



# State

Stav

Umožňuje objektu zmeniť svoje správanie keď sa zmení jeho vnútorný stav. Objekt potom vyzerá, akoby zmenil svoju triedu.

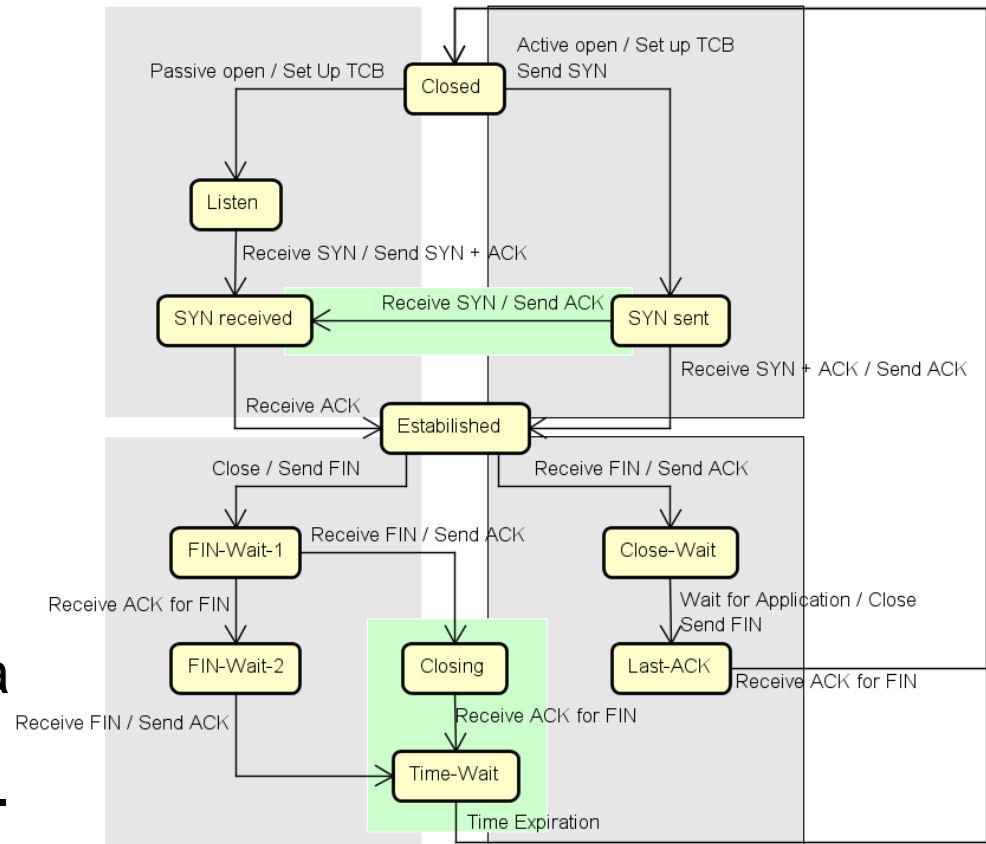
**Allow an object to alter its behavior when its internal state changes. The object will appear to change its class.**

Návrhové vzory



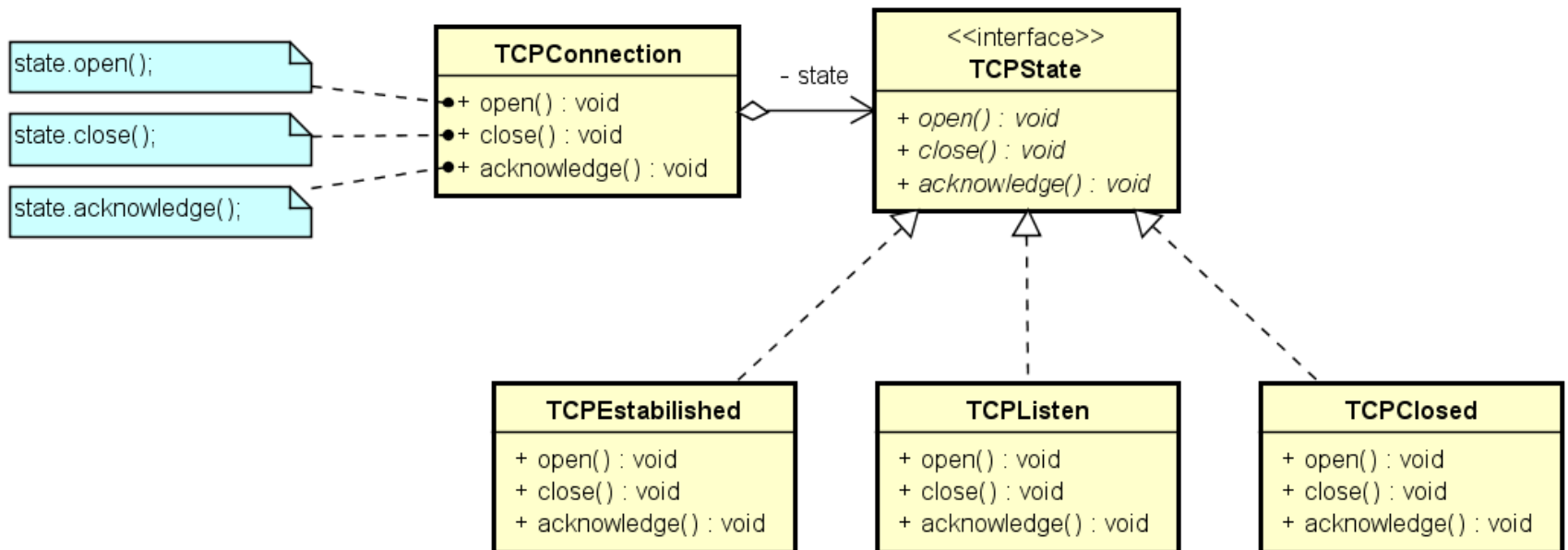
# Motivácia

- Majme TCP protokol, ktorý sa nachádza v rôznych stavoch, čo odlišuje aj jeho správanie.
- Jeden spôsob ako zabezpečiť rôzne reakcie v rôznych stavoch protokolu je „naplniť“ kód podmienkami.
- Vzor Stav toto rieši objektovou cestou – zmena správania sa dosiahne zmenou stavového objektu.



powered by Astah

# Motivácia

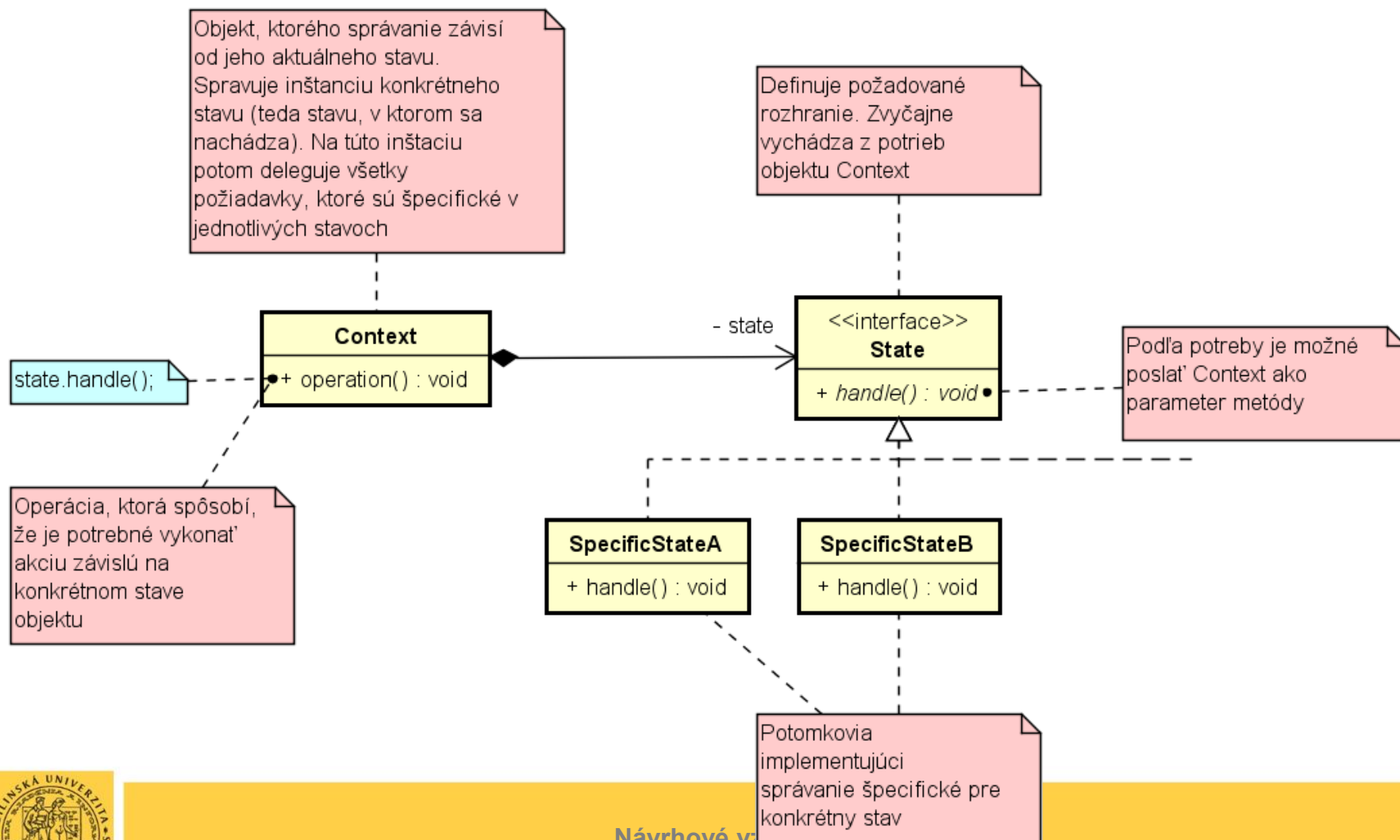


powered by Astah

# Aplikovateľnosť

- Stav je vhodné použiť, keď:
  - je správanie objektu závislé na jeho stave a tento stav sa počas behu programu mení.
  - majú operácie v kóde veľa podmienok/vetvení/vnorených výrazov, ktoré sú závislé od stavu objektu, ktorý býva spravidla reprezentovaný konštantou/vymenovaným typom (`enum`). Návrhový vzor stav v sebe zapúzdri každú takúto vetvu, stavy sa stanú nezávislé. Ďalším benefitom je podstatné sprehľadnenie kódu.

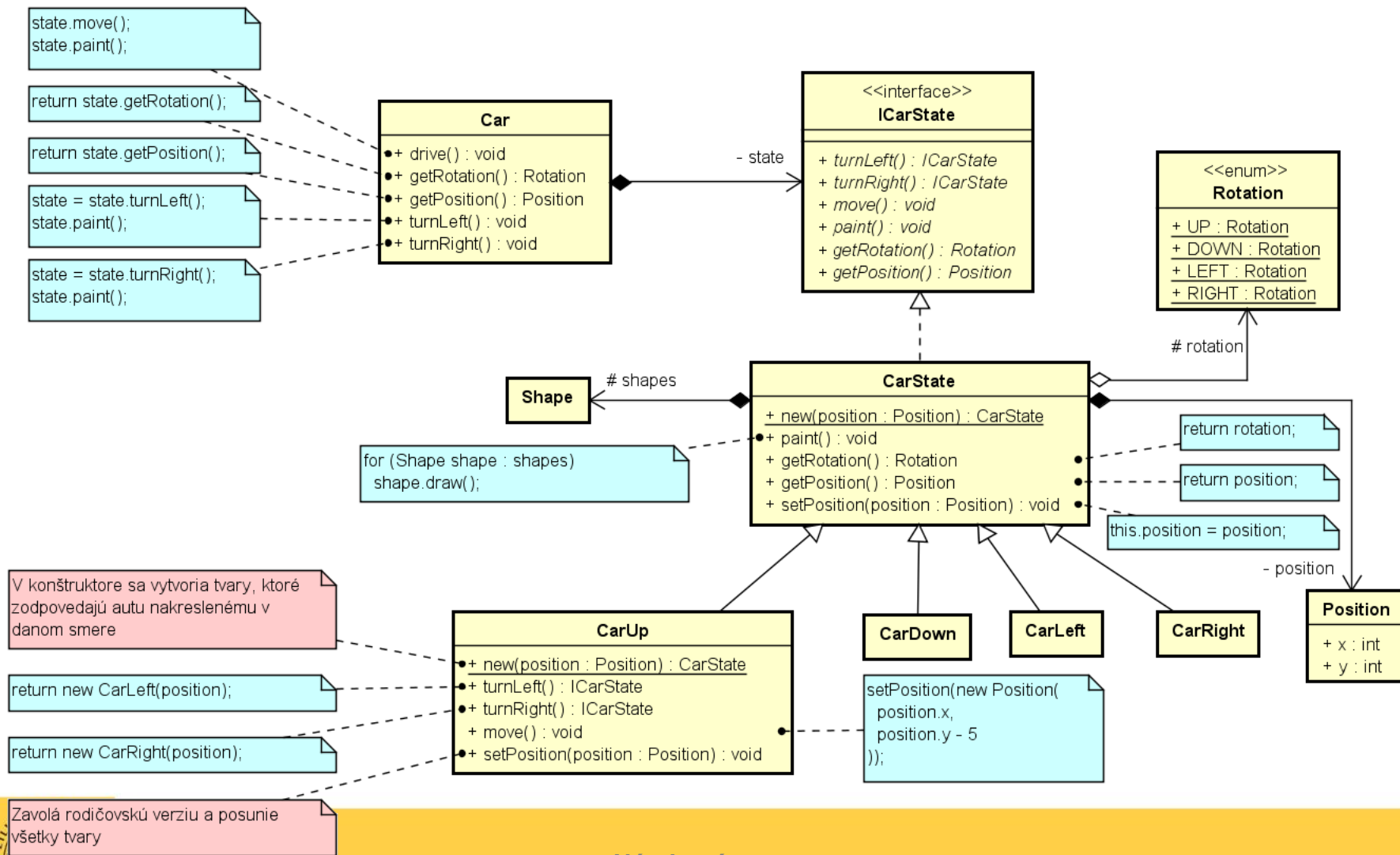
# Implementácia



# Dôsledky

- Centralizuje v sebe správanie špecifické pre jeden konkrétny stav a tým rozdeľuje správanie špecifické pre stavy, ktoré sa od seba líšia. Tým pádom je veľmi jednoduché rozšírenie objektu o ďalší stav, keďže stačí vytvoriť potomka, ktorý definuje správanie v danom stave (pozor, môže sa stať, že takých potomkov bude veľa) a nie je potrebné zanášať do kódu žiadne ďalšie vetvenie.
- Zmenu vnútorného stavu objektu je viac transparentná. Z pohľadu objektu, ktorý mení stav sa často jedná iba o zmenu hodnôt v atribútoch (niekoľko príkazov za sebou), čo niekedy môže spôsobiť nekonzistentný stav (môže sa stať, že sa zabudne priradiť hodnota niektorému atribútu). Návrhový vzor Stav tomuto prechádza, nakoľko z perspektívy objektu, ktorý mení stav, sa vždy jedná o atomickú operáciu, ktorú vykoná objekt Stav.
- Stavy je možné zdieľať (napr. ak nemajú žiadne atribúty) napr. s využitím vzoru Mušia váha.

# Príklad



# Príbuzné vzory

- *Flyweight* – pomocou Mušej váhy je možné zdieľať stavy.
- *Singleton* – stavy sú často Jedináčikovia.



# Upozornenie

- Tieto študijné materiály sú určené výhradne pre študentov predmetu 5I132 Návrhové vzory (Design Patterns) na Fakulte riadenia a informatiky Žilinskej univerzity v Žiline.
- Reprodukovanie, šírenie (i častí) materiálov bez písomného súhlasu autora nie je dovolené.

Ing. Michal Varga, PhD.  
Katedra informatiky  
Fakulta riadenia a informatiky  
Žilinská univerzita v Žiline  
[Michal.Varga@fri.uniza.sk](mailto:Michal.Varga@fri.uniza.sk)

