



Budapesti Műszaki és Gazdaságtudományi Egyetem  
Méréstechnika és Információs Rendszerek Tanszék

## Modell-alapú tesztelés

**Verbőczy Kristóf III. évf, (BSc) mérnökinformatikus szakos hallgató**  
**Konzulens: dr. Micskei Zoltán adjunktus, MIT**  
**Informatikai technológiák szakirány/Rendszertervezés ágazat**  
**Önálló laboratórium összefoglaló**  
**2015/16. II. félév**

A tesztelés fontos része a programok fejlesztésének. Különböző módszerek léteznek a tesztelésre, viszont ezek nagy része implicit, azaz a tesztelő fejében vannak csak meg a tesztesetek. Ezen a szemléleten változtat a modell-alapú tesztelés, a maga explicit nézetével.

A modell-alapú tesztelés alapvető működése a következőképpen néz ki. Egy specifikáció (legtöbb esetben egy informális szöveges leírása a követelményeknek) alapján elkészül az implementáció, ami az eseteket nagy részében programkódot jelent. Ugyanebből a specifikációból készül el a modell is. A modell lehet strukturális (pl.: osztálydiagram) vagy viselkedési (pl.: állapotdiagram). A modell alapján lehetőség van a tesztek származtatására. Ez történhet kézzel, de lehetőségünk van automatizálásra ebben a fázisban. Ezen tesztek absztraktabbak, mint az elkészült program, ezért szükség van valamire, hogy átalakítsa a tesztesetek olyan formára, hogy ellenőrizni lehessen vele a kód helyes működését.

A modell-alapú tesztelés számos motivációval rendelkezik, ezek közül néhány:

- újrafelhasználhatóság – ugyanabból a modelltől többféle tesztesetet lehet generálni, így egy modell, különböző teszt célokra vagy teszt típusra is használható
- automatizálás – növeli a hatékonyságot; csökkenti a manuálisan vétett hibák számát, ami egyébként nagy költségekkel járna
- hiányosságok felfedezése a fejlesztés korai szakaszában – mivel már a fejlesztés korai szakaszában el lehet kezdeni a modell készítését és a tesztelést, így a korai szakaszban lehetséges a hibák és hiányosságok felfedezése
- a karbantartás költsége kisebb – mivel a modell sokkal könnyebben módosítható, bővíthető, mint maga a rendszer

A féléves munkám során különböző cikkek alapján megismertem a modell-alapú tesztelés alapjait.

Annak érdekében, hogy az elméleti tudásomat mélyítsem egy esettanulmányt készítettem. A feladat az volt, hogy valósítsam meg Java nyelven a TCP protokoll kapcsolatfelvételt és lebontását, majd a modell-alapú tesztelésnek megfelelően ellenőrizsem le.

Az implementációt elkészítettem, majd a modellt is. Utóbbihoz a Yakindu Statecharts nevű modellező eszközt használtam. A modell alapján kézzel származtattam a teszteseteket a modelltől. A teszteseteket szintén kézzel alakítottam át Java-nak megfelelő programkódra.

Ezek után kipróbáltam egy modell-alapú tesztelésre kifejlesztett eszközt, a GraphWalker-t. Az implementáció ugyanaz maradt. Viszont a modellt át kellett alakítanom, ehhez a yEd Graph Editor nevű programot használtam. Ennek az eszköznek az előnye, hogy automatikusan generál tesztesetek a modelltől különböző megadott feltételek alapján. Ezen kívül még JUnit tesztvázakat is generál, ezzel megkönnyítve a tesztesetek átalakítását programkódra.

A félév során megismerkedtem a modell-alapú tesztelés alapjaival, és kipróbáltam egy egyszerűbb példán a gyakorlatban. A továbbiakban még ezen példán szeretném kipróbálni, hogy a Yakindu modelltől UPPAAL modellt generálok és ezen modellen temporális logikai kifejezések segítségével keresek gráf bejárásokat, azaz teszteseteket.