

PDV 08 2018/2019

## Detekce selhání

**Michal Jakob**

[michal.jakob@fel.cvut.cz](mailto:michal.jakob@fel.cvut.cz)

Centrum umělé inteligence, katedra počítačů, FEL ČVUT



# Detekce selhání

## Systemy založeny na **skupinách procesů**

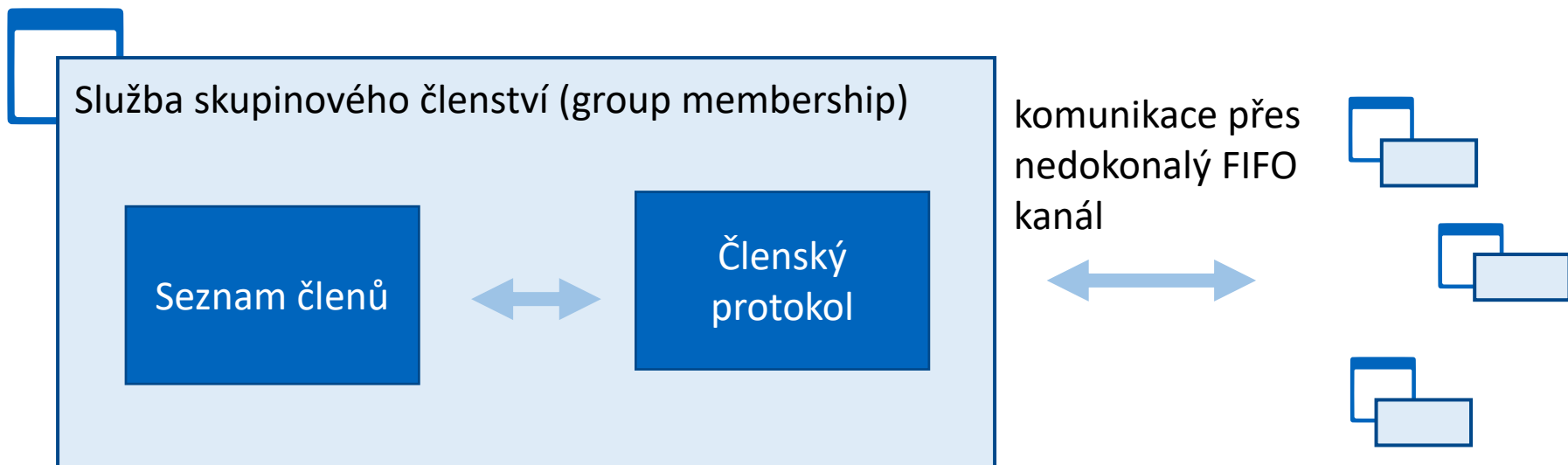
- cloudy / datová centra
- replikované servery
- distribuované databáze

Frekvence selhání roste lineárně s počtem procesů ve skupině  
=> selhání jsou *běžná*.

V následujícím předpokládáme **havárii procesu (crash-stop)** jako model selhání a komunikaci přes ztrátový/nedokonalý FIFO kanál.

# Obecněji: Správa skupin (group membership)

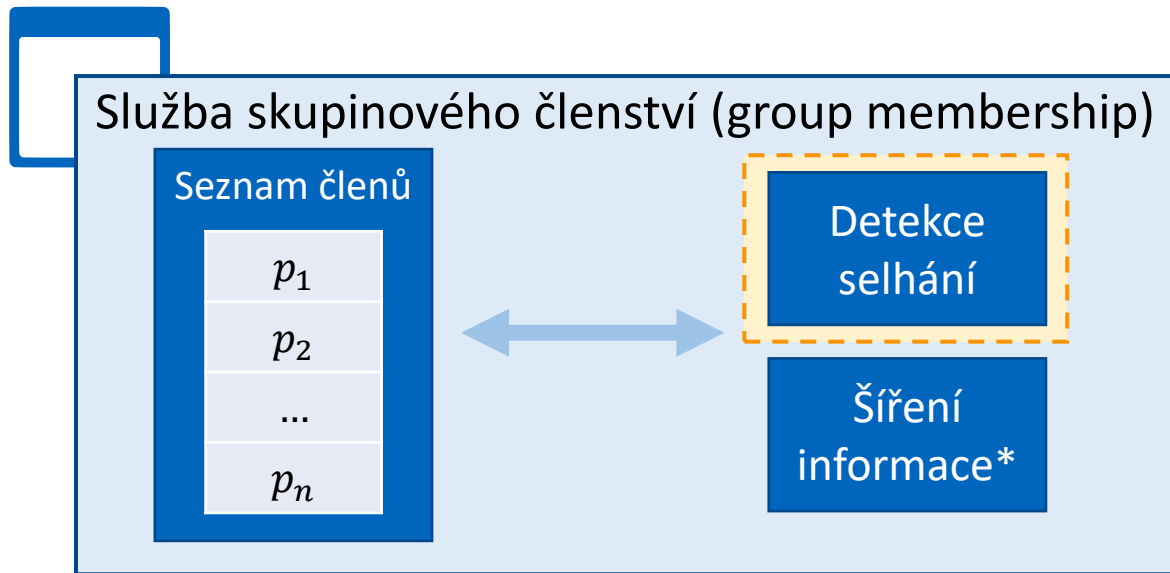
 proces ve skupině



Procesy ze skupiny,  
které dosud nesehaly

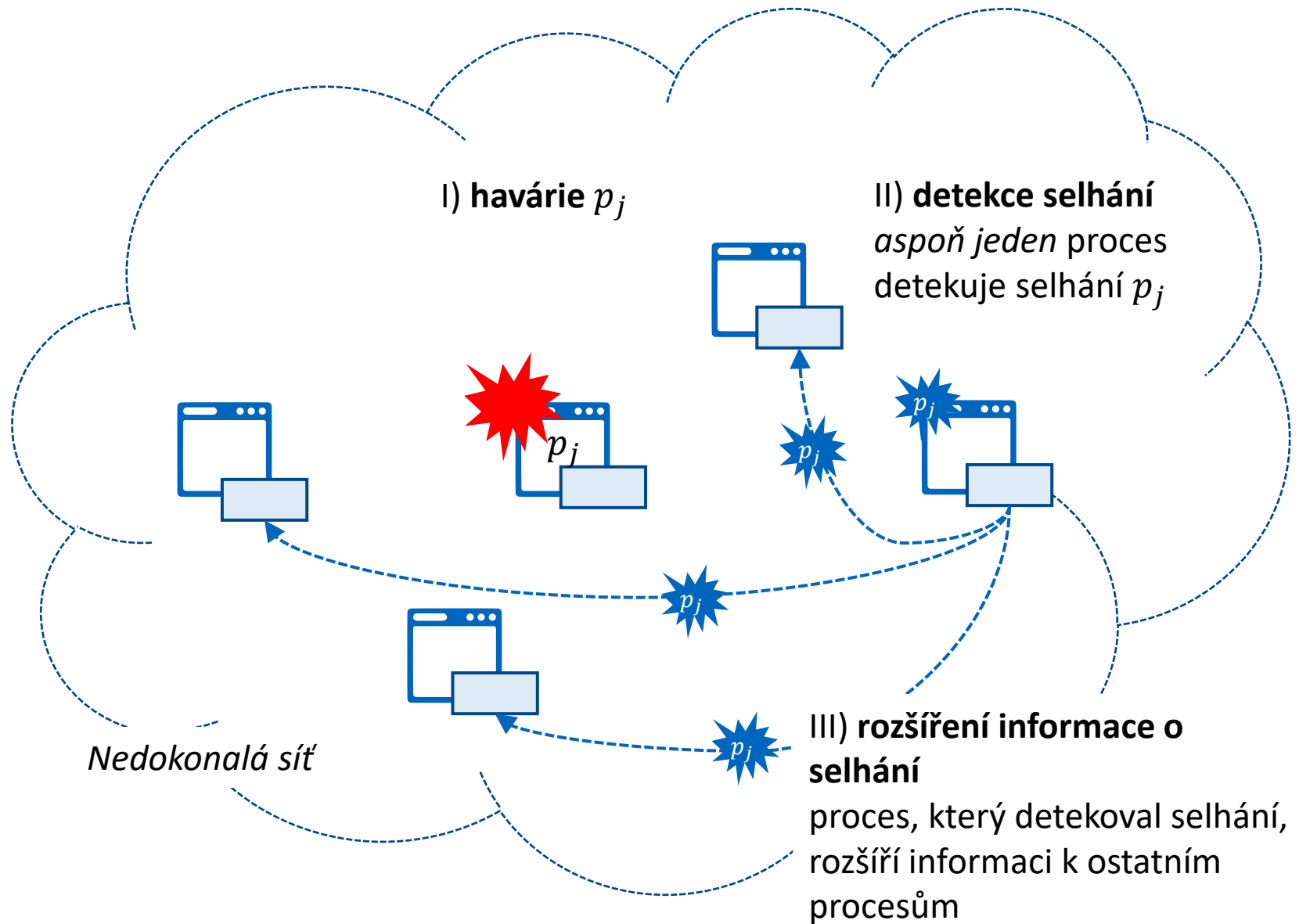
Udržuje seznam procesů při 1)  
příchodu nového procesu, 2)  
**tichého selhání procesu** a 3)  
odchodu procesu

# Dva podprotokoly



\*Informace o selhání  
(a také o vstupu / výstupu procesu do skupiny)

# Průběh detekce





# Vlastnosti detektorů selhání

# Požadované vlastnosti detektorů selhání



## Úplnost

= **každé selhání** je časem **detekováno** aspoň jedním **bezvadným** procesem

Detekce vadným (faulty) procesem se nepočítá.

## Přesnost

= **nedochází k mylné detekci**

Pokud je proces detekován jako havarovaný, tak skutečně havaroval (žádná false positives)

Nelze dosáhnout obou vlastností současně při nedokonalé komunikaci

(detekce selhání je ekvivalentní problému konsensu)

# Požadované vlastnosti detektorů selhání

## 100% Úplnost

Úplnost nelze obětovat <= potřebujeme vědět, že proces havaroval

Vždy splněna v prakticky používaných detektorech

## <100% Přesnost

Chybně detekované selhání vede ke zbytečným operacím, ale lepší než přehlédnutí havárie

Pouze částečně (pravděpodobnostní garance)



# Další vlastnosti detektorů selhání

## Rychlost detekce

= čas do okamžiku, kdy **první** proces detekuje selhání

## Škálovatelnost

= **počet** posílaných **zpráv** a **rovnoměrné rozložení** komunikační zátěže

Nechceme úzká hrdla a centrální body selhání

# Vlastnosti detektorů selhání

Úplnost

Přesnost

Rychlost

Škálovatelnost

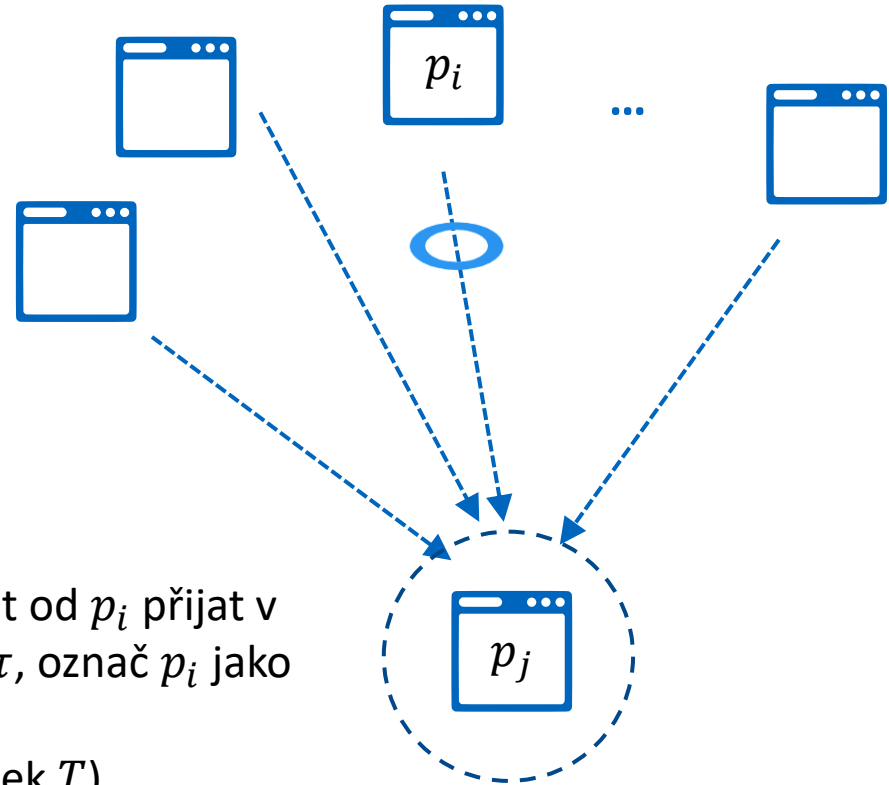
Chceme zajistit i v případě **současného selhání** více procesů



# Základní protokoly pro detekci selhání

# Centralizovaný heartbeat

- Hearbeats jsou **odesílány periodicky** (každých  $T$  časových jednotek) jednomu vybranému procesu.
- Heartbeat má **pořadové číslo**. (Po odeslání heartbeatů je inkrementován lokální **čítač heartbeatů** každého procesu.)

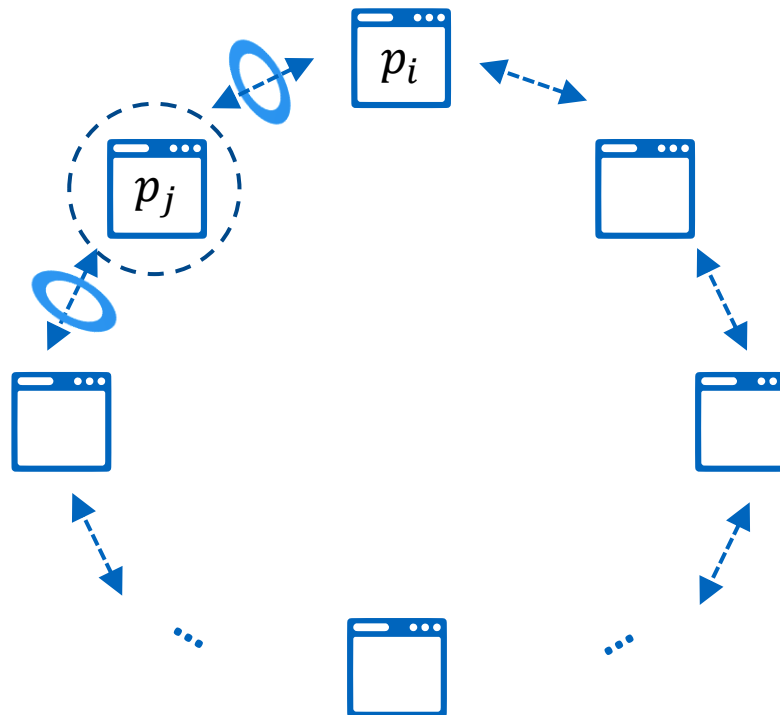


Není-li heartbeat od  $p_i$  přijat v časovém limitu  $\tau$ , označ  $p_i$  jako **havarovaný** ( $\tau$  typicky násobek  $T$ )

- ☺ **Úplný** pro všechny procesy s výjimkou  $p_j$
- ☹ Selhání  $p_j$  **není detekováno**.
- ☹ Při vysokém počtu procesů může být  $p_j$  **přetížený**.

# Kruhový heartbeat

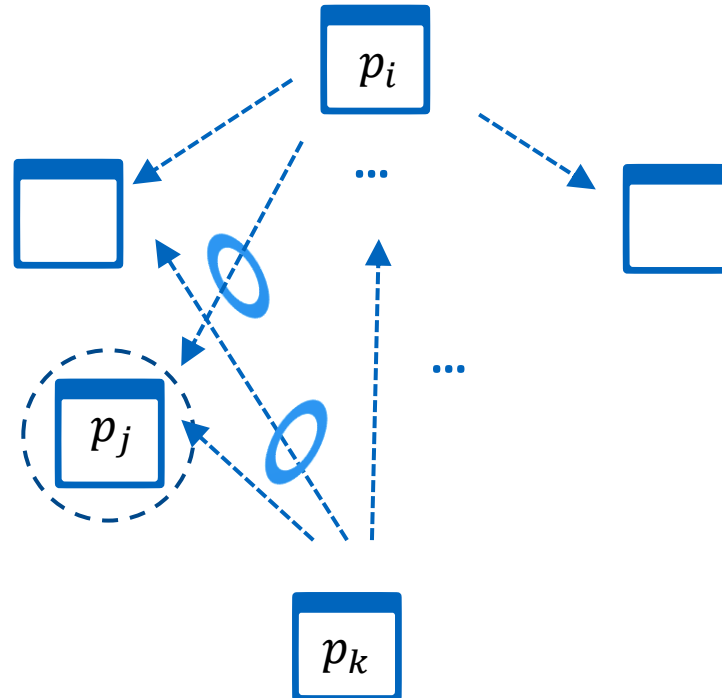
- Hearbeats jsou odesílány periodicky **sousedům každého** procesu (jedno- nebo oboustranně)



- ☺ **Není centrální bod**
- ☹ **Neúplné** při současném selhání více procesů
- ☹ Je třeba udržovat **kruh**

# Všichni-všem (all-to-all) heartbeating

- **Každý** proces posílá periodicky heartbeats **všem** ostatním procesům.



- ☺ **Rovnoměrná zátěž** všech procesů (ale poměrně **vysoká**).
- ☺ **Úplný**: pokud zůstane aspoň jede nehavaraný proces, detekuje selhání libovolného jiného procesu.
- ☹ **Nízká přesnost**: stačí, když jeden proces nedostane včas heartbeats a může označit všechny procesy jako havarované.



# Porovnání a analýza detektorů

# Porovnání/analýza detektorů

Úplnost

Vždy **garantována**

(Ne)Přesnost

$p_m$  ... pravděpodobnost, že dojde k **chybné detekci** během  $T$  časových jednotek

Rychlost

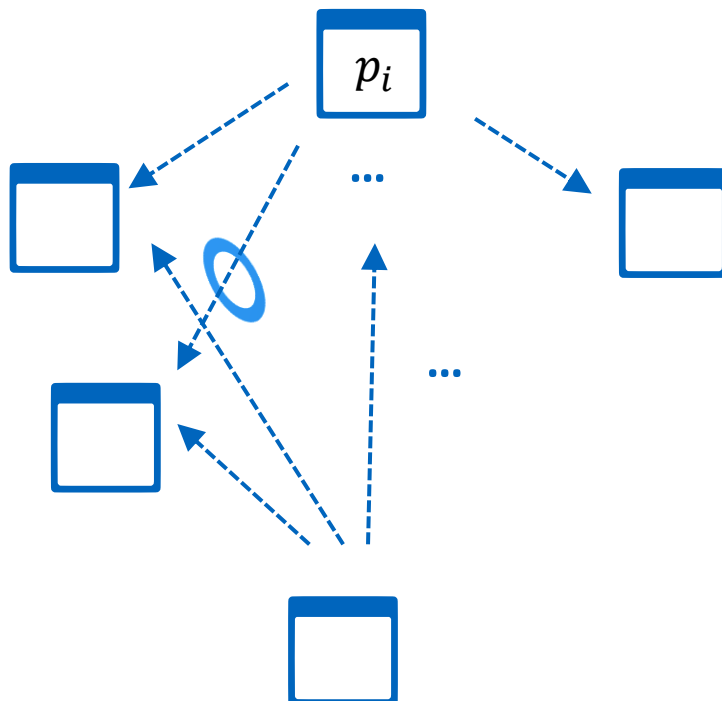
$T_d$  ... průměrný počet časových jednotek do **první detekce** selhání

Škálovatelnost

$L$ ... **komunikační zátěž**, tj. počet zpráv na proces a jednotku času



# Analýza all-to-all heartbeating



**Počet procesů:**  $N$

**Perioda heartbeatu:**  $T$

**Timeout interval:**  $\tau$

**Přesnost:** závisí na  
spolehlivosti komunikace

**Maximální čas do první  
detekce:**  $T_d = \tau + T$

**Komunikační zátěž:**  $L = \frac{N}{T}$

Komunikační zátěž je  
**lineární** v počtu procesů.

# Optimální detektor

All-to-all heartbeat má **lineární zátěž**  $L = O(N/T)$

- celkový počet posílaných zpráv ve skupině procesů pak roste **kvadraticky**

Proč?

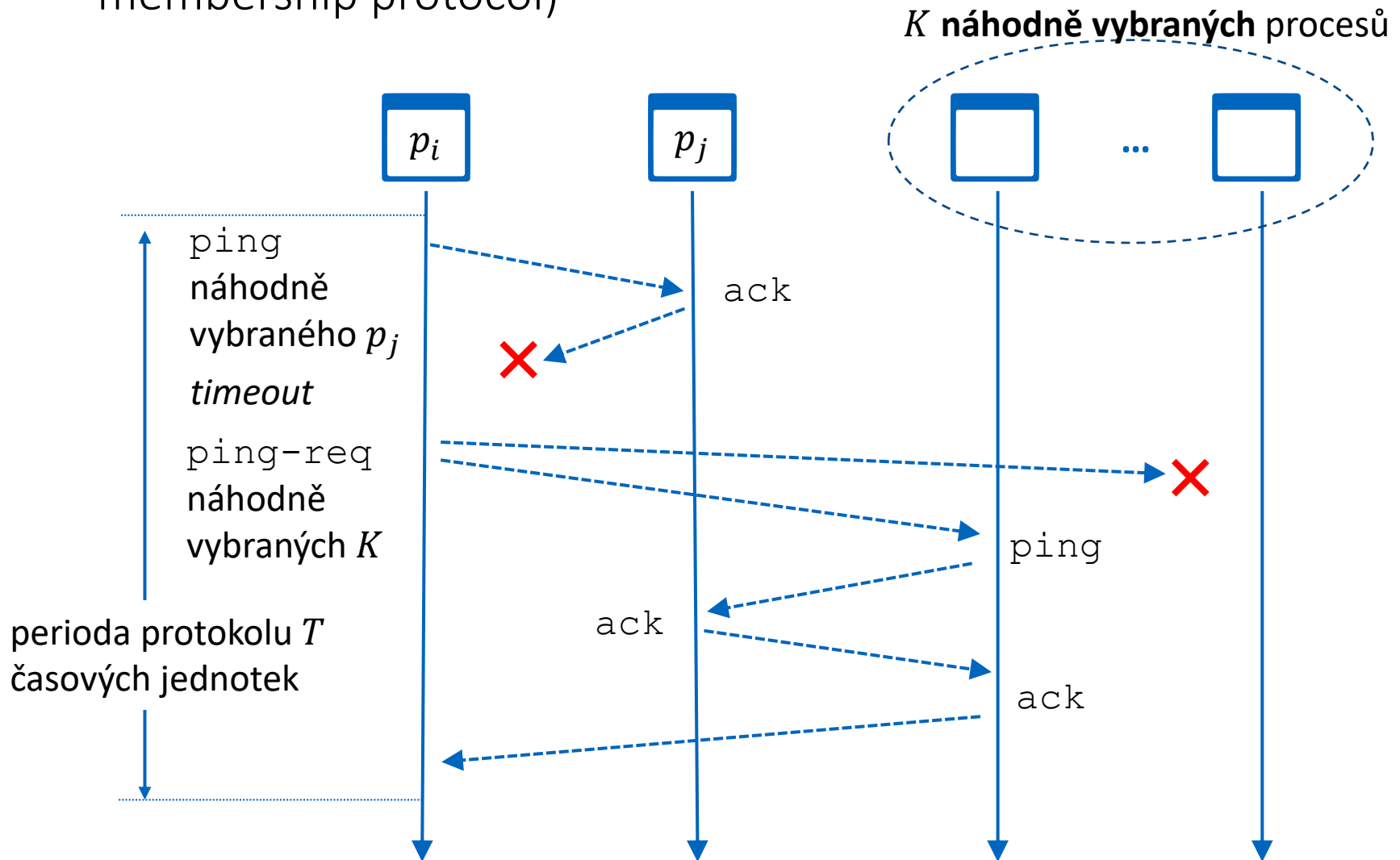
**Všechny** procesy se snaží detekovat selhání samy, informaci o detekovaném selhání si nevyměňují!

Pro **efektivnější detekci** selhání je potřeba

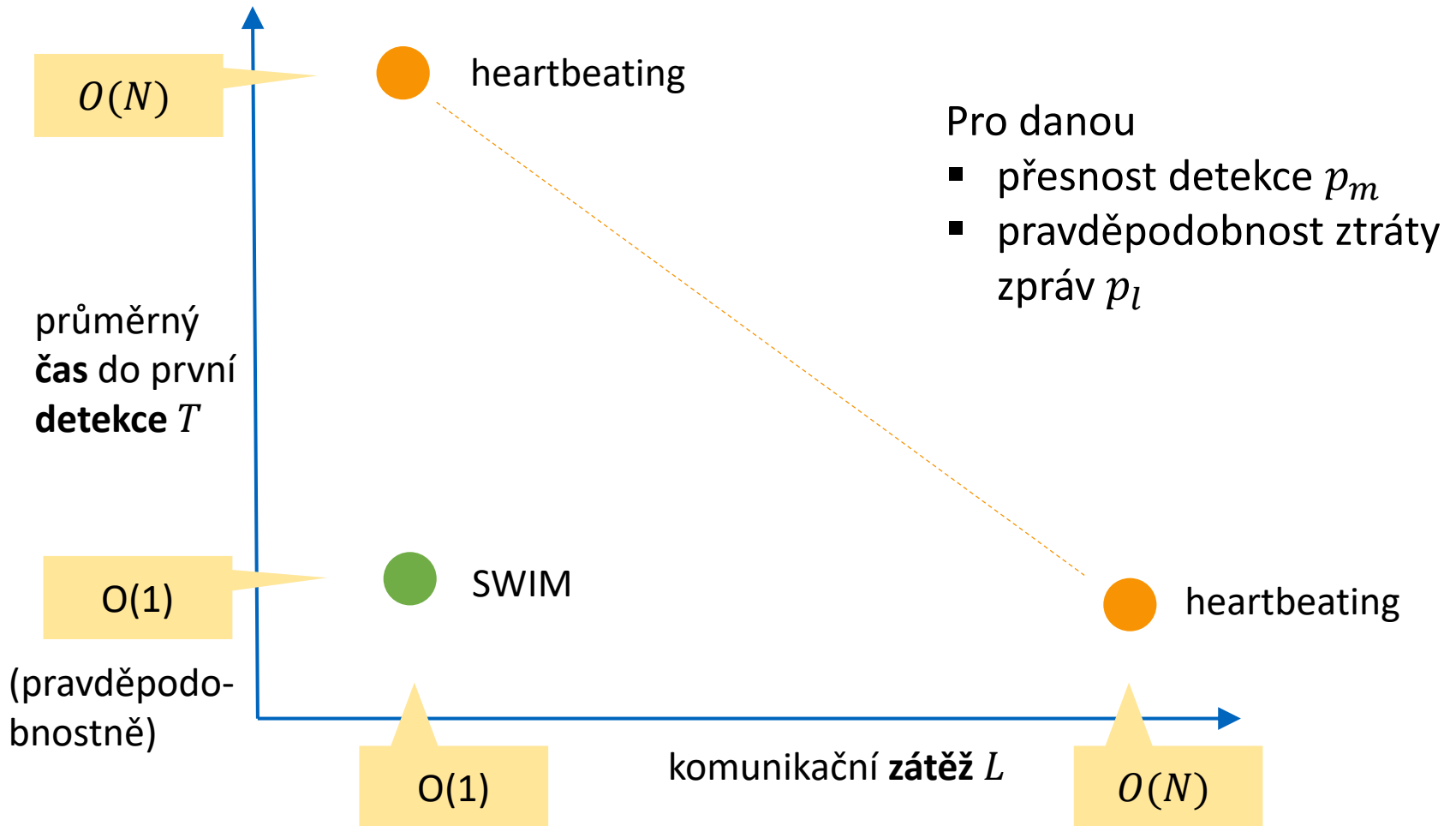
- oddělit detekci selhání a šíření informace o selhání
- použít jinou detekci než založenou na heartbeat (ale na opačném mechanismu)

# SWIM Detektor Selhání

(Scalable weakly consistent infection-style proces group membership protocol)



# SWIM versus heartbeating



# Další vlastnosti SWIM

Průměrný čas do první detekce $T_d$	<ul style="list-style-type: none"><li>▪ nezávisí na počtu procesů</li><li>▪ Lze ukázat, že <math>E[T_d] = \frac{e}{e-1}T \sim 1.58T</math> (pro velká <math>N</math>)</li></ul>
Komunikační zátěž $L$	<ul style="list-style-type: none"><li>▪ nezávisí na počtu procesů</li></ul>
(Ne)přesnost $p_m$	<ul style="list-style-type: none"><li>▪ lze nastavit nastavením <math>K</math></li><li>▪ klesá exponenciálně s rostoucím <math>K</math></li></ul>
Úplnost	<ul style="list-style-type: none"><li>▪ Selhání bude <i>časem</i> detekováno – průměrný čas <math>\frac{e}{e-1}T</math></li><li>▪ Čas do detekce lze omezit i deterministicky na <math>O(N)</math> (viz dále).</li></ul>

# Časově garantovaná úplnost

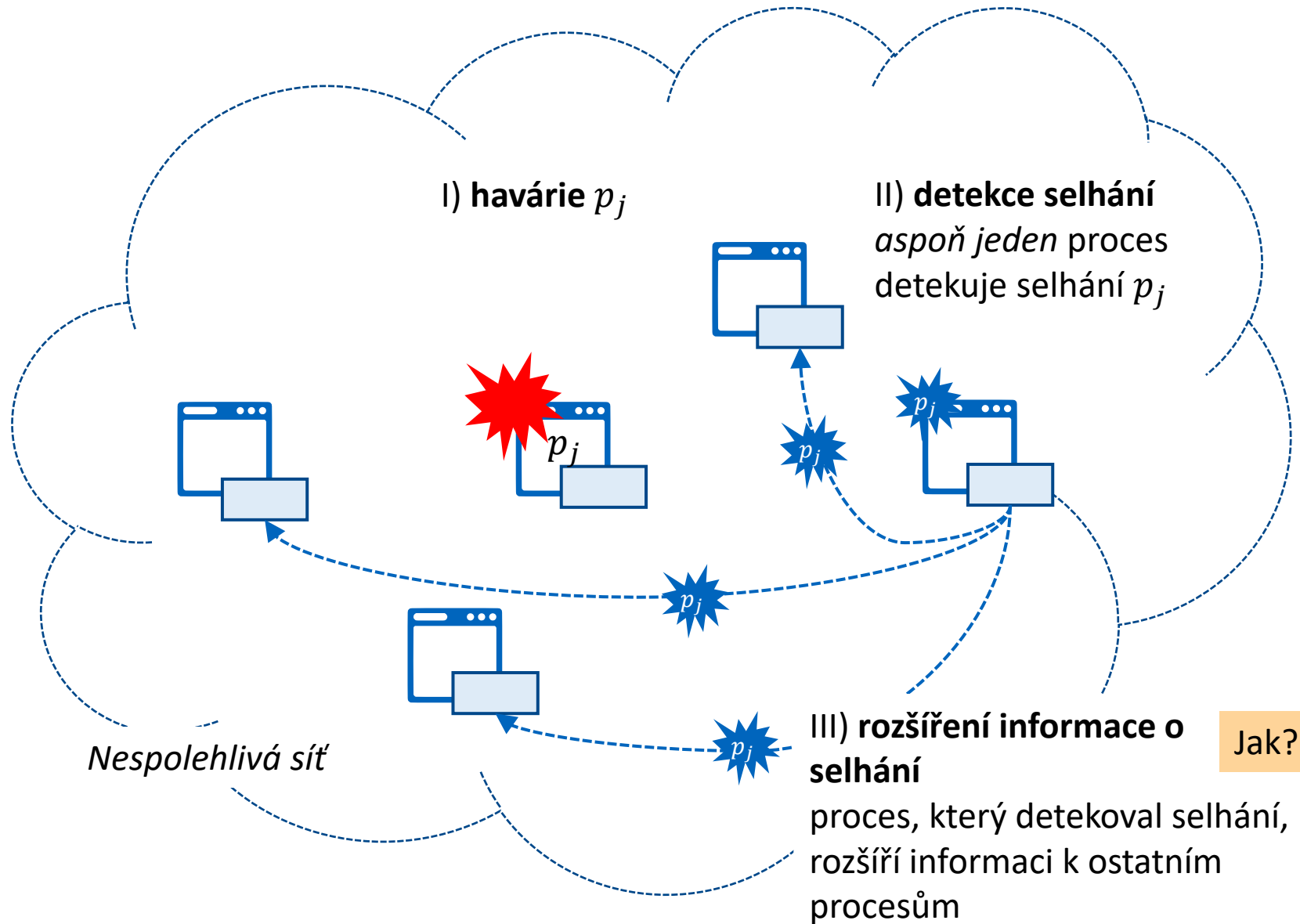
Hlavní myšlenka: Zvolit každý proces ze seznamu **právě jednou** při posílání `ping` zprávy.

- postupné procházení seznamu
- prermutace seznamu po obeslání všech procesů

Každé selhání je detekováno v maximálně  $2N - 1$  periodách protokolu

Ostatní vlastnosti SWIM jsou **zachovány**.

# Šíření informace



# Jak šířit informaci?

## Multicast (hardware / IP)

- ne vždy k dispozici

## Point-to-point (TCP / UDP)

- nákladné

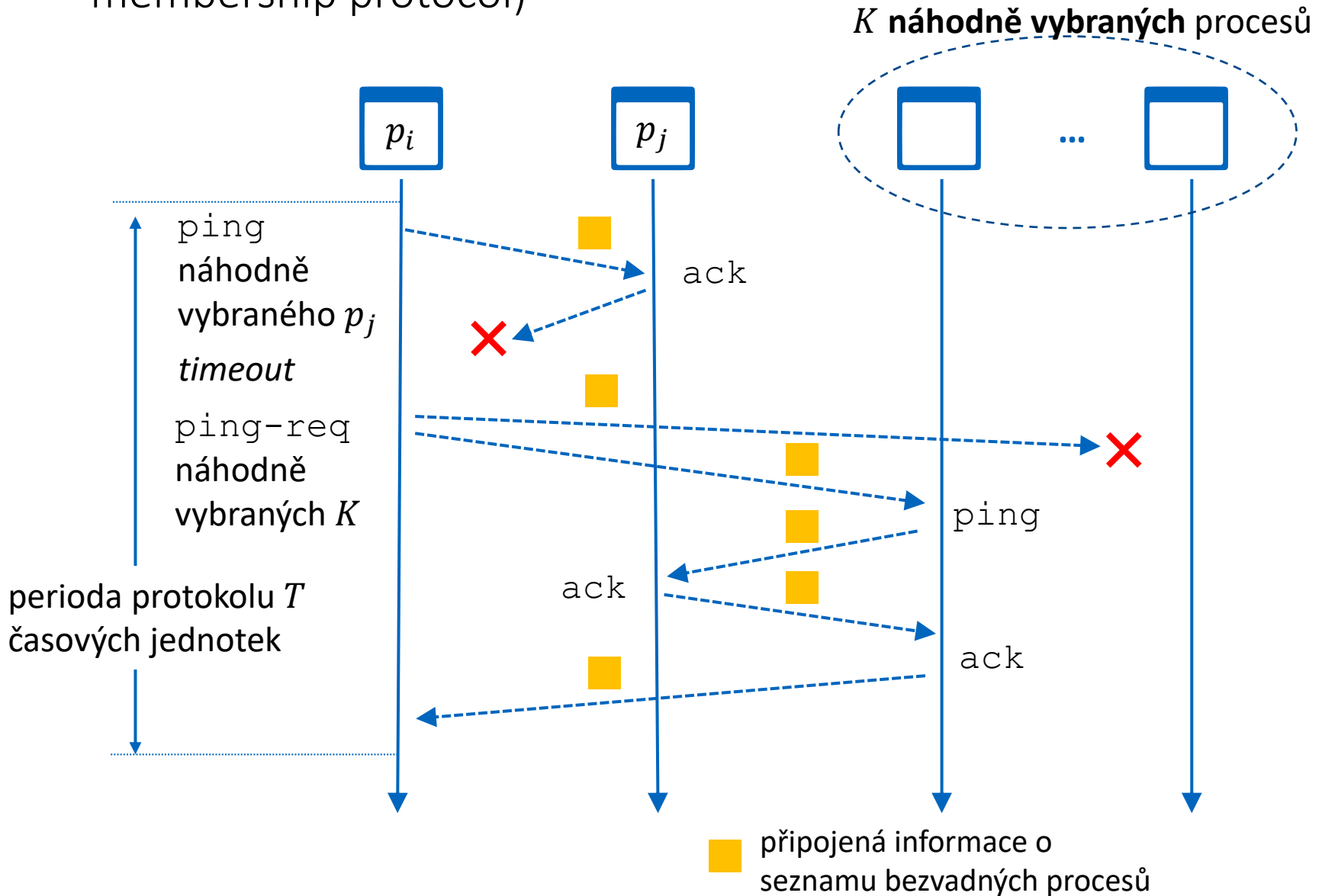
**Bez dodatečných zpráv:** připojit ke zprávám protokolu detekce selhání!

- tzv. infekční styl šíření informace



# SWIM Failure Detector

(Scalable weakly consistent infection-style process group membership protocol)



# Vlastnosti šíření informace

**Epidemický styl** šíření informací: Po  $\lambda \log N$  periodách protokolu lze očekávat, že  $N^{-2(\lambda-1)}$  procesů nedostalo aktualizaci (update).

## Implementace

- každý proces udržuje lokální buffer obdržených aktualizací
- může mít fixní délku nebo fixní dobu expirace
- doba expirace ovlivňuje míru konzistence

# Souhrn

Detekce selhání **klíčový stavební** prvek pro spolehlivé DS.

Jednoduchá detekce pomocí **heartbeat** vyžaduje lineární komunikační zátěž každého uzlu.

**Pravděpodobnostní SWIM** algoritmus je škálovatelný s konstantní zátěží.