



Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Inkrementális lekérdezés-kiértékelés az i3QL technológiával



Lakó Attila III. évf, (BSc) mérnökinformatikus szakos hallgató
Konzulens: Bergmann Gábor, MIT
Beágyazott információs rendszerek, Informatikai technológiák
szakirány/Rendszertervezés ágazat
Önálló laboratórium. összefoglaló
2014/15. II. félév

Napjainkban a szoftverfejlesztés terén egyre nagyobb szerepet kap a modellvezérelt tervezés, mely során a modelleken különféle tesztek és ellenőrzések sokaságát hajthatjuk végre automatikusan, még a kész szoftver előállításá előtt. Ezzel a célberendezésbe vagy végfelhasználóhoz kerülő szoftver minőségét növelhetjük, a költségek csökkentése mellett.

A modellvezérelt fejlesztési folyamatok során a modellek automatizált feldolgozása támogatható a modellek felett deklaratívan megfogalmazható lekérdezésekkel. Kulcskérdés a lekérdezések kiértékelésének hatékonysága, ezért a különböző alkalmazási helyzetekhez többféle megközelítésű végrehajtó motor létezik. A hagyományos megoldások keresés alapján szolgáltatnak eredményt minden lekérdezéskor, míg az ún. inkrementális algoritmusok előre kiszámolt válaszokat szolgáltatnak, amelyeket azonban a modell minden változásakor karban kell tartaniuk.

Inkrementális feldolgozást támogat többek között a Scala alapon megvalósított i3QL technológia. Az i3QL deklaratív SQL-szerű szintakszist biztosít, ami a relációs algebra műveleteinek az inkrementális változatán alapul. Könyvtárként van integrálva a Scala-ba, ami lehetővé teszi a programozónak, hogy hagyományos Scala kódot használjon egy i3QL lekérdezés nem-inkrementális mellékszámításaihoz valamint, hogy könnyedén integrálhassa az inkrementális számításokat egy nagy projektbe. Teljesítmény növelés érdekében az i3QL optimalizálja a felhasználó által definiált lekérdezéseket algebrai törvények és részleges kiértékelés alkalmazásával.

Az i3QL egy bizonyos LMS (Lightweight Modular Staging) keretrendszert használ, aminek az alap ötlete, hogy magas szintű, generikus programból futási időben speciális, extrém gyors alacsony szintű kódot generál.

A félév során az első feladat az i3QL beüzemelése és kipróbálása volt. Mivel se a readme, se a dokumentáció nem jelezte, hogy az LMS melyik verziójával működőképes a Master verziója, ezért több próbálkozás után csak egy alternatív, újabbat tudtam csak működésre bíni. Fejlesztő környezetnek az IntelliJ IDEA-t választottuk az SBT támogatása miatt. Az i3QL projektjeinket SBT-vel (Single Build Tool) fordítjuk, ami egy olyan scala fordító, amely egy project mellett megadott build.sbt fájlban szereplő könyvtárakat képes beimportálni automatikusan. Ilyen könyvtár az i3QL és az LMS is.

További kitűzött feladatok a demó projektek kipróbálása valamint saját meta modell és lekérdezések létrehozása volt. Egyik megvalósítása se volt zökkenőmentes. A demók egy része nem fordult. A dokumentáció és a működő demók alapján kezdtem neki a saját kód készítésének. A probléma az volt, hogy az LMS keretrendszer megköveteli, hogy a saját típusainkon való egyes műveleteket át emeljünk Rep[] típusúvá, ami az LMS egy típusa a futási idejű feldolgozáshoz. A működő demó meta modellje az i3QL könyvtárba volt belefördítve a Rep típusú műveleteivel együtt ezért bárhol elérhető, mivel egy külső projektből nem módosíthatjuk ezért ez nem volt járható út. A készítővel való levelezés után egy egyszerű megvalósítást használva, melynek csak egy műveletét kellett Rep típusúként megvalósítani és nem volt szükséges az i3QL könyvtárba elhelyezni, sikerült tetszőleges metamodelleket létrehozni és hozzá lekérdezéseket írni.

Sajnos a felmerülő problémák miatt a technológia teljesítményének lemérése a további feladat maradt, amelyet a tanszéken fejlesztett Train Benchmarkkal tervezek megvalósítani, amelyet kifejezetten inkrementális lekérdező technológiák teljesítménymérésére készítettek.