

Domain specifikus modellezési nyelvek vizsgálata

Önálló laboratórium feladat összefoglalója (1. félév)

Ökrös András (CYJXHJ)

Konzulens: Ráth István

**BME Méréstechnika és Információs Rendszerek Tanszék
Informatikai infrastruktúra tervezése szakirány, 2005/2006. I. 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. Az ilyen magas szintű modellekből modelltranszformációval állítjuk elő a platform-specifikus megvalósítást - ennek gyakran alkalmazott módja a kódgenerálás.

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. Épp ezért e féléves önálló laboratóriumi munkám céljából azt tűztem ki, hogy megismerkedjek a rendelkezésre álló eszközökkel, felmérjem milyen szolgáltatásokat kínálnak, mennyire könnyen integrálhatók a jelenlegi fejlesztési környezetekbe.

A féléves munka során a következő ipari eszközöket vizsgáltam meg: MetaCase MetaEdit+, Microsoft DSL Tools, illetve az Eclipse EMF - GEF / GFM „technikák”. Egy próbapéldát valósítottam meg mindegyikben, majd ennek tapasztalatai alapján alkottam egy szempontrendszert, és értékeltem a programokat. A szempontok a következők voltak: Nyelv specifikus jóformáltsági kényszerek kezelése, szimuláció támogatása, modell analízis (kód generálás előtt), modelltranszformációk támogatása, multi domain modellezés, integrálás meglévő fejlesztő rendszerekhez. Két szempont kivételével egyikhez sem nyújtottak megfelelő támogatást a szoftverek. A nyelv specifikus kényszerekre egyedül a MetaCase nyújtott támogatást, bár ez a szoftver a meglévő rendszerekhez történő integrálás során maradt el a többitől. A Microsoft DSL Tools a Visual Studio része, így integrálása meglévő fejlesztési környezethez nagyon jó (feltéve, ha Visual Studiot használjuk). Az Eclipse eszközök természetesen kitűnően vannak integrálva a népszerű keretrendszerbe.

Összefoglalva, a tanulság az volt, hogy a jelenlegi modellezési nyelv fejlesztő eszközök sok hiányossággal rendelkeznek. Céлом így a további félévekre, hogy a tanszéki domain specifikus modellezési támogatás nyújtó eszközt, a ViatraDSM-et, kiegészítsem olyan funkciókkal, amelyet az ipari szoftverek nem, vagy alig támogatnak; ilyen például a szintakszisvezérelt szerkesztés összetett kényszerek alapján.