

# Принципи проектування

# Шаблони розробки ПЗ

# Шаблони розробки ПЗ

- У розробці ПЗ часто зустрічаються проблеми, які вже розв'язувалися раніше в інших проектах.
- У зв'язку з тим, що контексти, в яких дана проблема розв'язувалась, можуть різнитися
  - (інший тип додатку, інша платформа або інша мова програмування),
  - все зазвичай закінчується повторенням проектування і реалізації даного розв'язку,
  - тим самим виникає ситуація «повторного винаходу колеса».

# Стиль мислення експерта

- Під час розв'язку конкретних проблем **експерти** звичайно не намагаються розробити нове рішення, яке відрізняється від уже існуючих.
- Дії експерта:
  - згадують аналогічну проблему, яку вони вже розв'язували,
  - стараються повторно використати суть раніше прийнятого рішення для рішення нової проблеми.
- Такий «стиль мислення» в термінах пар «проблема - рішення», є загальним для сукупності різних предметних областей, таких, як:
  - архітектура;
  - економіка;
  - програмна інженерія.

# Навіщо потрібні шаблони

- Шаблони дозволяють базуватися на колективному досвіді кваліфікованих інженерів по проектуванню.
- Вони фіксують існуючий досвід розробки, що добре себе зарекомендував.
- Кожен шаблон має справу з конкретною проблемою, що багато разів зустрічається в області проектування і реалізації.

# Визначення шаблону

- Шаблон – це опис добре перевіреної, узагальненої схеми розв'язку деякої проблеми (задачі), що часто повторюється під час розробки ПЗ, яка виникає в деяких специфічних умовах (контексті).
- Схема розв'язку проблеми задається шляхом
  - визначення використовуваних (складових) компонент;
  - їх відповідальностей;
  - способів їх взаємодії.

# Властивості шаблонів

- 1. Шаблони описують розв'язок** для задач проектування, що часто повторюються, які виникають в деяких специфічних ситуаціях.
- 2. Шаблони документують** накоплений досвід проектування, що добре себе зарекомендував.
- 3. Шаблони визначають і описують абстракції**, які знаходяться на вищому рівні, ніж рівень окремих класів і екземплярів або компонентів.
- 4. Шаблони надають спільний словник термінів і абстрактно-ресумішних принципів**

## Властивості шаблонів (2)

- 5. Шаблони є засобами документування архітектур ПЗ.**
- 6. Шаблони підтримують конструювання ПЗ з певними властивостями.**
- 7. Шаблони допомагають розробляти складні і різнорідні архітектури ПЗ.**
- 8. Шаблони допомагають боротися зі складністю ПЗ.**



# Типи патернів

- В ОО аналізі і проектуванні розроблено багато різних патернів.

## 1. Архітектурні патерни.

- Описують фундаментальні способи структурування програмних систем.
- Ці патерни відносяться до рівня систем і підсистем, а не класів.

## 2. Патерни проектування.

- Описують структуру програмних систем в термінах класів.
- Найбільш відомими в цій області є 23 патерна, описані в [GoF].

## 3. Патерни аналізу.

- Надають загальні схеми організації процесу об'єктно-орієнтованого моделювання.

# Типи шаблонів розробки ПЗ

- На етапі аналізу системи:
  - **шаблони аналізу** (analysis patterns ) – комбінації класів для опису стандартних задач прикладної області;
- На етапі проектування системи
  - **шаблони архітектури** (architecture patterns).
  - **шаблони проектування** (design patterns);
  - специфічні для конкретної мови програмування **ідіоми**.

# Шаблони архітектури ПЗ

- Архітектурні шаблони описують базові схеми структурної організації програмних систем.
- Архітектурні шаблони
  - надають набір наперед визначених підсистем,
  - визначають їх відповідальності
  - включають правила і рекомендації по організації взаємодії між ними.

- Шаблони архітектури ПЗ є шаблонами самого високого рівня в системі шаблонів ПЗ.
- Вони допомагають визначити базову структуру програмної системи.
- Кожна робота по розробці ПЗ управляється її структурою:
  - детальний опис підсистем;
  - комунікація і взаємодія між різними частинами системи;
  - їх наступне розширення.
- Кожен архітектурний шаблон допомагає розробнику досягти деякої глобальної властивості системи, що розробляється.
  - Наприклад, адаптованість інтерфейса.

# Види архітектурних шаблонів

1. Layers (рівні),
2. Pipes and Filters (канали і фільтри),
3. Blackboard (інформаційна "дошка"),
4. Broker (брокер),
5. Model-View-Controller (Модель-Представлення-Контролер),
6. Presentation-Abstraction-Control (Представлення-Абстракція-Контролер),
7. Microkernel (мікроядро),
8. Reflection (відображення).

# **Шаблони (патерни) проектування**

# Шаблони проектування

- Шаблони проектування це шаблони середнього рівня.
- Вони менше за масштабом, ніж шаблони архітектури, але знаходяться на вищому рівні, ніж специфічні для мов програмування ідіоми.
- Застосування шаблонів проектування не впливає на базову структуру ПС, але може сильно вплинути на архітектуру підсистем.

Як задачі проектування  
розв'язуються за допомогою патернів

- Патерни проектування дозволяють різними способами розв'язувати багато задач, з якими постійно зіштовхуються проектувальники ОО додатків.
- Задачі, що найчастіше зустрічаються :
  - *Пошук підходящих об'єктів*
  - *Визначення ступеня деталізації об'єкта*
  - *Опис інтерфейсів об'єкта*



# Класифікація патернів проектування

<b>Рівень</b>	<b>Мета</b>	<b>Породжуючі патерни</b>	<b>Структурні патерни</b>	<b>Патерни поведінки</b>
<b>Клас</b>		<b>Factory Method</b>	<b>Adapter (класу)</b>	<b>Interpreter Template Method</b>
<b>Об'єкт</b>		<b>Abstract Factory Singleton Prototype Builder</b>	<b>Adapter (об'єкту) Decorator Proxy Composite Bridge Flyweight Facade</b>	<b>Iterator Command Observer Visitor Mediator State Strategy Memento Chain of Responsibility</b>

