

Kényszerek kezelése ViatraDSM alatt

Önálló laboratórium feladat összefoglalója (2. félév)

Ökrös András (CYJXHJ)

Konzulens: Ráth István

**BME Méréstechnika és Információs Rendszerek Tanszék
Informatikai infrastruktúra tervezése szakirány, 2006/2007. II. félév**

A modellvezérelt rendszerfejlesztés napjainkban egyre népszerűbb, az iparban széles körben elterjedt szoftverfejlesztési módszertan. A manapság elterjedt modellvezért módszertanok, mint például az MDA, jellemzően az UML nyelvet használják a tervezés során. Több probléma is akad az UML-lel, sok alkalmazáshoz nehézkes a használata; az UML Profile-ok nem eléggé szigorúak; túl általános így nehézkes belőle a kódgenerálás is. Továbbá, míg a rendszertervezők számára az UML eszközök nagyjából megfelelőek, ám egy adott terület szakértőjének ezek az eszközök túl általánosak és nehezen kezelhetőek.

Egyelőre úgy tűnik, hogy a problémára a legjobb megoldást a domain specifikus modellezési nyelvek jelentik. Ennek lényege, hogy az általános célú UML modellek helyett / mellett döntően olyan nyelvet használunk, mely jól illeszkedik a tervezni kívánt rendszer követelményeihez, alkalmazási környezetéhez.

Bár az ipari igény nagy rájuk, és a terület is dinamikusan fejlődik, a domain specifikus modellezési nyelveket előállító eszköztervező nem igazán tud kiforrott fejlesztőeszközök közül választani. Az ilyen eszközök egyik legfontosabb hiányossága, hogy a nyelv specifikus kényszerek kezelése nem megoldott bennük.

A féléves munkám során megismerkedtem a tanszéken fejlesztett eszközökkel, amelyek domain specifikus modellezési nyelvek tervezéséhez használhatók. Ez magában foglalta a Viatra és a ViatraDSM rendszerek megismerését is. A félév során a programok „tanulása” mellett a kényszerek kezelésével foglalkoztam. Milyen kényszereket tudunk vizsgálni, ezeket hogy érdemes csoportosítani. A kényszerek ellenőrzését szerkesztés közben szeretnénk megvalósítani, így a kényszerek kiértékelésének a sebessége is fontos lesz számunkra a későbbiekben.

Egy példakényszert alaposabban is megvizsgáltam. Petri hálók token kapacitáskorlátjával foglalkoztam, ennek a kényszernek az ellenőrzéséhez létrehoztam egy viatra parancsfájlt (.vtcl fájl), amely bizonyos konvenciókat betartva bejárata a modellteret, és a kényszert megszegő elemeket megjelölte. A munka másik fele a ViatraDSM kiegészítése volt, hogy a felhasználó a szerkesztési felületről el tudja indítani az ellenőrzést a már létrehozott petri hálókön.

A távlati cél az on-the-fly, vagyis a szerkesztés közbeni kényszerkiértékelés megvalósítása a ViatraDSM tanszéki rendszerhez.