Volba leadera a chord systém

Paralelní a distribuované systémy, přednáška 8

Tomáš Urbanec

Katedra informatiky PřF UPOL

14.11.2024

Co nás čeká?

1. Volba leadera
   - Algoritmy Bully a Ring
   - Raft: Volba leadera
   - Ad-hoc sítě
   - Další možnosti
2. Chord systém
   - $\approx$ distribuovaná hashovací tabulka

Volba leadera

Základy

Volba leadera

- Leader = koordinátor, vstupní uzel DS, . . . obecně významný uzel.
- Úkol?
  - komunikace s uživatelem,
  - koordinace,
  - vzájemné vyloučení,
  - topologie,
  - . . .
- Všechny uzly se musí shodnout, kdo je leader.
- V různých okamžicích mouhou být různí leadeři.

Základy

Volba leadera

- Souvisí s dalšími problémy
  - Vzájemné vyloučení
  - Obecná shoda v DS
- Předpoklady
  - Uzly mají ID (preferujeme větší).
  - Všichni o všech ví.
  - Uzly mohou vypadnout.
    - Běžný uzel (následovník)?
    - Leader?

Bully algoritmus

Volba leadera

- „Nejsilnější vyhrává.“
- Garcia-Molina, 1982
- Kandidující uzly:
  - ty, které zjistí, že leader nereaguje.
  - nově přidané (zotavené).
  - ty, které převzaly kandidaturu (níže).

→ zpráva ELECTION(id) všem s vyšším ID.

- Uzel obdrží zprávu ELECTION(*cid)* a
  - pokud je jeho $id > cid$, odpoví OK a sám se stává kandidátem.
- Pokud uzel obdrží zprávu OK přestává kandidovat.
- Nakonec to vzdají všichni kromě jednoho (nejvyšší id)-
- Ten všem oznámí, že je novým leaderem. Jak to pozná?
- V nejhorším případě $O(n^2)$ zpráv.
- (tabule)

Ring algoritmus

Volba leadera

- Logický kruh (overlay).
- Všichni znají následovníka (i další uzly).
- Volby zahajuje ten, kdo zjistí, že leader nereaguje (i více).
- Nakonec kandidují všichni běžící.
- Pošle zprávu ELECTION se svým *id* následovníkovi.
- Pokud následovník nefunguje, přeskočí ho (opakování).
- Uzel obdrží zprávu ELECTION:
  - neobsahuje jeho *id* $\rightarrow$ přidá své *id* do seznamu a pošle dále
  - obsahuje jeho *id* $\rightarrow$ zpráva LEADER se stejným seznamem
- Zpráva LEADER všem oznámí, kdo je leader (max id).
- Původně odesílající uzel ji nakonec zahodí.
- Vždy $O(n)$ zpráv.
- (tabule)

Raft

Základy

- Raft (Ongaro, Ousterhout, 2013)
- Algoritmus pro shodu v DS s možnými chybami (příště).
- Široce používaný.
- Základ dalších algoritmů.

$\approx$ Distribuovaný log aplikovaných operací.

- Stejné operace ve stejném pořadí všude.

  $\rightarrow$ Stejný stav všude.

- Dnes jen část (volba leadera).
- Více uvidíme příště.

Raft

Volba leadera

- Stavy uzlů:
  - Leader
  - Následovník
  - Kandidát
- Většinové kvórum (!)
- *Term* – Volební období.
- *Timeout* – doba po kterou následovník uznává leadera.
- *Heartbeat* – leader se pravidelně hlasí (obnovuje timeout).
- *Split-vote* – nikdo nezískal kvórum (restart, náhodné zpoždění)
- Na počátku jen následovníci.
- Při vypršení timeoutu následovník vyvolá volby.

Raft

Volba leadera

- Vyvolání voleb – následovník
  - zvýší svůj term,
  - stane se kandidátem,
  - hlasuje sám pro sebe,
  - požádá o hlas všechny ostatní (zpráva).
- Uzel obdrží žádost o hlas
  - Pokud ještě nehlasoval, potvrdí hlas.
  - Jinak ignoruje.
- Výsledek
  - Vyhraje volby
    - kandidát dostane většinu hlasů,
    - prohlásí se leaderem.
  - Jiný kandidát se prohlásí leaderem (zpráva) a
    - má term větší nebo roven mému $\rightarrow$ návrat k následovnictví.
    - má term menší než můj $\rightarrow$ odmítnu ho.
  - Dlouho nevyhrává nikdo – timeout $\rightarrow$ restart voleb.
- (tabule)

Další úvahy

Volba leadera

- Co když leader má mít nějaké další vlastnosti?
  - Vhodnou polohu/latenci
  - Vyšší výkon
  - Dostatek zdrojů
  - ...
- Co když se síť často mění (ad-hoc, mobilní, ...)?
- Co když síť není spolehlivá?

Algoritmus pro ad-hoc sítě

Volba leadera

- Vasudevan et al., 2004. Řeší úvahy výše.
- Kandidáti jsou nakonec všichni.
- Vybrán ten, který nejlépe plní požadavky.
- Uzel, který chce volby, pošle ELECTION svým sousedům.
- Uzel, který obdrží ELECTION
  - Je to poprvé?
    - nastaví odesílatele jako rodiče,
    - přepošle všem ostatním,
    - čeká na všechna potvrzení, poté sám potvrdí.
  - Už má rodiče? $\rightarrow$ jen potvrdí přijetí
- Vzniká strom.
- Všichni okolo mají rodiče?
  - Jsem list.
  - Rychle dostanu odpovědi.
  - Rychle odpovídám rodiči.
- V potvrzení rodiči nejlepší leader z podstromu.
- (tabule)

Poznámky

Volba leadera

- Split brain problem (nejen u volby leadera).
- Rozdělení sítě $\rightarrow$ v každé části volby $\rightarrow$?
  - (ukázka) Bully algoritmus to nezvládne.
  - (ukázka) Raft to zvládne.

$\rightarrow$ nutná většina hlasů.

- Uvidíme ještě u shody v DS.

Další možnosti

Volba leadera

- Zookeeper
- Chang-Roberts (upravený Ring)
- Gossip protokoly
- Velké sítě
  - Proof of work
  - Proof of stake

Chord systém

Chord systém

- $\approx$ Distribuovaná hashovací tabulka
- Klíč-hodnota (klíč-server obshaující hodnotu)
- Peer-to-peer, ring
- Máme m bitové klíče (2^m klíčů).
- Uzly mají unikátní *id* z téhož prostoru.
- Záznam s kličem k je uložen na uzlu s nejmenším *id* větším než k ($succ(k) = id$).
- Úkol: na dotaz k najít v systému $Succ(k)$ (uzel s hodnotou k)
- Triviálně uzel *id* zná následovníka ($Succ(id + 1)$)
  - Pak pokud hodnotu neznám já, posílám dále.
  - Ale to je pomalé ($O(n)$)

Chord systém

- Lze řešit rychleji – přidáme zkratky
- Uzel má tzv. *Finger Table* (FT_{id}) – vyhledávací tabulka (velikost omezená m)
- $FT_{id}[i] = Succ(id + 2^{i-1})$
- Dotaz na k musíme přeposlat někomu před $Succ(k)$ ale co nejblíže:
  - $FT_{id}[j] \leq k \leq FT_{id}[j+1]$
  - Modulární aritmetika pro udržení v kruhu.
- Mají-li FT_{id} jen jeden záznam $\rightarrow$ triviální řešení.
- (tabule)

Changelog