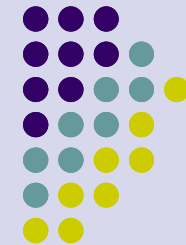


# Poruchy

---



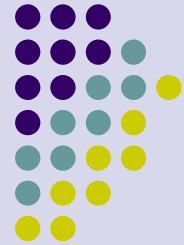
Přednášky z Distribuovaných systémů  
Ing. Jiří Ledvina, CSc.



# Odolnost proti poruchám

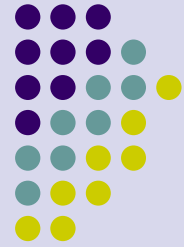
- partial failure – částečná chyba
- error isolation – ostatní komponenty nejsou zasaženy
- automatická obnova z částečných chyb

# Synchronní a asynchronní systémy



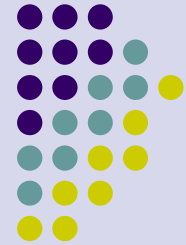
- synchronní systém
  - systém reaguje do stanoveného časového intervalu
- asynchronní systém
  - u systému není zaručen časový interval, během kterého musí systém odpovědět

# Odolnost proti poruchám (spolehlivost) závisí na



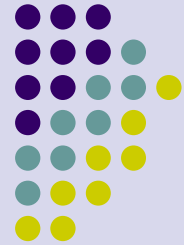
- dostupnost (availability)
  - část doby, během které systém splňuje svou specifikaci
  - pravděpodobnost, že je systém funkční v danou dobu
- spolehlivost (reliability)
  - Míra úspěchu se kterým systém přizpůsobí své chování k nějaké specifikaci
  - Pravděpodobnost že systém selhal během dané doby.
  - Typicky používané pro popis systémů, které nelze opravit nebo kde je kritický spolehlivý nepřetržitý provoz systému

# Odolnost proti poruchám (spolehlivost) závisí na



- bezpečnost (safety)
  - Jestliže systém dočasně selže, jeho specifikace se přizpůsobí a nic katastrofického se nestane
- udržovatelnost (maintainability)
  - Míra jak snadno je možné systém opravit

# Základní pojmy



- error (chyba, omyl, odchylka) – část stavu systému, která může vézt k poruše.
- fault (porucha, nedostatek, chyba) – chyba ve vnitřních stavech komponent systému nebo v návrhu systému
- failure (selhání) – odchylka z chování, které je popsáno v jeho specifikaci
- erroneous state (chybový stav) – takový vnitřní stav systému, při kterém existují okolnosti, za nichž další zpracování normálním algoritmem povede k selhání (failure), které není přisuzované následující poruše.

# Závislosti

---



- Fault (porucha) → error (chyba) → failure (selhání)



# Typy poruch

- stálé poruchy (Hard faults)
  - stálá (permanent) – pokračování po opravě komponenty – programové, technické chyby
  - výsledná selhání jsou označována jako "tvrdá"
- dočasné poruchy (Soft faults)
  - přechodné nebo občasné
  - přechodná (transient) – objeví se a zmizí – opakování operací
  - občasný (intermittent) – obtížně diagnostikovatelný – chybný kontakt
  - představují více než 90% všech chyb
  - výsledná selhání jsou označována jako "měkká"



# Zpracování chyb

---



- prevence (preventing)
- odstranění (removing)
- předpovídání (forecasting)

# Odolnost proti poruchám (fault tolerance)

---



- systém může provádět služby, vedoucí k prevenci chyb



# Typy selhání (failure)

- zhroucení (crash) – server se zastaví, dokud se nezastavil, pracoval dobře
- vynechání (omission) – server neodpoví na příchozí požadavek
  - při příjmu – server chybuje při příjmu
  - při posílání – server chybuje při vysílání
- časování – reakce je mimo časový interval
- chybná reakce – server odpovídá nekorektně
  - chybná hodnota – chybná hodnota v odpovědi
  - chybná změna stavu – odchylka od korektního postupu řízení
- svévolná (arbitrary) – libovolné odpovědi v libovolnou dobu – Byzantinské chyby

# Fail silent systems

---



- server nedává najevo své problémy
- ostatní mohou reagovat nekorektně (zastavení serveru)

# Fail-safe server (zabezpečený proti poruše)

---



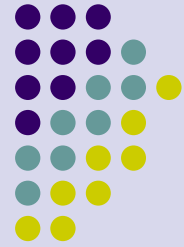
- server produkuje náhodné výstupy
- rozpoznatelný ostatními procesy



# Maskování chyb a redundance

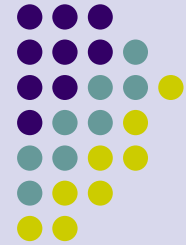
- cílem je skrýt výskyt selhání (failure)
- informační (datová) redundance – replikace nebo redundantní kódování dat - obnova (Hammingovy kódy), parita v paměti,
- časová redundance – násobné provádění
  - opakování přenosu. timeout
- fyzikální redundance – komponenty navíc – hardware, software
  - práce s redundantními komponentami, nikoliv s daty
  - odolnost vůči ztrátě nebo poruchám komponent
  - násobné fyzické komponenty ( $P_n = p_n$ )
  - nadbytečné procesy podobným významem

# Process resilience (pružnost, elasticnost)



- ochrana proti selhání procesů – replikace procesů do skupin
- organizování několika identických procesů do skupin
- zprávy ve skupině přijímají všechny procesy
- skupina může být
  - statická
  - dynamická (join, leave, destroy – group management)
- proces může být členem několika skupin

# Process resilience (pružnost, elastičnost)



- organizace skupiny může být
  - plochá (flat) – neexistuje centrální prvek, komplikované
  - hierarchická – ztráta koordinátora znamená zhroucení
- členství ve skupinách
  - server skupiny – jednoduché, centrální prvek
  - distribuované – spolehlivé skupinové doručování
- problém – opuštění skupiny – není indikováno zhroucení
  - testování přítomnosti procesů
  - ale zpožděná odpověď
  - opuštění (leave) a připojování (join) musí být synchronní
  - znovuvytvoření skupiny – může iniciovat jakýkoliv proces





# Maskování poruch a replikace

- skupina procesů – může maskovat poruchy (identické procesy) – replikace procesů
- dva způsoby řešení
  - primary backup protocols
  - replicated write protocols



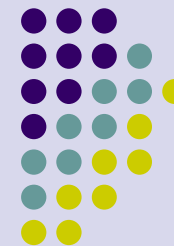
# Maskování poruch a replikace

- primary backup protocols
  - hierarchické uspořádání (primary, backup)
  - primary koordinuje všechny operace zápisu
  - výpadek se indikuje
    - primary - periodickým vysíláním zprávy (žiji)
    - backup – dotazovací zprávou (žiješ?)
    - problém s nastavením timeoutu (asynchronní systém)
  - pokud se primary zhroutí, backup – vyvolání algoritmu výběru – volba nového primary
  - procesy jsou organizovány hierarchicky

# Maskování poruch a replikace

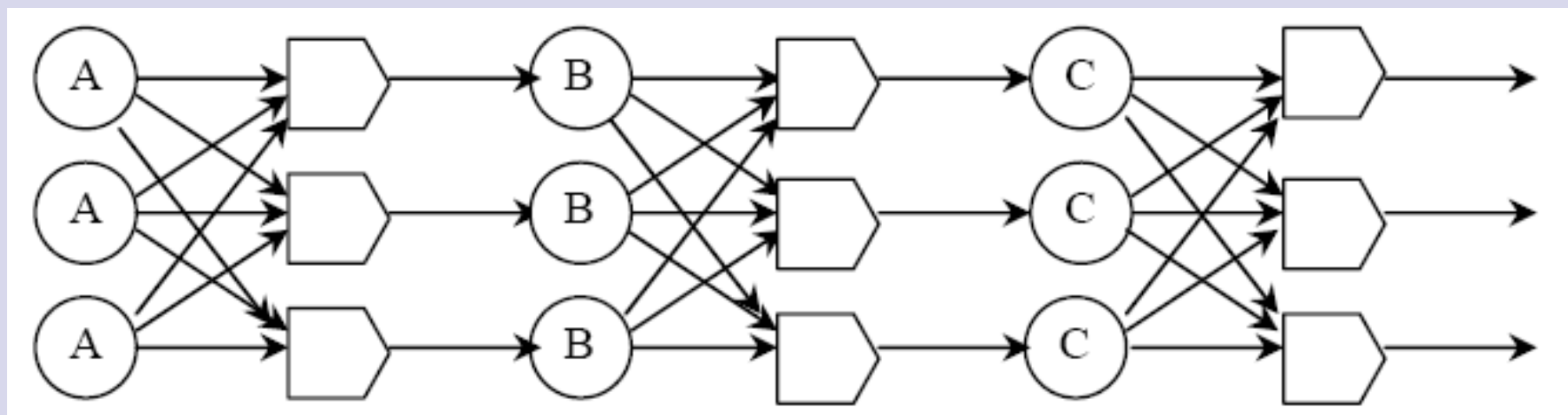
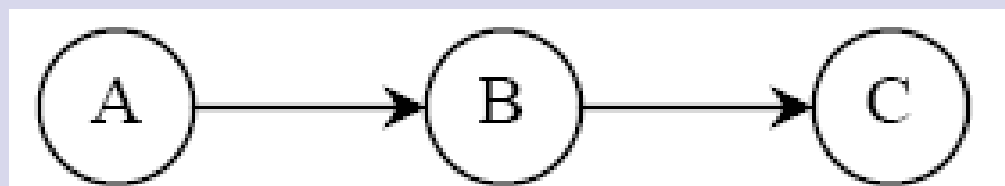


- replicated write protocol
  - používá aktivní replikaci nebo protokoly založené na hlasování
  - ploché (flat) skupiny – protokoly založené na hlasování pro realizaci oprav
  - neexistuje úzké místo vzhledem k chybám



# Př. Aktivní replikace

- Např. trojnásobná modulární redundance





# Skupiny procesů a tolerance chyb

- kolik je třeba replikovat procesů
- systémy s replikací zápisu – pouze zapisují (zjednodušené)
  - systém je  $k$ -odolný proti poruchám – může přežít chybu v  $k$ -komponentách
  - stačí  $k+1$  komponent –  $k$  nefunguje, jeden stačí
- Byzantinské chyby
  - pro systém  $k$ -odolný proti poruchám musí být  $2k+1$  procesů
  - $k$  – produkuje chybnou odpověď
  - $k+1$  – produkuje správnou odpověď
  - řešení pomocí hlasování a majority
  - neví se ale kolik procesů selhalo



# Dohoda v systémech s poruchami

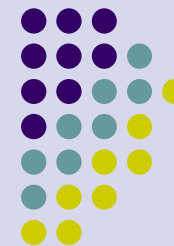
- náhrada hlasování
- replikované procesy mají dosáhnout dohody
  - výběr koordinátora
  - rozhodnutí o provedení/zrušení transakce
  - rozdělování úloh mezi dělníky
  - synchronizace
- základní cíl
  - bezchybné procesy musí dosáhnout dohody v konečném počtu kroků



# Dohoda v systémech s poruchami

- slavný problém – problém dvou armád
  - dosažení dohody pomocí jednobitové informace
  - dosažení dohody o započetí útoku
  - komunikace nespolehlivým kanálem
- dva základní případy pro řešení
  - generálové – bez chyby, komunikace nespolehlivá – nemá řešení
  - generálové s chybami, komunikace spolehlivá – problém Byzantinských generálů
  - výsledek – pro  $3m+1$  procesů musí být  $2m+1$  korektních
  - není-li spolehlivá komunikace, musí být všechny procesy v pořádku

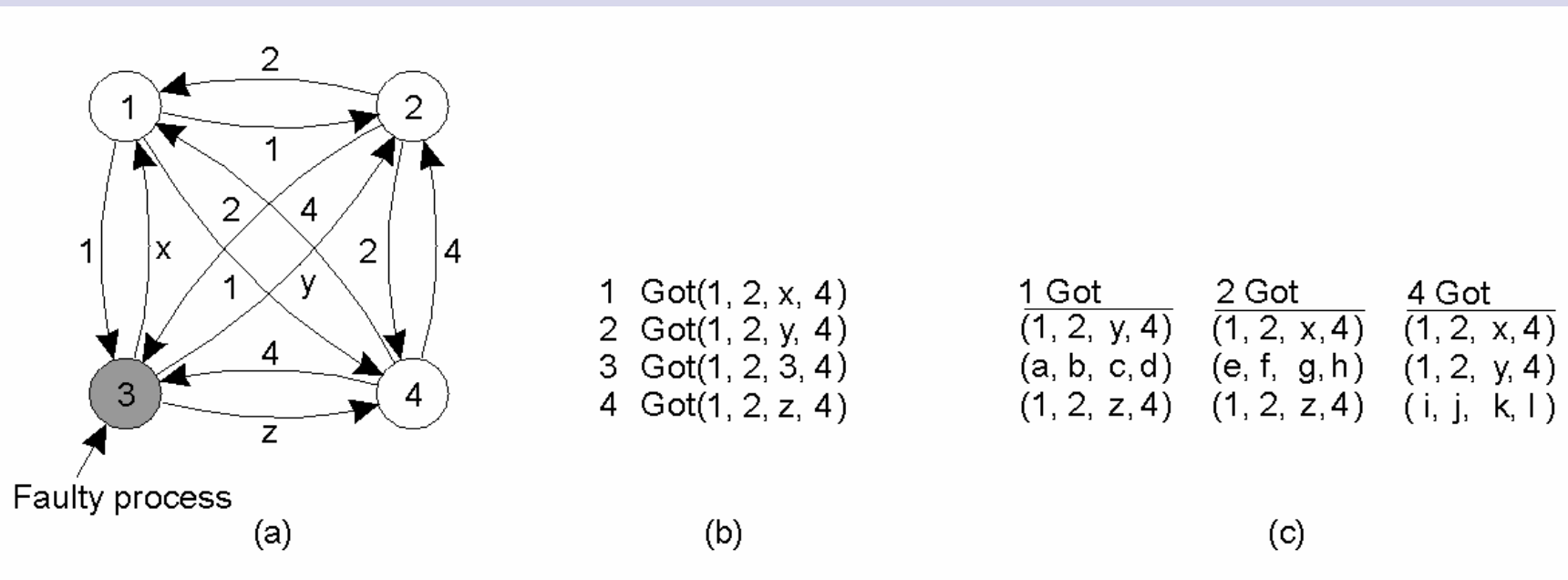
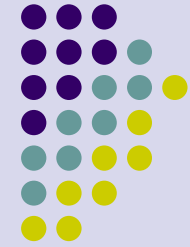
# Byzantinští generálové



- Problém Byzantinských generálů pro tři věrné a jednoho zrádce.
  - Generálové oznamují sílu svých baterií (v jednotkách 1000 vojáků).
  - Vektory vytvořené generály na základě (a).
  - Vektory které každý generál obdrží ve kroku 3.



# Byzantinští generálové





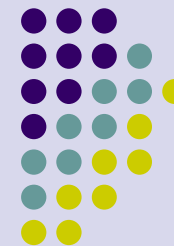
# Obnova po chybě

- redundantnost pomáhá odstranit chyby "za běhu"
- chyby se stávají transparentní vzhledem k okolí
- netransparentní zajištění systému proti chybám – obnova po chybě (failure recovery)
- kontrolní body (checkpoint) – periodické ukládání stavu
  - koordinované (synchronizované hodiny, blokování)
  - nekoordinované
- logování (logging) – zaznamenávání operací se stavem
  - synchronní, asynchronní

# Proces obnovy

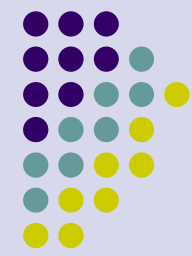


- host obnovený do předchozího stavu musí opakovat všechny operace až do místa chyby
- musí poslat duplicitní zprávy
- ostatní musí být schopni rozpoznat duplicitní zprávy a zahazovat je
- ostatní host systémy se také musí vrátit do předchozího stavu (kaskádní rollback)
- všechny systémy se musí dostat do téhož stavu (recovery line)



# Incarnation Numbers (etapa)

- sekvenční inkarnační čísla
- každé období je jimi charakterizováno
- jsou přítomna v každé zprávě
- v systému jsou zapamatována ve stálé paměti
- pokud se systém probouzí (převtěluje), zasílá všem nové inkarnační číslo
  - < - duplicitní zpráva, zahodit
  - > - čekej na zprávy pro obnovu
  - = zpracuj zprávu



# Kontrolní body

