



# **Проектування на основі пакетів**



# Проектування на основі пакетів

## Зміст

1. Проектування на основі пакетів
2. Рекомендації по організації пакетів
3. Формування пакетів на базі груп нестійких класів
4. Базові пакети повинні бути стійкими
5. Факторизація незалежних типів



# Проектування на основі пакетів

Якщо деякий пакет  $X$  в значній мірі залежить від конкретної групи розробників, то його необхідно зробити максимально стійким (не підлягає зміні в нових версіях системи), оскільки в іншому випадку підвищується залежність від цих розробників.

Це досить очевидно, але іноді даному питанню не приділяється достатньо уваги, що приводить до високих витрат.

В процесі побудови діаграм розробник може розмістити типи даних і пакети достатньо довільно. Проте в процесі проектування фізичної архітектури спосіб розбиття системи на пакети визначається взаємним впливом елементів системи.



# Проектування на основі пакетів

## Зміст

1. Проектування на основі пакетів
- 2. Рекомендації по організації пакетів**
3. Формування пакетів на базі груп нестійких класів
4. Базові пакети повинні бути стійкими
5. Факторизація незалежних типів



# Рекомендації по організації пакетів

Основний інтуїтивний підхід до розбиття системи на модулі має на увазі групування елементів за принципом високого зацеплення – тісного взаємозв'язку частин пакета в рамках реалізації загальних задач, служб, політик і функцій.

Також показником групування може бути відносно висока ступінь внутрішнього зв'язування, яка визначає належність типів деякому кластеру.

Сімейство функціонально зв'язаних інтерфейсів можна розмістити в окремий пакет, який не містить реалізацію класів. Не мається на увазі пара зв'язаних один з одним інтерфейсів, а мається на увазі сімейство, яке включає не менше трьох інтерфейсів.



# Проектування на основі пакетів

## Зміст

1. Проектування на основі пакетів
2. Рекомендації по організації пакетів
- 3. Формування пакетів на базі груп нестійких класів**
4. Базові пакети повинні бути стійкими
5. Факторизація незалежних типів



# Формування пакетів на базі груп нестійких класів

Пакет – це базова структурна одиниця розробки програмного продукту. Припустимо, існує великий пакет Р1, який містить приблизно 30 класів, деяка підмножина якого (приблизно 10 класів) постійно модифікується. В цьому випадку варто розділити пакет Р1 на два окремих пакети Р1-а і Р1-б, включив в Р1-б десять нестійких класів.

Таким чином пакет ділиться на стійку і нестійку частини чи на групи класів, робота над якими ведеться одночасно.

Отже, якщо робота над класами пакету ведеться одночасно, значить, пакет згруповано правильно.

Таке перегрупування є наслідком врахування трудомісткості створення пакету. На ранніх ітераціях дуже складно сформувати добру структуру пакетів застосування. Вона постійно змінюється на стадії розвитку і по закінченню стадії розробки повинна бути сформована стійка структура більшості пакетів.

Основна стратегія – зниження залежності від нестійких пакетів.



# Проектування на основі пакетів

## Зміст

1. Проектування на основі пакетів
2. Рекомендації по організації пакетів
3. Формування пакетів на базі груп нестійких класів
- 4. Базові пакети повинні бути стійкими**
5. Факторизація незалежних типів





# Базові пакети повинні бути стійкими

Якщо структура базових пакетів (від яких залежить більшість інших) нестійка, то підвищується ймовірність помилок в застосуванні.

Існує декілька способів підвищення стійкості пакетів:

Включення до пакету переважно інтерфейсів і абстрактних класів.

Відсутність залежності від інших пакетів чи залежність тільки від дуже стійких пакетів, або інкапсуляція залежності всередині пакету.

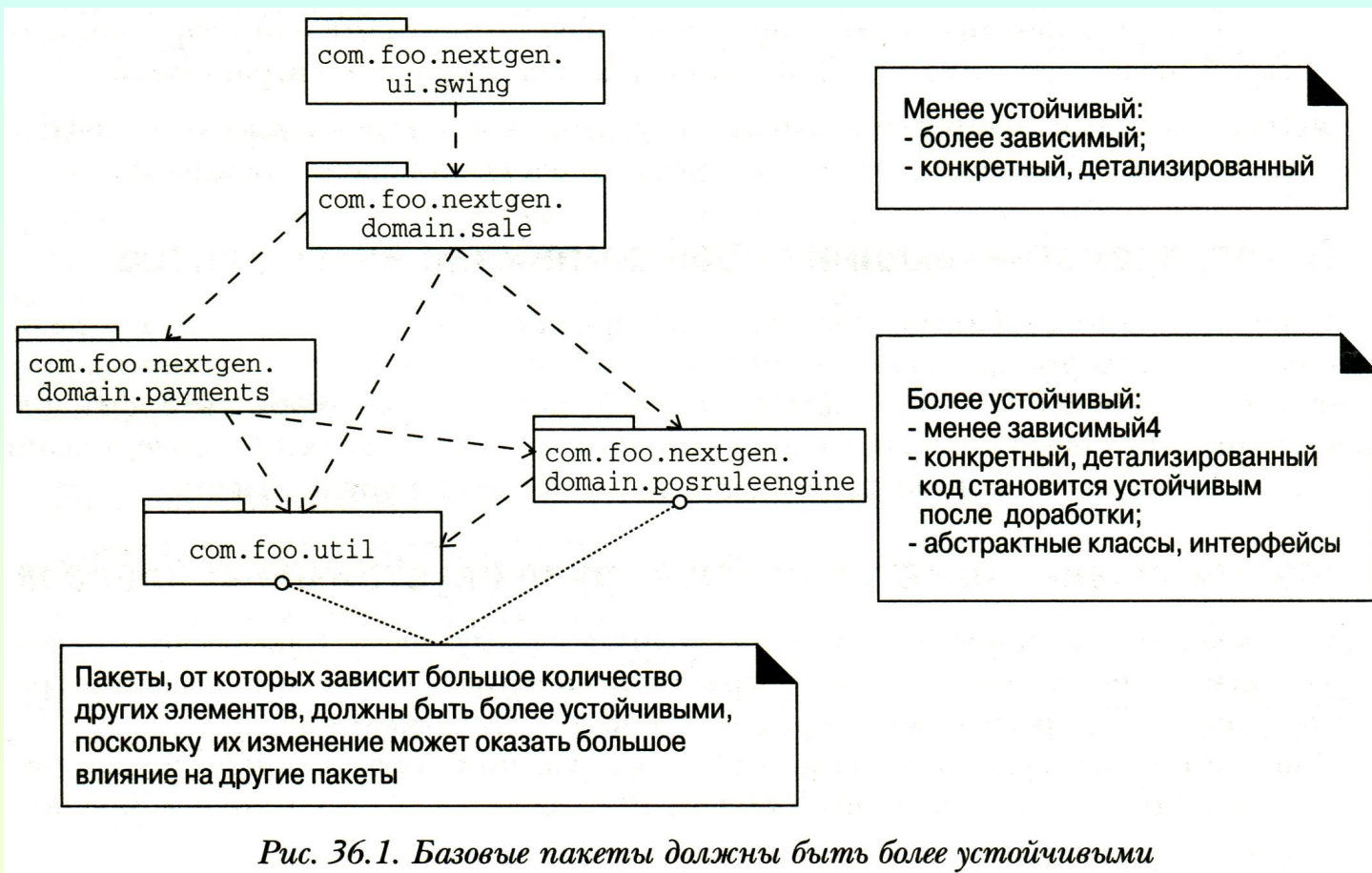
Наприклад в пакеті можна приховати реалізацію правил за “фасадним” об’єктом. Тоді навіть при зміні реалізації правил залежні пакети не будуть здійснювати вплив один на одного.

Включення до пакету відносно стійкого коду випробуваного раніше.

Затвердження стабільного графіку розробки, який передбачає нечасте внесення змін.



# Стійкість базових пакетів





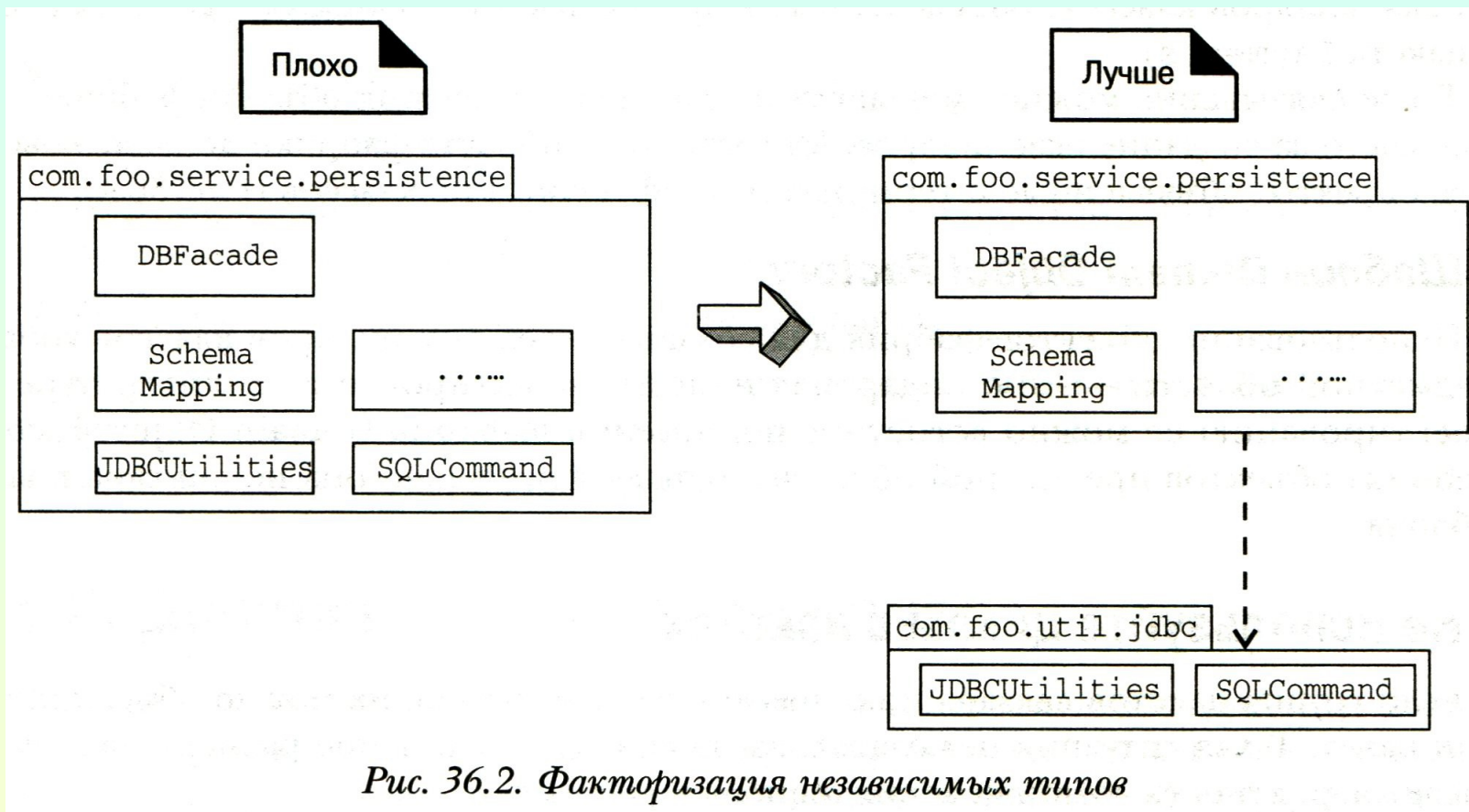
# Проектування на основі пакетів

## Зміст

1. Проектування на основі пакетів
2. Рекомендації по організації пакетів
3. Формування пакетів на базі груп нестійких класів
4. Базові пакети повинні бути стійкими
- 5. Факторизація незалежних типів**



# Факторизація незалежних типів



Класи, які використовуються незалежно один від одного чи в різному контексті, слід розміщати до різних пакетів.



# **Проектування на основі пакетів**

**Дякую за увагу**