



Budapesti Műszaki és Gazdaságtudományi Egyetem  
Méréstechnika és Információs Rendszerek Tanszék

## Modellalapú regressziós tesztelő eszköz fejlesztése

**Molnár Gábor II. évf, (MSc) mérnökinformatikus szakos hallgató**  
**Konzulens: dr. Micskei Zoltán Imre, adjunktus, MIT**  
**Szolgáltatásbiztos rendszertervezés szakirány**  
**Diplomaterv 1. összefoglaló**  
**2014/15. II. félév**

Regressziós tesztelés esetén azt ellenőrizzük, hogy a módosított programban továbbra is működnek-e a korábbi funkciók. Ez nagy rendszerek esetén akár órákig vagy egy napig is eltarthat. Egy felfedett hibáról így csak későn értesülnek a fejlesztők és a javítása is költséges (legalább az új kiadás elkészítése).

A teszt optimalizálás feladata ennek a futásidőnek a csökkentése, illetve a visszajelzés gyorsítása. Többek között ilyen módszer a teszt kiválasztás, amikor a verziók közötti változtatások alapján meghatározzuk azokat a teszteket, amelyek ellenőrzik ezeket a módosításokat, illetve melyek azok a tesztek, amelyek nem kapcsolódnak egy változtatáshoz sem, így elhagyhatók.

Az önálló laboratóriumi munkám során elkészítettem egy modellalapú teszt optimalizáló eszköz prototípusát és a teszt kiválasztó modult, így a diplomatervemben ezt folytattam. A feladat tehát az eszköz által értelmezhető tesztmodell építése volt.

A hosszabb távú tervek között az alábbi bemenetek támogatása szerepelt:

- kód fedési jelentés tesztmodellé alakítása,
- teszt generáláshoz használt modellek átalakítása,
- forráskód alapján történő modellépítés.

Ebben a félévben az első két feladattal foglalkoztam.

A legtöbb code coverage eszköz képes XML formátumba exportálni a jelentését, így amennyiben ez kellő részletességgel tartalmazza a teszt-alkalmazáskód viszonyokat (contributed coverage), úgy felhasználható teszt kiválasztáshoz. A félév során a Java-s Apache Cobertura illetve az EclEmma-hoz kapcsolódó JaCoCo eszközöket vizsgáltam meg.

Ezek sajnos nem produkáltak kellő részletességű jelentéseket. Így a munkát a .NET-es OpenCover<sup>1</sup> eszközzel folytattam, amely jelentései a köztes kód szintjén tartalmazzák az elvárt relációt. Az OpenCover XML jelentését egy Java-s SAX alapú parserrel dolgoztam fel, majd a korábban definiált EMF modellre alakítottam.

A második feladatrész azért érdekes, mert itt alkalmazható egy egyszerű modelltranszformáció is, így már a modellezés során kialakítható a rendszer tesztmodellje is. A bemeneti modelleket a tanszék az R3-COP projekt során fejlesztett teszt generáló eszköze szolgáltatja, így regressziós tesztelés alkalmazásával azonosíthatóak, hogy mely teszteket szükséges újragenerálni egy (modell)módosítás esetén. Továbbá az eszköz képes azonosítani azokat az elemeket, amelyekhez még nem készültek tesztesetek, ezzel is csökkentve az újragenerálási időt.

Utóbbi feladat megoldása nem várt nehézségekbe ütközött, pl. mivel a határérték-analízis szempontjait is figyelembe kell venni a tesztek kiválasztásakor (pl. egy él be van-e húzva vagy nincs). Valamint a nem triviális annak eldöntése, hogy a módosítás érvényteleníti-e a korábban generált teszteket.

A következő félévben így a továbbra is a modellbemenetek feldolgozása és a forráskód parse-olása lesz a feladat. Utóbbi vélhetően az Eclipse JDT vagy MoDisco könyvtárával lesz megvalósítva.

---

<sup>1</sup> <https://github.com/OpenCover/opencover>