



Programovanie v jazyku C#

Návrhové vzory

prednáška 10
Ing. Ján Magyar, PhD.
ak. rok. 2023/2024 ZS

Návrhový vzor

- všeobecné znovupoužiteľné riešenie pre opakujúce sa problémy v softvérovom inžinierstve
- popis riešenia, nie samotné riešenie
- formalizovaný best practice
- nie sú nevyhnutné, ale zjednodušujú implementáciu

Štruktúra návrhového vzoru

- definuje komponenty a ich rolu
- definuje vzťah medzi komponentmi
- nešpecifikuje funkcionálnosť (závisí od prípadu použitia)
- nešpecifikuje implementáciu (je to na programátorovi)

Použitie návrhových vzorov

- pre časté problémy
- programovací jazyk môže ponúkať implicitné riešenie problému
- zvyčajne pre objektovo-orientované jazyky a programy
- osvedčený spôsob písania kvalitného softvéru

Výhody použitia návrhových vzorov

- rýchlejší vývoj
- bezpečnejšie a overené riešenia
- rieši aj skryté problémy
- lepšia čitateľnosť a štruktúra riešenia
- rozdelenie do komponentov pre znovupoužitie kódu
- zdokumentované riešenie

Dokumentácia návrhových vzorov

- meno a klasifikácia
- cieľ - prečo by sa mal použiť?
- alternatívne mená
- motivácia - ukázkový problém
- použiteľnosť - popis problémov pre ktoré sa vzor dá použiť
- štruktúra - diagram triedy
- členovia - zoznam komponentov

Dokumentácia návrhových vzorov (pokračovanie)

- spolupráca - ako interagujú triedy a objekty
- dôsledky - výsledky, následky, kompromis
- implementácia - ukázkový prípad použitia
- ukázkový kód - ako sa implementuje v niektorom jazyku
- známe prípady použitia - reálne využitia
- podobné vzory - porovnanie s ďalšími vzormi

Kritika návrhových vzorov

- môžu naznačiť chýbajúcu podporu v programovacom jazyku
- môžu byť implementované inými prístupmi
- zvyšuje zložitosť riešenia pri nevhodnom použití

Typy návrhových vzorov

- kreačné - ako vytvoriť objekt?
- štrukturálne - ako realizovať vzťah medzi objektmi?
- behaviorálne - ako môžu komponenty komunikovať?
- konkurenčnosť - pre viacvláknové programy

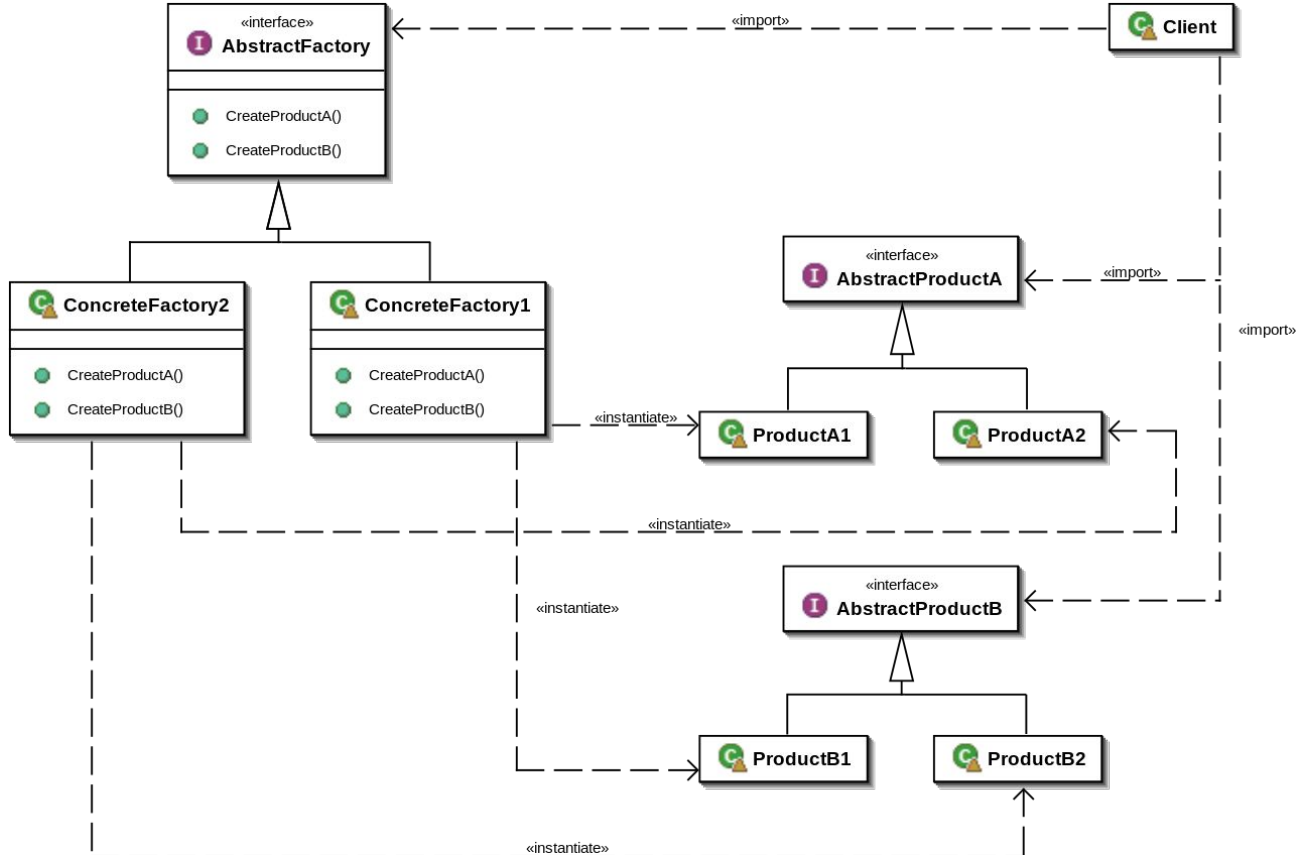
Kreačné návrhové vzory

- abstract factory
- builder
- factory method
- prototype
- singleton

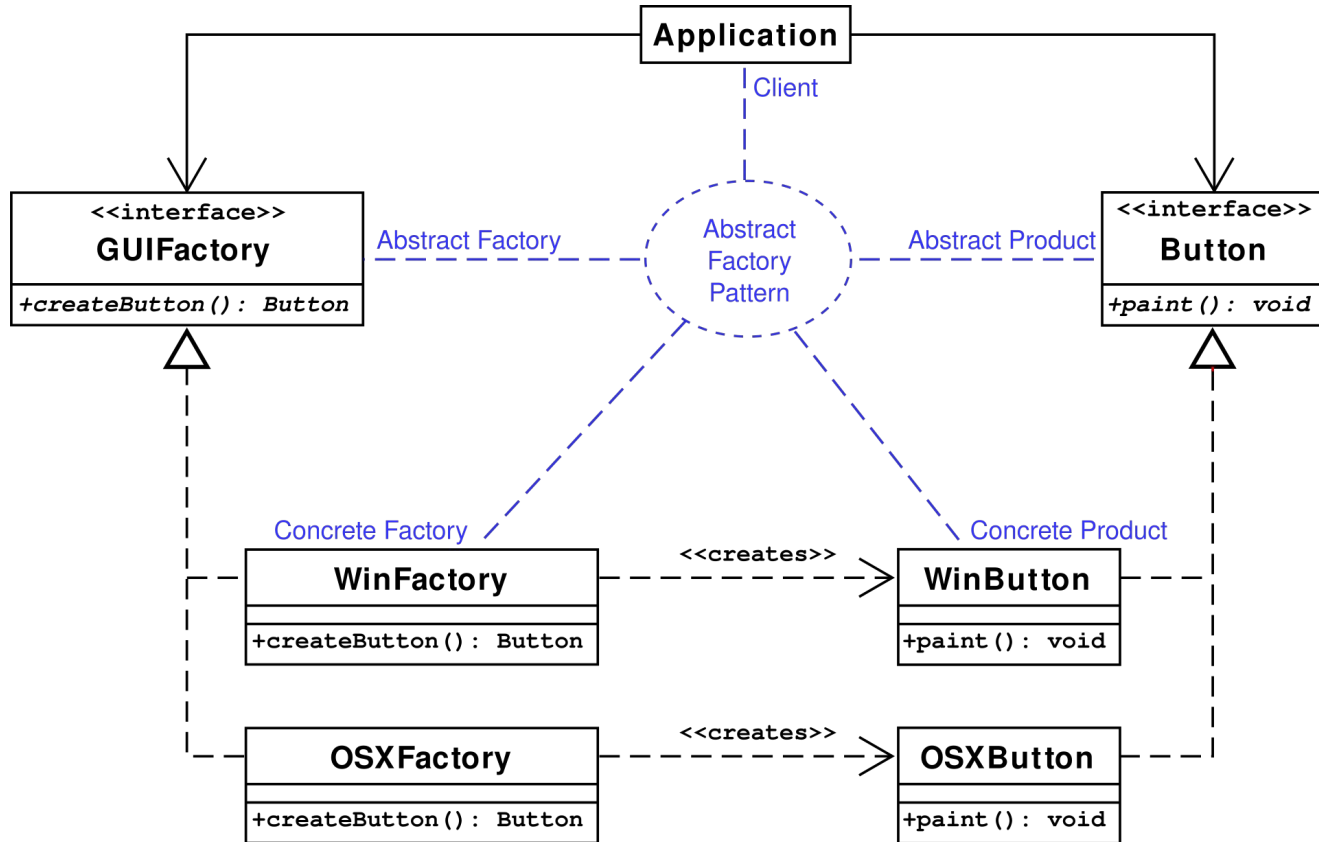
Abstract factory

- poskytnúť rozhranie pre vytvorenie skupiny závislých objektov bez špecifikácie konkrétnej triedy
- Ako môže byť aplikácia nezávislá od toho, ako sa vytvoria jej objekty, a objekty, ktoré potrebuje?
- vytvorenie objektov je skryté v osobitnom objekte
- úloha vytvorenia objektov je delegovaná factory objektu

Abstract factory



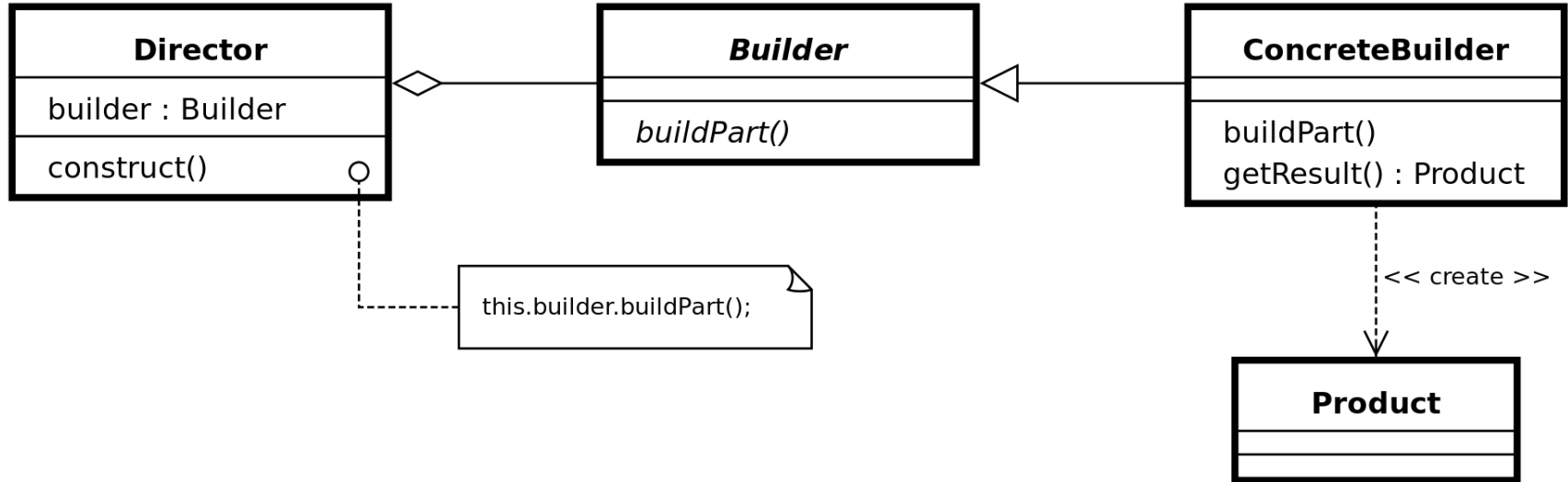
Abstract factory



Builder

- pre zložité objekty, oddelíme ich reprezentáciu od ich vytvorenia
- rovnaký proces môže vytvoriť rôzne reprezentácie
- Ako môžeme zjednodušiť triedu, ktorá obsahuje vytvorenie zložitého objektu?

Builder



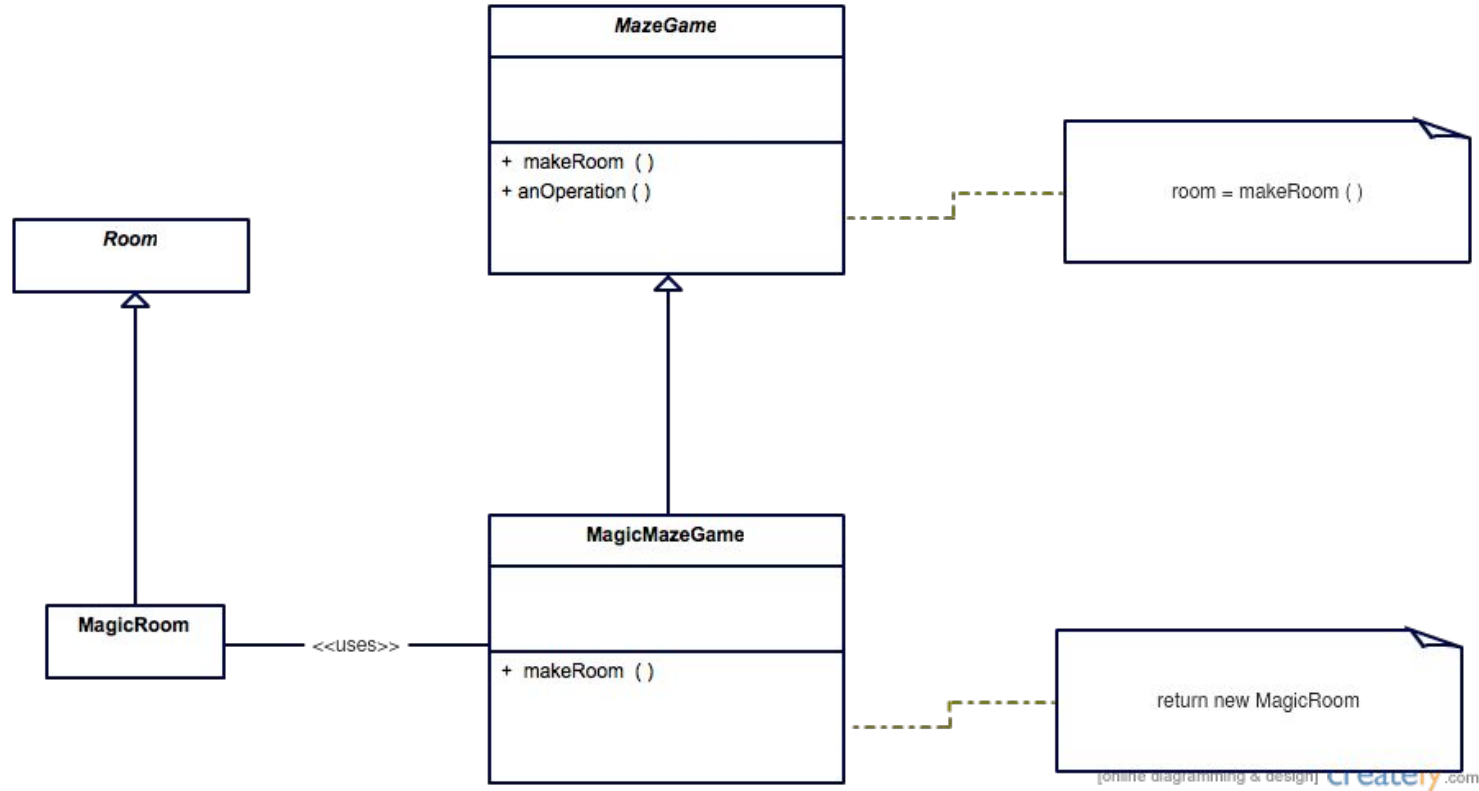
Použitie vzoru Builder

- môžeme meniť vnútornú reprezentáciu objektu
 - enkapsulovaný kód pre vytvorenie a reprezentáciu
 - môžeme kontrolovať proces vytvorenia
-
- musíme zdefinovať builder pre každý typ produktu
 - triedy buildera môžeme meniť
 - ťažší dependency injection

Factory method

- rozhranie pre vytvorenie jediného objektu, ale podtriedy rozhodujú, inštanciu ktorej triedy majú vytvoriť
- vytvorenie inšancií je úlohou podtried
- osobitná operácia (factory method) je zodpovedná za vytvorenie objektu, objekt vytvoríme zavolaním tejto metódy

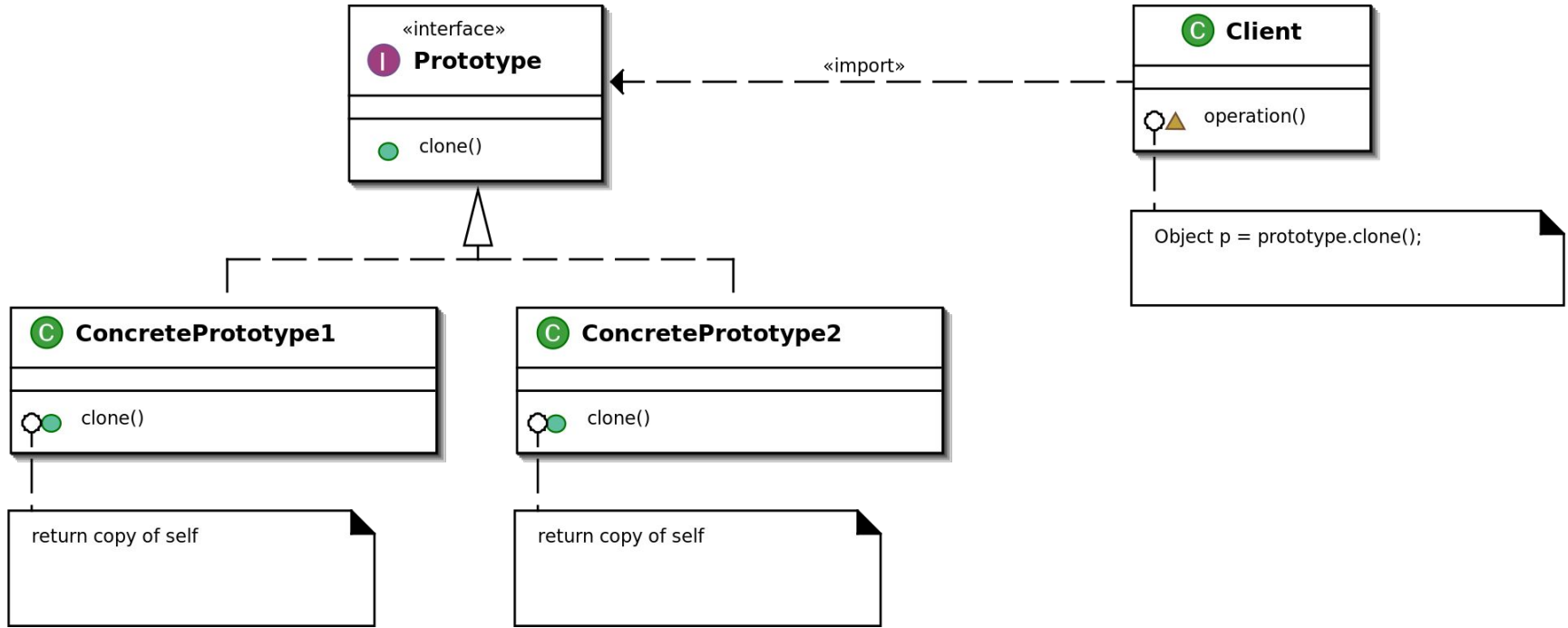
Factory method



Prototype

- objekty vytvárame na základe prototypovej inštancie
- vytváranie inštancií použitím už existujúceho objektu
- lepší výkon, menšia záťaž na pamäť
- môžeme špecifikovať počas behu, ktorý objekt sa má vytvoriť - dynamicky načítané triedy
- definujeme abstraktnú triedu s metódou `clone()`, ktorú implementujú konkrétne podtriedy

Prototype



Singleton

- zabezpečuje, že trieda má iba jednu inštanciu (alebo žiadnu)
- poskytuje prístup k jedinej inštancii
- môžeme kontrolovať vytvorenie inštalácie
- možné lazy initialization
- globálny stav

Singleton

Singleton

- singleton : Singleton
- Singleton()
- + getInstance() : Singleton

Ďalšie kreačné návrhové vzory

- dependency injection
- lazy initialization
- multiton
- object pool
- resource acquisition is initialization

otázky?