

# Szkript nyelvi környezet fejlesztése űrszonda szimulátorhoz

## Önálló laboratórium feladat összefoglalója (1. félév)

**Mátyás Zsolt (VHC4LO)**

**Konzulens: Tróznai Gábor, Micskei Zoltán**

**BME Méréstechnika és Információs Rendszerek Tanszék  
Informatikai infrastruktúra tervezése szakirány, 2005/2006. I. félév**

Az önálló labor keretében az RMKI lehetőséget biztosít arra, hogy az egyetemi hallgatók az Intézet úrkutatással és űrtechnikai fejlesztésekkel kapcsolatos eredményeit megismerhessék, s hogy a tehetséges és különös elhivatottságot érző hallgatók a kutatási és fejlesztési munkába bekapcsolódhassanak. A ROSETTA az ESA (European Space Agency) egyik folyamatban lévő, üstökös-kutatásra irányuló missziója. A szonda 2004-ben indult és 2014-ben éri el a Churyumov-Gerasimenko üstökösöt, ahol az orbiter és leszállóegység különválik, és előbbi üstökös-körüli pályán, utóbbi az üstökös felszínére ereszkedve kezdi meg a méréseket. Az RMKI, a program egyik magyar résztvevőjeként, a PHILAE leszállóegység vezérlő és adatgyűjtő számítógépét készítette. Jelenleg a fedélzeti számítógép szoftverének karbantartása mellett az intézet végzi a leszállóegység kísérletei és a központi számítógép szoftveres szimulátorának fejlesztését is.

Az önálló labor keretében végzendő munka célja kezdetben szoftver fejlesztése a különböző fedélzeti egységek szimulátorainak vezérlését végző XML és XSL alapú script nyelv fájljainak szerkesztéséhez, kényelmes és hibamentes összeállításához. Az önálló labor keretében végzendő sikeres munkához a C++ programozási nyelv, Borland C++ Builder5 fejlesztőkörnyezet, valamint XML technológia ismerete, vagy ezek gyors megismerésére való képesség, továbbá angol nyelv közép-fokú ismerete szükséges. A munka során a KFKI RMKI-ban személyes konzultációra lesz lehetőség.

A félév során megismerkedtem az elterjedtebb XML séma nyelvekkel (DTD, XSD, Relax NG), összehasonlítottam őket annak érdekében, hogy eldöntsem, melyik felel meg legjobban a teljes feladat végrehajtásához. A megtervezett program így nem csak a KFKI feladatkiírását fogja teljesíteni, hanem más XML fájlokat is létrehozhatunk vele a definiált sémák alapján.

A feladatkiírásban a megvalósításhoz C++ volt specifikálva, de ettől - a konzulensemmel egyeztetve - eltértem. A lehetséges programnyelvek közül, a Java-t találtam legmegfelelőbbnek, mivel a program jelenlegi fázisában a grafikus felület kevésbé dominál, viszont a Java elterjedtségének köszönhetően a lehető legszélesebb körben futtatható lesz a program.

XML megjelenítési módszerek közül szinte kizárólagosan XSLT alkalmazásával találkoztam, amik különböző HTML és szöveges fájl formátumokba generálnak kimenetet. Az általános fa vizualizációk viszont rendkívül széles skálán mozognak. Ezek közül az szokásos csomópont-él megjelenítés bizonyult a leghasználatóbbnak XML struktúrákhoz.

A félév utolsó harmada folyamán megismerkedtem a Java DOM fa szerkezetet létrehozásának egy-egy módszerével, a hozzá tartozó Java osztályokkal, implementációban pedig XML beolvasás, módosítás és kiírás valósult meg. Majd a lehetőségek közül a JAXP API beolvasó, transzformáló és validáló lehetőségeit tanulmányoztam. Ezekhez természetesen szükség volt XPath és XSLT ismereteim elmélyítésére is.