

Object, Class, and Profile Diagrams. Class Stereotypes and Associations.

Резюме

Обект: Представява реален или концептуален елемент с **идентичност, състояние и поведение**. **Идентичността** прави всеки обект уникален, **състоянието** - свойствата на обекта в даден момент, а **поведението** - как обектът реагира на заявки от др.обекти.

Клас: Описание на множество обекти с общи свойства (атрибути), поведение (операции), отговорности, връзки и семантика. Класът е шаблон за създаване на обекти.

Стереотипи: Разширяват модела: «entity», «boundary», «control».

Асоциации: двупосочни връзки между обекти от различни класове. Включват **роля и множественост**, като например 1..1 (точно един), 0..1 (нула или един), 1..* (един или повече).

Обектно-ориентиран анализ (ООА): Процес на дефиниране на **проблема** чрез обекти, включващ реални обекти, с които системата трябва да взаимодейства, и кандидат софтуерни обекти за изследване на различни решения.

Обектно-ориентиран дизайн (OOD): Процес на дефиниране на **решението** чрез компоненти, интерфейси, обекти, класове, атрибути и операции, които ще удовлетворят изискванията. Два вида: **архитектурен** (дефиниране на компонентите) и **компонентен** (дефиниране на класовете и интерфейсите в рамките на компонент).

Диаграми на класове: Показват колекция от декларативни (статични) елементи като класове, пакети и техните връзки.

Гранични класове- взаимодействие м/у системата и нейната среда; напр. потребителски интерфейси

Контролни класове- контролно поведение за един/повече use cases.

Entity класове- информация и свързаното с нея поведение, което трябва да се съхранява (например данни за събития, лица и т.н.).

Abstraction - показване същественото, скриване несъщественото

Delegation - reusing classes

Encapsulation - затваряне на всички части на абстракцията в **клас**;

Information Hiding - скриване на части от абстракцията в **обект**.

Наследяване: Различават се два основни класа наследяване - **Sub-typing** (логическа класификация) и **Implementation inheritance** (използване на имплементацията на един клас за удобство в друг).

Class-Responsibility-Collaborations (CRC) card - класове в презентация като карти: описва класа, атрибути му и отговорности

Диаграми на обекти- моментна снимка на инстанции на класове в опр. момент, помагайки за изясняване на класовете и наследяването.

Профилните диаграми са структурни диаграми, предоставящи общ механизъм за разширяване и персонализиране на UML-модели за конкретни домейни или платформи. Вкл.: **stereotypes**; **tagged value definitions**- keyword-value pairs of attributes; **constraints** - as a string enclosed in brackets near associated element; OCL boolean expression

visibility ::= { + | - | # | ~ }

Derived Attributes- / - attribute may not be strictly necessary

Optional Property Modifiers { id }, { readOnly }, { unique }

Direction (optional) - in, inout, out, or return

Self-Associations - един екземпляр от класа има асоциации към други екземпляри от същия клас