

Heartbeat Protokollen

Bachelorprojekt i Datalogi 2015 - Synopse

Erik Allin – smt504

Dennis Olesen – cwb759

23. februar 2015

Problemformulering

Er det muligt at konstruere et bibliotek der leverer ressourcehåndtering for et high-availability system med minimalistisk opsætning?

I hvilken grad kan et sådan system håndtere fejl på de tilsluttede maskiner?

Afgrænsning af problemformulering

Som udgangspunkt, så er målet med det implementerede program, at det som udgangspunkt skal kunne afvikles på et lokalt netværk.

Systemet vil som udgangspunkt, fokusere på følgende punkter: Leader-election, logging samt load-balancing.

Desuden vil fokus i højere grad ligge på, at holde styr på liveliness og aktivitetsfordeling, end egentligt datahåndtering eller bearbejdning.

Begrundelse

Emnet er interessant, da man i forbindelse med udvikling af et distribueret system vil være afhængig af, at kunne sikre sig konsistens og høj tilgængelighed af det involverede system. Metoder til at opdage når en maskine fejler, samt derefter at foretage de nødvendige fejl-procedurer, vil altså være, at finde i størstedelen af de distribuerede systemer.

Ved at konstruere et bibliotek, som gør det simpelt for brugeren at implementere disse metoder vil der både kunne spares udviklingstid for fremtidige projekter, der omhandler distribuerede systemer, og være et fælles produkt at kunne optimere og opdatere, fremfor at skulle tage hensyn til hvert enkelt nye system. Endvidere forventer vi, at man i forbindelse med dette samtidig kan opretholde en balancering af systemernes ressourcer, i forbindelse med de regelmæssige liveliness tjek.

Et sådant bibliotek vil herved kunne sikre fremtidige applikationer der bruger distribuerede systemer, en fælles og gennemtestet implementation til både load-balancing og liveliness tjek af deres systemer.

Arbejdsopgaver

1. Analysere heartbeat-protokollen.

Produkt: Analyse-afsnit.

Ressourcekrav: Fundet mindst to eksempler på implementationer af heartbeat-protokollen.

Projekt-interne afhængigheder: Ingen

Tidsforbrug: 4-5 dage

Deadline: 26/02-2015

2. Opstille en API for vores bibliotek.

Produkt: En API for vores produkt, samt en beskrivelse af modulerne.

Ressourcekrav: Analyse for system funktionalitet og opsætning.

Projekt-interne afhængigheder: Ingen

Tidsforbrug: 2-3 dage

Deadline: 05/03-2015

3. Implementering

Produkt: Vores bibliotek.

Ressourcekrav: Vores API, samt bagvedliggende analyse.

Projekt-interne afhængigheder: Ingen

Tidsforbrug: 15-25 dage

Deadline: 19/4-2015

3.1 Connection / Discovery:

Produkt: Forbindelse til lokale servere via broadcasting.

3.2 Liveliness:

Produkt: Periodiske heartbeats, tjek af status for systemer, der tidligere svarede på broadcast fra punkt 3.1

3.3 Leader election:

Produkt: Håndtering af ledere, hvis liveliness viser sig, at forhindre en nuværende leder i at kontakte resten.

3.4 Logging:

Produkt: Replikerede logge til vedligeholdelse af konsistens.

3.5 Load balancing:

Produkt: Opdatering af hvilke servere næste forespørgelse går til, afhængigt af en belastningsrapport givet ved de periodiske heartbeats.

4. Dokumentation

Produkt: En dokumentation for det udviklede bibliotek.

Ressourcekrav: Det implementerede produkt.

Projekt-interne afhængigheder: Ingen

Tidsforbrug: 2-3 dage, men sammenløbende med implementationer.

Deadline: 28/04-2015

5. Afprøvning

Produkt: Et forhåbentligt fejlfrit produkt.

Ressourcekrav: Det implementerede produkt.

Projekt-interne afhængigheder: Ingen

Tidsforbrug: 3-4 dage

Deadline: 07/05-2015

6. Brugervejledning

Produkt: Vejledning til produktet.

Ressourcekrav: Dokumentationen for produktet.

Projekt-interne afhængigheder: Ingen

Tidsforbrug: 3-5 dage

Deadline: 21/05-2015

7. Rapport

Produkt: Rapport af projektet.

Ressourcekrav: Al udarbejdet materiale

Projekt-interne afhængigheder: Ingen

Tidsforbrug: Løbende under hele kurset

Deadline: 28/05-2015

Tidsplan

	Uge 1/2	Uge 3/4	Uge 5/6	Uge 7/8	Uge 9/10	Uge 11/12	Uge 13/14	Uge 15/16	Uge 17/18
Synopse									
Analysere heartbeat									
Opstille API									
Implementering									
Dokumentation									
Afprøvning									
Brugervejledning									
Rapport									
Ekstra tid									
Datoer	2/2 – 15/2	16/2 – 1/3	2/3 – 15/3	16/3 – 29/3	30/3 – 12/4	13/4 – 26/4	27/4 – 10/5	11/5 – 24/5	25/5 – 7/6
Mulige forsinkelser				OSM-eksamen	Påskeferie	Skemafri uge			
Projektfrister		Synopse				Midtvejsrapport			

Evt. metodiske overvejelser, relevant litteratur o.lign

<https://www.usenix.org/system/files/conference/atc14/atc14-paper-ongaro.pdf>

- Beskrivelse af Raft consensus algoritmen - til leader election og logging.

<http://research.microsoft.com/en-us/um/people/lamport/pubs/time-clocks.pdf>

- Beskrivelse af logiske klokker til at sikre konsistens.