

## Лекція 8

# СТРУКТУРНИЙ АНАЛІЗ ПОТОКІВ ДАНИХ (DFD – DATA FLOW DIAGRAMS)

### 8.1 Призначення діаграм потоків даних

Так само, як і діаграми IDEF0, діаграми потоків даних моделюють систему як набір дій, з'єднаних одна з одною стрілками. Діаграми потоків даних також можуть містити два нових типи об'єктів: об'єкти, що збирають і зберігають інформацію — *сховища даних* і *зовнішні сутності* — об'єкти, які моделюють взаємодію з тими частинами системи (або іншими системами), які виходять за межі моделювання. На рис. 8.1 наведено зовнішній вигляд діаграми потоків даних.

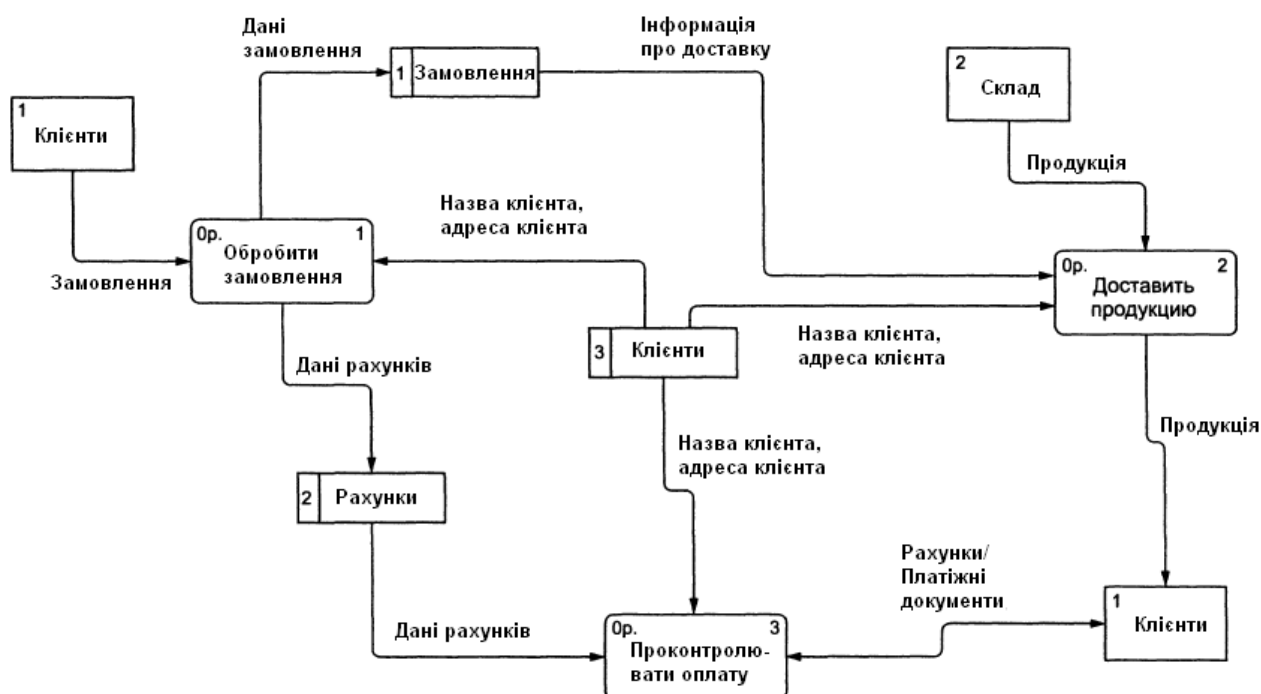


Рисунок 8.1 – Приклад діаграми DFD

На відміну від стрілок в IDEF0, які ілюструють відношення, стрілки в DFD показують, як об'єкти (включаючи й дані) реально переміщуються від однієї дії до іншої. Це подання потоку разом зі сховищами даних і зовнішніми сутностями забезпечує відбиття в DFD-моделях таких фізичних характеристик системи, як *рух* об'єктів (потоки даних), *зберігання* об'єктів (сховища даних), *джерела* і *споживачі* об'єктів (зовнішні сутності).

Побудова DFD-діаграм в основному асоціюється з розробкою програмного забезпечення, оскільки нотація DFD спочатку була розроблена для цієї мети. Зокрема, графічне зображення об'єктів на DFD-діаграмах відповідає прийнятому

Крісом Гейном (Chris Gane) і Трішем Сарсоном (Trish Sarson), авторами DFD-методу, відомого як метод Гейна - Сарсона. Іншою розповсюдженою нотацією DFD є так званий метод Йордана - Де Марко (Yourdon - DeMarco).

## 8.2 Синтаксис і семантика діаграм потоків даних

На відміну від IDEF0, що розглядає систему як множину дій, що взаємно перетинаються, у назвах об'єктів DFD-діаграм переважають іменники. Контекстна DFD-діаграма часто складається з одного функціонального блоку і декількох зовнішніх сутностей. Функціональний блок на цій діаграмі зазвичай має ім'я, що збігається з ім'ям всієї системи (рис. 8.2).

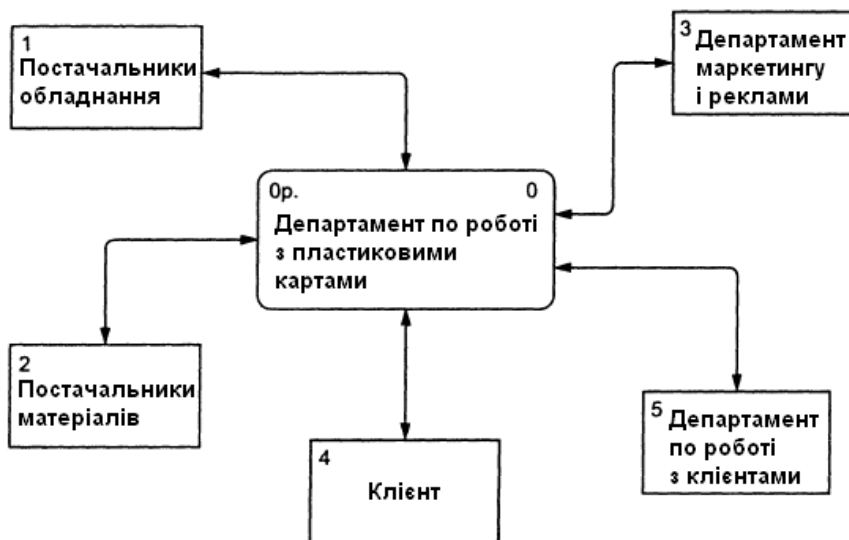


Рисунок 8.2 – Контекстна діаграма DFD

Додавання на діаграму зовнішніх посилань не змінює фундаментальної вимоги, що модель повинна будуватися з єдиної точки зору і повинна мати чітко визначені мету і границі, що вже обговорювалося раніше.

### 8.2.1 Функціональні блоки

Функціональний блок DFD моделює деяку функцію, що перетворює якусь сировину в якусь продукцію (або, у термінах IDEF, вхід у вихід). Хоча функціональні блоки DFD і зображуються у вигляді прямокутників із закругленими кутами, вони майже ідентичні функціональним блокам IDEF0 і діям IDEF3. Як і дії IDEF3, функціональні блоки DFD мають входи й виходи, але не мають керування і механізму виконання як IDEF0. У деяких інтерпретаціях нотації DFD Гейна - Сарсона механізми виконання IDEF0 моделюються як *ресурси* і зображуються в нижній частині прямокутника (рис. 8.3).



Рисунок 8.3 – Елемент DFD-діаграми

### 8.2.2 Зовнішні сутності

Зовнішні сутності забезпечують необхідні входи для системи і/або є приймачами для її виходів. Одна зовнішня сутність може одночасно надавати входи (функціонуючи як постачальник) і приймати виходи (функціонуючи як одержувач). Зовнішні сутності зображуються як прямокутники (рис. 8.4) і зазвичай розміщуються біля країв діаграми. Одна зовнішня сутність може бути розміщена на одній і тій же діаграмі в декількох екземплярах. Цей прийом корисно застосовувати для скорочення кількості ліній, що з'єднують об'єкти на діаграмі.



Рисунок 8.4 – Позначення зовнішньої сутності

### 8.2.3 Стрілки (потоки даних)

Стрілки описують пересування (потік) об'єктів від однієї частини системи до іншої. Оскільки всі сторони прямокутника, що позначає функціональний блок DFD, рівнозначні (на відміну від IDEF0), стрілки можуть починатися і закінчуватися у будь-якій частині блоку. В DFD також використовуються двонапрямлені стрілки, які потрібні для відображення взаємодії між блоками (наприклад, діалогу типу наказ – результат виконання). На рис. 8.5 двонапрямлена стрілка позначає взаємний обмін інформацією між департаментами маркетингу і реклами та пластикових карт.

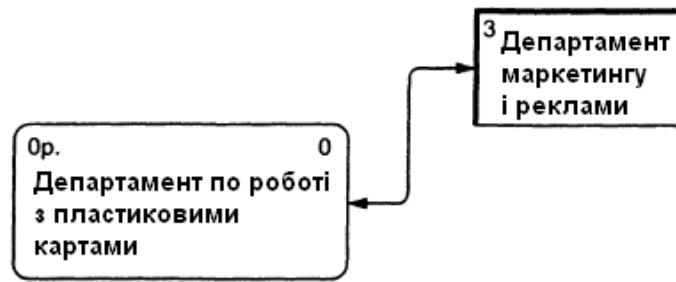


Рисунок 8.5 – Двонаправлений потік між блоком і зовнішньою сутністю

### 8.2.4 Сховища даних

У той час як потоки даних представляють об'єкти в процесі їхнього руху, сховища даних моделюють їх у всіх інших станах. При моделюванні виробничих систем сховищами даних служать місця тимчасового складування, де зберігається продукція на проміжних стадіях обробки. В інформаційних системах сховища даних представляють будь-який механізм, що підтримує зберігання даних для їхньої проміжної обробки. На рис. 8.6 наведено приклад позначення сховищ даних на DFD-діаграмах.



Рисунок 8.6 – Позначення сховища даних на DFD-діаграмі

### 8.2.5 Розгалуження і об'єднання

Стрілки на DFD-діаграмах можуть бути розбиті (розгалужені) на частини, і при цьому кожний отриманий сегмент може бути перейменований таким чином, щоб показати декомпозицію даних, які переносяться даним потоком (рис. 8.7).



Рисунок 8.7 – Розгалуження стрілки, що ілюструє декомпозицію даних

Стрілки можуть і з'єднуватися між собою для формування так званих комплексних об'єктів. Приклад такого об'єднання наведений на рис. 8.8.

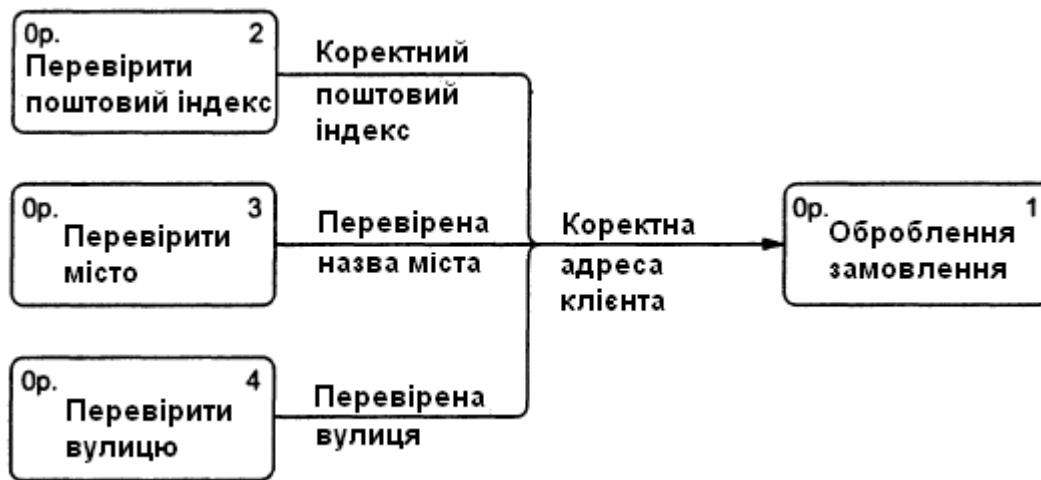


Рисунок 8.8 – Об'єднання потоку в один

### 8.3 Побудова діаграм потоків даних

#### 8.3.1 Два підходи до побудови DFD-моделей

Діаграми DFD можна будувати з використанням підходу, аналогічного структурному методу аналізу і проектування, застосовуваному в IDEF0. Спочатку будується модель фізичної реалізації реальної системи, що використовується користувачами в цей час. Потім створюється логічна модель поточного стану системи для моделювання основних вимог існуючої системи. Після цього створюється нова логічна модель для відбиття основних параметрів пропонованої розроблювальної системи. Нарешті, створюється нова фізична модель, що реалізує логічну модель нової системи.

На сьогодні при розробці інформаційних систем завойовує все більшу популярність альтернативний підхід, відомий як *розподіл подій*, у якому для моделювання системи будується кілька моделей DFD. Спочатку будується логічна модель, яка відображає систему як набір дій і описує, що повинна робити система.

Потім будується *модель оточення*, що описує систему як об'єкт, що відповідає на події, породжувані зовнішніми сутностями. Така модель зазвичай складається з опису призначення системи, однієї діаграми контекстного рівня і списку подій. Контекстна діаграма містить один функціональний блок, що представляє систему в цілому, і зовнішніх сутностей (оточення), з якими система взаємодіє.

На заключному етапі створюється *модель поведінки*, що показує, як система обробляє ті або інші події. Ця модель починається з єдиної діаграми з одним функціональним блоком на кожен відповідь системи на подію, описану в моделі оточення. Сховища даних у моделі поведінки використовуються для моделювання даних, які повинні зберігатися в проміжках між обробкою подій. Потoki застосовуються для з'єднання елементів діаграм між собою і для перевірки узгодженості моделей поведінки і оточення.

При підготовці такого роду моделей до різних презентацій зазвичай необхідна їх "чистка". При цьому може застосовуватися як створення спрощених батьківських діаграм за допомогою об'єднання декількох функціональних блоків в один, так і декомпозиція деяких елементів для більш ясного сприйняття моделі.

### 8.3.2 Нумерація об'єктів

В DFD кожний номер функціонального блоку може містити в собі префікс, номер батьківської діаграми і власне номер об'єкта (рис. 8.9). Номер об'єкта унікальним образом ідентифікує функціональний блок на діаграмі. Номер батьківської діаграми і номер об'єкта в сукупності забезпечують унікальну ідентифікацію кожного блоку моделі.

Унікальні номери присвоюються також кожному сховищу даних і кожній зовнішній сутності незалежно від розташування об'єкта на діаграмі. Кожний номер сховища даних містить префікс D (від англійського Data Store) і унікальний номер сховища в моделі (наприклад, D3).



Рисунок 8.9 – Компоненти номера функціонального блоку DFD

Аналогічно кожний номер кожної зовнішньої сутності містить префікс E (від англійського External entity) і унікальний номер сутності в моделі (наприклад, E5).

Отже, діаграми потоків даних (DFD) забезпечують зручний спосіб опису переданої інформації як між частинами системи, яка моделюється, так і між системою та зовнішнім світом. Ця якість визначає область застосування DFD – вони використовуються для створення моделей інформаційного обміну організації, наприклад моделі документообігу. Крім того, різні варіації DFD широко застосовуються при побудові корпоративних інформаційних систем.