

Databasparadigm för hantering av videoinformation

POPULÄRVETENSKAPLIG SAMMANFATTNING **Eric Ottosson, Suraj Sinha**

Relationsdatabasen har länge varit det dominerande paradigmet inom inbyggda system. En uppsjö av nya databasparadigm ställer dock frågan om det i dagsläget hade varit möjligt att avvika från industristandarden. Finns det något att vinna i att byta paradigm?

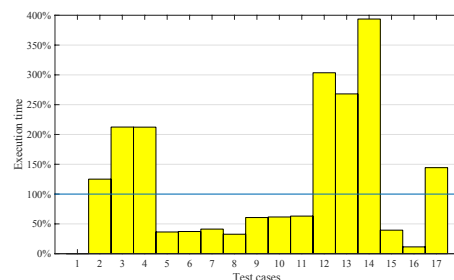
Inledning

Hos inbyggda system i dagens mjukvaruindustri är den så kallade relationsdatabasen en industristandard. För den storskaliga datainsamling som pågår inom områden som videoövervakning, social media etc. så stöter dessa databaser på problem när de utsätts för stora belastningar. Att data dessutom är under ständig förändring gör det också svårt för dessa databaser, eftersom de har begränsad flexibilitet. Utifrån detta har ett flertal databaser, under det gemensamma namnet NoSQL, utvecklats för att tillgodose dessa behov. Många av dessa riktar sig mot tjänster såsom Facebook, Twitter etc. men det vore intressant att även se om dessa typer av databaser har användningsområden inom videoövervakning och inbyggda system. Det vore trevligt om man kunde byta ut en relationsdatabas mot en mer flexibel NoSQL databas utan att förlora prestanda. Ett större utbud av användbara databaser vid utveckling av inbyggda system skulle också vara positivt.

Resultat

Det vi kom fram till med vårt arbete var att det är möjligt att använda en NoSQL databas i en övervakningskamera, men att urvalet är väldigt begränsat. Det visade sig att ett byte från en relationsdatabas till en dokumentdatabas i vissa fall kunde leda till ökad prestanda, men att dessa databaser inte är särskilt lämpade för inbyggda system och de krav som ställs på dessa. Till exempel så visade det sig att dokumentdatabasen EJDB var betydligt snabbare än relationsdatabasen SQLite på att lägga in (0.002s vs 1.8s) och ta bort (1s vs 9s) inspelningar i en databas. Dock så hade EJDB större resursförbrukning än SQLite, och i kameran så växlade EJDBs minnesförbrukning mellan 1-12% medan SQLite låg mellan 1-3%. EJDBs filstorlek var också dubbelt så stor (10MB) som SQLite. Dessutom riskerade EJDB att

förlora inspelningar och bli korrupt vid strömavbrott på grund av hur den sparade information. Så även om det går att använda NoSQL i en övervakningskamera så är de databaser som finns i dag inte särskilt lämpade på grund av hållbarhetskrav och begränsade resurser.



Denna figur visar hur exekveringstiden för EJDB förhåller sig till SQLite i de olika testfallen. Över 100% motsvarar en långsammare tid, och under 100% en snabbare tid.

Utvärderingsmetod

Det här examensarbetet, som utfördes på Axis Communications i Lund, fokuserade på att utvärdera en relationsdatabas mot en NoSQL databas. För att kunna välja en NoSQL databas var det viktigt att ha en bra bild över vad för information det är man vill spara, samt hur denna används. Detta ledde till en utvärdering av datamodellen i en av Axis övervakningskameror. Förutom detta så etablerades ett antal krav som skulle hjälpa oss att utesluta databaser som inte gick att använda på kameran. En undersökning av NoSQL databaser som kunde användas visade att det inte fanns särskilt många databaser som lämpade sig för användning inom inbyggda system, och endast ett fåtal av dessa erbjöd samma funktionalitet som en relationsdatabas. Efter att ha valt ut en databas så designades ett antal testfall baserat på hur Axis använder sin datamodell. Därefter utfördes mätningar av exekveringstider och resursförbrukning (CPU och minne) för dessa testfall.