач, які були поставлені перед інформаційною системою, і на основі аналізу предметної області, побудована ER-діаграма ВНЗ (рис. 21).

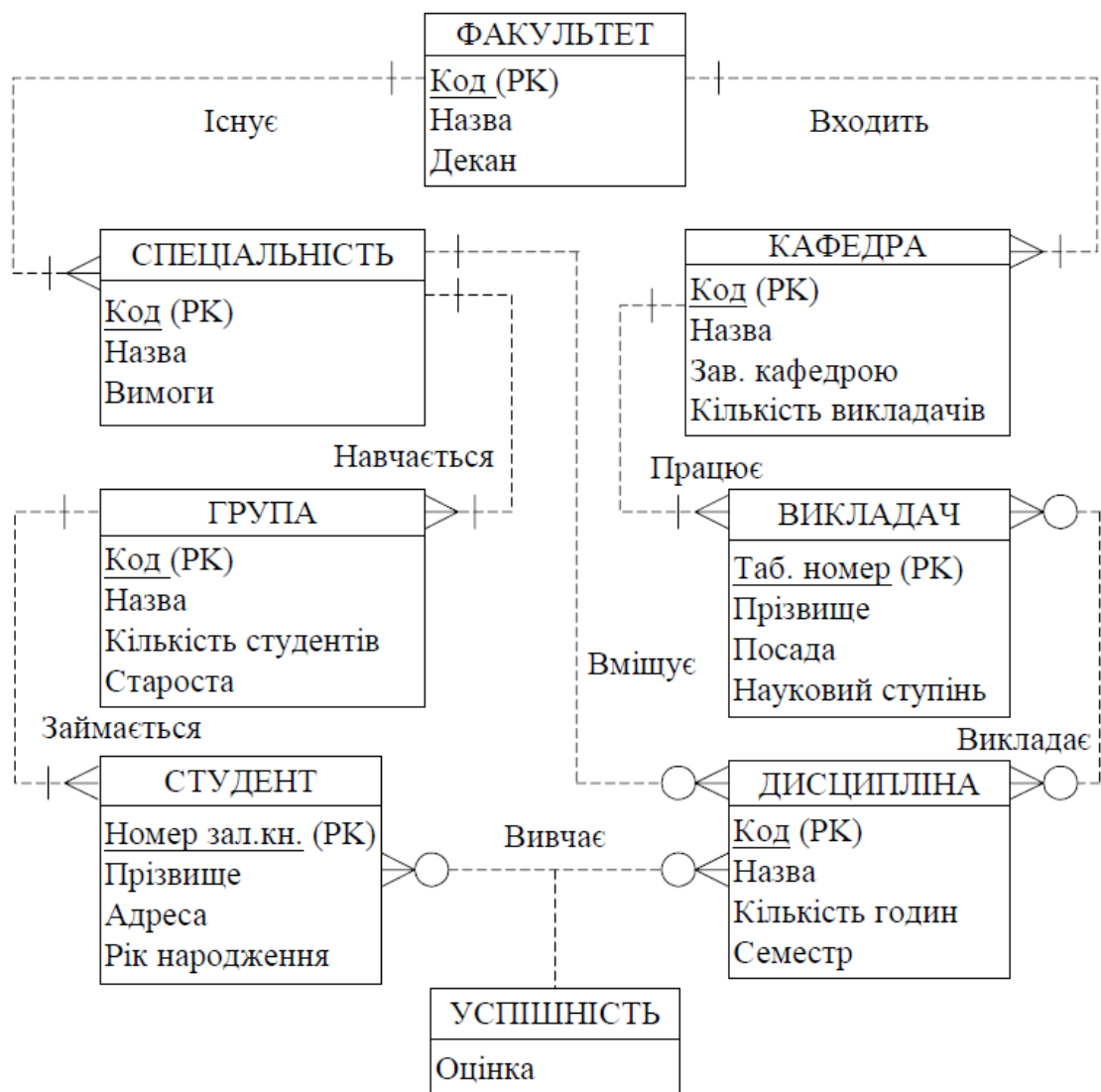


Рис. 21. ER-діаграма предметної області ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД

Розроблений концептуальний проект необхідно перевірити на збитковість та на відповідність транзакціям користувачів.

Перевірка на збитковість передбачає перевірку ER-моделі з метою виявлення збиткових даних і вилучення їх, в тому випадку, якщо вони визначені. Збиткові зв'язки виявляються в тому, що між двома сутностями є декілька шляхів і вони дублюють один одний (це не відноситься до зв'язків, які представляють різні асоціації).

Перевірка моделі на відповідність транзакціям користувачів виконується на основі таких підходів:

- перевірка того, чи представляє модель всю інформацію (сутності, атрибути, зв'язки), яка необхідна для кожної транзакції;
- перевірка по ER-діаграмі маршруту кожної транзакції.

Перевірка моделі на збитковість та на відповідність транзакціям користувачів дозволяє зробити висновок, що концептуальний проект відповідає всім необхідним вимогам.

Слід звернути увагу на те, що розроблений концептуальний проект не є єдиним проектом, який відповідає поставленій задачі. Можливі варіанти розробки системи із застосуванням інших зв'язків між сутностями, або із застосуванням розширеної ER-моделі.

Застосування ER-діаграм дозволяє забезпечити просте і наглядне уявлення про головні логічні об'єкти БД і про зв'язки, які між цими об'єктами існують. Також до переваг ER-діаграми слід віднести те, що вони добре інтегрують з реляційною моделлю.

Недоліком ER-моделей є те, що вони мають недостатні можливості для представлення відношень і обмежень, можуть бути складні при наявності багатьох об'єктів, не мають засобів для опису операцій маніпулювання даними.

Контрольні запитання

1. Дати визначення сутності. Що таке сильна сутність, слабка сутність?
2. Дати визначення атрибута. Що таке простий атрибут, складний атрибут, композитний атрибут?
3. Дати визначення ступеня зв'язку. Що таке кардинальне число?
4. Що таке ідентифікаційний і не ідентифікаційний зв'язок?

5. Навести символні позначення, які застосовуються в діаграмах "сутність – зв'язок".
6. Пояснити, що таке підтипи сутностей і навести приклади.
7. Як відображається наслідування на діаграмах "сутність – зв'язок"?
8. Навести приклади зв'язків 1:N для таких різновидів зв'язків: необов'язково-необов'язково, необов'язково-обов'язково, обов'язково-необов'язково, обов'язково-обов'язково.
9. Дати визначення рекурсивного зв'язку і навести приклади рекурсивних зв'язків 1:1, 1:N, M:N.
10. Пояснити переваги і недоліки ER-моделі.

Розробив: к.т.н., доц. Полотай О.І.