



Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Szuperskalázható modellezés

Szárnas Gábor (U944EQ), I. évf., MSc mérnök inf. szakos hallgató
Konzulens: Izsó Benedek, Ráth István, Varró Dániel, MIT
Szolgáltatásbiztonságra tervezés szakirány
Önálló laboratórium 1. összefoglaló
2011/12. II. félév

A féléves munkám egy hosszabb projekt része, amelynek célja egy elosztott modelltároló rendszer kialakítása. A modellvezérelt tervezés egyre szélesebb körű alkalmazásának következménye, hogy a modellek komplexitása egyre nő, mind a kézzel készített, mind a generált modellek esetén.

A növekedésnek jelenleg komoly határt szab, hogy a modelltároló rendszerek csak egyetlen számítógépen képesek működni, ezért csak korlátozott méretű modellek tárolására alkalmasak. A problémára egy olyan horizontálisan skálázható rendszer jelentené a megoldást, amely újabb számítógépekkel bővítve egyre nagyobb méretű modellek tárolására lenne képes. Az elosztott tárolás mellett megfogalmazódik a modell párhuzamos feldolgozásának és biztonságos tárolásának igénye is.

A rendszer alapját egy adatbázis-kezelő rendszer képi, amely képes az elosztott működésre. Az elmúlt években a jól bevált relációs adatbázis-kezelők mellett tucatjával jelentek meg ún. NoSQL elosztott adatbázis rendszerek. Ezek specializált architektúrájukkal a Web 2.0-s alkalmazások igényeinek egy-egy aspektusát próbálják minél hatékonyabban lefedni, a relációs adatbázisokhoz képest gyakran egyszerűbb adatmodellt, gyengébb konzisztenciát, de magasabb rendelkezésre állást, jobb skálázhatóságot nyújtanak. A NoSQL adatbázis rendszerek gyakran egy-egy használati esetre jellemző terhelésre optimalizáltak. A modellalapú tervezőrendszerek tipikus felhasználásai: a modell eseményvezérelt szerkesztése, frissítése, modelltranszformációk és jólfomáltsági validációk futtatása.

A félév során a feladatom a jelenlegi elosztott adatbázis-kezelő rendszerek vizsgálata és egy alkalmas rendszer kiválasztása az elosztott modelltároló megvalósításához és használati eseteinek hatékony kiszolgálásához. A rendszerek összehasonlítása során a rendszerek programozói interfészét, replikációs képességét, licenzelési lehetőségeit vizsgáltam meg. A modellvezérelt fejlesztés gerince gráfok hatékony mintaillesztése és gráftranszformációk futtatása, ezért részletesebben a HyperGraphDB és a Neo4j gráfadatbázisokat vizsgáltam meg.

A HyperGraphDB rugalmas, hipergráf alapú adatmodellt nyújt. Képes az egyes csomópontokban Java objektumok (POJO-k) tárolására. A gráf elemzése ugyanakkor nehézkes, mert a rendszer lekérdezőnyelve még nem készült el.

Konkrét méréseket a Neo4j rendszeren végeztem, amely 2007-ben jelent meg és jelenleg a legnépszerűbb elosztott gráfadatbázis. Csak hagyományos irányított gráfok tárolását teszi lehetővé, azonban két lekérdezőnyelvet is nyújt: a Gremlin útbejárások, a Cypher mintaillesztések megfogalmazását teszi lehetővé. Mindkét nyelv folyamatos fejlesztés alatt áll.

Az implementáció előzménye az Izsó Benedek és Ráth István által készített „TrainBenchmark” teszt-sorozat, amely különböző Java- és RDF-alapú tárolórendszerekbe töltött EMF modelleken futtat jólfomáltsági kényszereket kiértékelő lekérdezéseket [1].

Az implementáció közben kiderült, hogy a Neo4j önmagában nem támogatja POJO-k tárolását, ezért az EMF modellről a gráf adatmodellre történő leképzést a Spring Framework alkalmazás keretrendszer Spring Data adatelérési rétegével végeztem el. A jólfomáltsági kényszereket a Cypher lekérdezőnyelven készítettem el. A mérések eredménye, hogy a Neo4j sebességében némileg elmarad az Eclipse OCL-ben és jelentősen, 2-3 nagyságrenddel az EMF-IncQueryben [2] végzett lekérdezésektől.

Az elosztott rendszerben történő működés vizsgálatát nyáron tanszéki szakmai gyakorlaton, ősszel TDK-témaként tervezem végezni.

[1] <https://trac.inf.mit.bme.hu/Ontology/wiki/TrainBenchmark/>

[2] <http://viatra.inf.mit.bme.hu/incquery/>