

# **RCP alkalmazások átültetése webes környezetre**

## **Önálló laboratórium feladat összefoglalója**

**Farkas Péter (HOE1F4)**

**Konzulens: Dr. Varró Dániel**

**BME Méréstechnika és Információs Rendszerek Tanszék  
Rendszertervezés ágazat, 2008/2009. II. félév**

Az önálló labor feladatomban az volt, hogy megismerkedjem korszerű webes technikákkal és megvizsgáljam, hogy milyen lehetséges egy már meglévő vastagkliens alkalmazást átültetni webes környezetre. A vastagkliens a tanszéken kifejlesztett VIATRA2 modell-transzformációs keretrendszer volt, mely Eclipse Rich Client Platform-n (RCP) készült.

A félév elején megismerkedtem jó pár technológiával webes- és vastagkliens alkalmazások terén. A webes az Eclipse Rich Ajax Platform (RAP) volt, melynek elsődleges célja, hogy RCP szerűen lehessen webre is fejleszteni. Rekonstruáltam a példa programokat. Ezek többsége nem haladtam egy „hello world” színvonalát, viszont voltak összetettebbek is. Amik jó kiindulási alapot nyújtottak, hogy tovább fejlesztve, tanulmányozhassam a számomra fontos részeket.

Ennek eredményeképpen elkészítettem egy könyvtárkezelő webes programot. A program egy fájlfastruktúrában (treeview) megjeleníti a kiválasztott partícióban található könyvtárakat. Maga a RAP-os komponens is támogatja a modell lassú betöltését (lazy load) kliens oldalon, de a programban található modell is lassú betöltéssel épül fel a szerveren.

A félév második felébe felmértem, hogy milyen lépéseket kell végrehajtani ahhoz, hogy webes alkalmazás formájában is a feladat lehessen használni a VIATRA2-t. Sajnos a RAP API (még) nem tartalmaz mindent, amit az RCP API-ja. Így ki kellett gyűjtenem a meglévő forráskódból, hogy mik nincsenek meg.

A gyűjtés azért volt kicsit komplikáltabb, mert nem csak a konkrét használni kívánt osztályok, hanem sok esetben előfordult, hogy az őseik, vagy olyan interfészek / osztályok, amiket használnának sem találhatók meg RAP-ban. Szépen sorban haladtam a hibákkal, kikértem, az RCP számára fent tartott CVS-ből az eredeti komponenst és a függőségeit is. Kikommenteztem a függvények törzsét, így létre hozva egy „skeletont”.

Még ez sem volt elég, bele kellett nyúlni az eredeti VIATRA2 kódba is. Például nem egyezik (teljesen) az erőforrás kezelés a két „módszerben”. Ezért, hogy könnyen észre lehessen venni és meglehessen találni minden esetben a változtatás elé beírtam, hogy „`//begin fape`”, a végére „`//end fape`”. Amelyik sorra nem volt szükség azt nem töröltem, hanem csak kikommenteztem.