

**JEŠTĚ K SIMULACI...**

# Co kdybychom místo stěhování písku chtěli modelovat něco jiného?

hlavní cyklus zůstane stejný  
změní se

- reprezentace a inicializace modelu (PisekA, PisekB...)
- jaké **procesy** budou v modelu
- jaké **události** budou v modelu
- **jak** budou jednotlivé procesy **zpracovávat** události  
třeba **AUTO** a událost **ZAČNI\_NAKLÁDAT**:
  - *zjistí kolik můžeš naložit -  $\min(\text{PisekA}, \text{nosnost})$*
  - *pokud je to  $\leq 0$ , tak skonči*
  - *jinak nastav vlastnost **kolikVezu***
  - ***PisekA** zmenši o **kolikVezu***
  - *naplánuj si událost **NALOŽENO** na okamžik  
**Čas+dobaNakládání** // další význam slova „programování“*

# Složitější úloha

**Obchodní dům** sestává z několika oddělení ležících v různých patrech, mezi nimiž jezdí výtah. Do obchodního domu přicházejí **zákazníci** se svými **seznamy oddělení**, která chtějí navštívit. V každém oddělení obsluhuje jedna **prodavačka**, která má svou **rychlost** udávající, za jak dlouho obslouží jednoho zákazníka, **zákazníci čekají ve frontě** a pokud zákazník čeká déle, než je jeho **trpělivost**, odejde z fronty a jde jinam, pokud to už není jeho poslední nákup.

Na výtah **zákazníci také čekají ve frontě**, ale tady se **trpělivost neuplatňuje**. Výtah se řídí **algoritmem výtahu** (směr jízdy mění jen tehdy, když nevezde žádné pasažéry a zároveň na něj nikdo nečeká ve směru jízdy a zároveň na něj někdo čeká ve směru, odkud přijel).

**Cíl: Chceme spočítat, v kolik hodin odejde poslední zákazník.**

# Příklad vstupních dat

Vstupní data pro simulaci obchodního domu

=====

Oddělení:

	jméno	patro	rychlost obsluhy
O	papírnictví	0	5
O	potraviny	0	15
O	drogerie	0	6
O	textil	1	15
O	nábytek	2	20
O	elektronika	3	5
O	CD-DVD	3	4

Výtah

	jméno	kapacita	doba nastoupení	doba vystoupení	doba jízdy z patra do patra
V	elefant	10	1	1	1

Zákazníci

	jméno	příchod	trpělivost	seznam oddělení...
Z	Adam	10	1	CD-DVD
Z	Alžběta	10	5	textil drogerie nábytek
Z	Bořík	10	1	papírnictví elektronika
Z	Barborka	10	1	CD-DVD papírnictví
Z	Cyril	10	2	potraviny papírnictví
Z	Cecilka	10	5	potraviny textil drogerie nábytek
Z	David	11	3	CD-DVD elektronika
Z	Dana	11	2	potraviny textil drogerie nábytek

# Návrh modelu

## Seznam procesů

- zákazník
- oddělení
- výtah

## Seznam událostí

Jak budou procesy zpracovávat události

Je to návrh, takže neexistuje **jediné správné řešení.**

# Pomůcka: diagram stavů

V jakých stavech se proces může nacházet?

Přechody mezi stavy:

- jak vzniknou
- co je při nich potřeba udělat

Obrázky

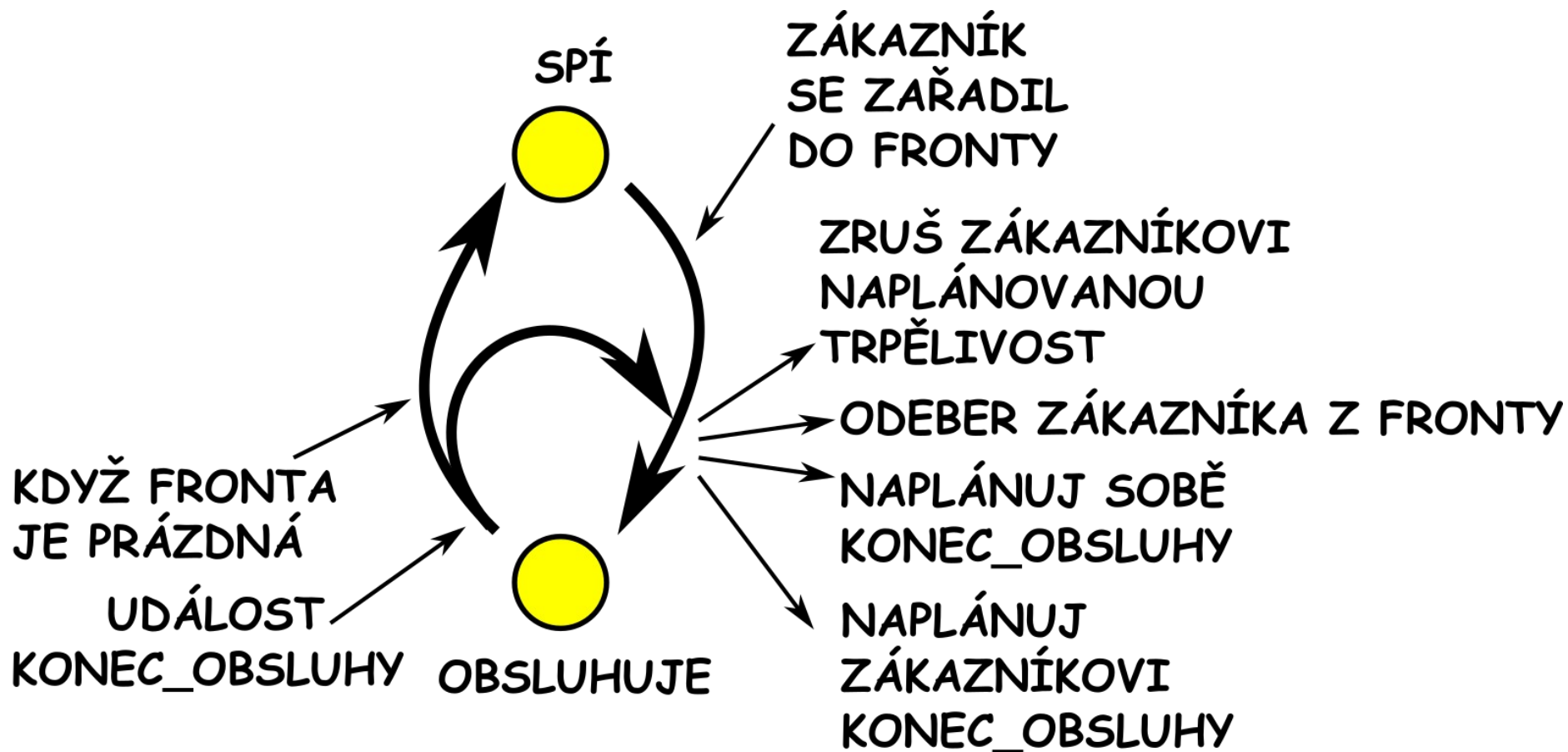
- lze rychle nakreslit
- jsou názorné
- snáz se nad nimi uvažuje/diskutuje (oproti zdrojovému kódu).

*Kreslení obrázků dotažené do krajnosti: UML2*

*<https://www.omg.org/spec/UML/2.5.1/PDF> (796 stran)*

# Pomůcka: diagram stavů

## ODDĚLENÍ



# Jak může vypadat řešení

<https://ksvi.mff.cuni.cz/~holan/py/obchod.txt>

[https://ksvi.mff.cuni.cz/~holan/py/obchod\\_data.txt](https://ksvi.mff.cuni.cz/~holan/py/obchod_data.txt)



# PROGRAMOVÁNÍ ŘÍZENÉ UDÁLOSTMI

# Programování řízené událostmi

## Co známe z diskrétní simulace:

Můžeme mít složitý systém, který

- nemá nějaké ústřední velení, které by všechno řídilo
- místo toho se skládá z mnoha částí, které jsou připravené na to, aby zpracovávaly události
- ta složitá funkčnost vzniká z toho, že každý dělá to, co umí.

# Programování řízené událostmi

## Ten hlavní program:

```
while (ještě_není_konec)
{
    Udalost u = PřijmiUdalost();
    ZpracujUdalost( u );
}
```

# Kdo zpracovává události?

Všichni

Hierarchie objektů

Úřad

Polymorfie a virtuální metody

# Události

## Od uživatele:

- klávesnice
  - stisknout
  - pustit
- myš
  - stisknout
  - pustit
  - pohnout

# Jak je to třeba ve Windows

Komponenty

Hierarchie

Propadání událostí (úřad)

Polymorfismus (virtuální metody)

Transformace lo-level událostí na hi-level

# Jak je to v C# a ve VisualStudios

Knihovna s komponentami

(WFC), WinForms, (VCL), WPF, XNA, WinUI...

Vytvořit aplikaci

Skoro vždy jde spustit

Visuální návrh

Designer-Toolbox-Inspektor

Properties Left, Top, Width, Height, Visible, Enabled, Text

Události

Rozdělený zdrojový kód partial class

Sdílení obslužných proměnných parametr Sender