

Felhasználói felületek modell-alapú fejlesztése

Jósvai Eszter V.inf., eszter.josvai@gmail.com

Tóth Dániel V.inf., daniel.toth@gmail.com

Konzulens: Balogh András, Méréstechnikai és Információs Rendszerek Tanszék
abalogh@mit.bme.hu

A számítástechnika, informatika fejlődése nemcsak technológiai előrelépést hozott, hanem a felhasználók köre is meglehetősen kiszélesedett. Kezdetben a tervező, programozó és felhasználó ugyanaz a személy volt. Mára azonban nagyobb százalékban vannak azok, akik csak végfelhasználói a rendszereknek. Mivel ők munkájuk segédeszközeként használják az alkalmazásokat, ezért hangsúlyt kell fektetni a felhasználói felület minél jobb megtervezésére.

A jelenlegi felület-generáló programok többnyire a „drag-and-drop” technikát alkalmazzák. Tehát meghatározott felületelemeket helyezhetünk el üres űrlapokon. Ilyenek a Visual Editor for Java (Eclipse Plug-in), a QT Designer vagy az AUIML Toolkit.

A Visual Editorral és az AUIML Toolkit-tel Java forráskódot, a QT Designerrel xml leírást ill. C++ kódot nyerhetünk ki, amit felhasználhatunk a teljes alkalmazás összeállításánál. Az egyes szerkesztőkkel tehát nyelvhez, platformhoz kötött képernyőleírást hozhatunk létre. Az AUIML Toolkit ábrázolási lehetőségei a másik két eszközzel szemben meglehetősen korlátozottak.

A mi elképzelésünk - a modell-alapú szoftverfejlesztés filozófiáját követve - olyan felület - generáló módszer kifejlesztése volt, amely a fentiekkel ellentétben platformfüggetlen UML modellből kiindulva forráskódot hoz létre. Így egy teljes rendszert leíró UML modellhez - amely későbbi kódgenerálás alapját adja - az alkalmazáshoz tartozó felhasználói felület modelljét is hatékonyan csatolni lehetne.

Másik előnye a modell-alapú megközelítésnek a későbbi változtatások könnyű és gyors bevezetése. Nyilvánvalóan már a tervezés során figyelembe kell venni a használhatósági és célszerűségi szempontokat, de gyakran vita tárgyát képezi a végfelhasználókkal a felhasználói felület kinézete, elrendezése. Így mindössze a modellben kell a kívánt változtatásokat elvégezni, ami jóval egyszerűbb, mint teljes kódrészletek újírása.

A megvalósítás során a VIATRA (Visual Automated Model Transformations) modell-transzformációs keretrendszert vettük igénybe. A szabványos UML XMI formában lévő képernyőmodell ebbe a keretrendszerbe került be, amelyben gráftranszformációs eljárások segítségével generáltuk a kódot. Az általunk definiált képernyőt leíró metamodellbe igyekeztünk az általánosan használatos felületelemeket belefoglalni. Ez alapján készíthetők el a konkrét képernyőket leíró modellek, melyben az események hozzárendelése is lehetséges. Dolgozatunkban néhány jellemző példán keresztül mutatjuk be a módszert és a fejlesztés menetét.

Irodalom:

1. The OMG Model Driven Architecture
<http://www.omg.org/mda>
2. Paulo Pinheiro da Silva and Norman W. Paton: User Interface Modelling with UML
<http://www.cs.man.ac.uk/img/umli/publications/imkb.html>