

C++11 dialektus használatának támogatása örökölt környezetben

Havas Dávid

I. évf. programtervező informatikus MSc

Antal Gábor

II. évf. programtervező informatikus MSc

*Témavezetők: Dr. Siket István, Dr. Beszédes Árpád
SZTE TTIK Szoftverfejlesztés Tanszék*

A szoftverfejlesztésben alkalmazott technológiák napjainkban nagyon gyorsan fejlődnek. A fejlesztők szeretnék minél nagyobb százalékban kiaknázni a fejlesztések adta új lehetőségeket ezzel is megkönnyítve a saját munkájukat, valamint javítani tudják az általuk fejlesztett kód hatékonyságát. Azonban ezeket a törekvéseket gyakran felülírják az ipari partnerek által támasztott feltételek, azaz, hogy milyen platformon kell működképesnek lennie adott terméknek. Ezek a platformok legtöbbször nem a legfrissebb technológiákat alkalmazzák. Továbbá, nem csak egy platformra kell a kompatibilitást biztosítani, hanem számos különböző környezetben kell helytállnia a szoftvernek. Emiatt a szoftverfejlesztő vállalatoknál szigorú fejlesztési irányelveket kell meghatározni, hogy teljesíteni tudják a partnerek által támasztott elvárásokat, ami korlátozza az új technológiák bevezetésének és használatának a lehetőségét. Ez csökkenti a programozók produktivitását és korlátozza a fejlődési lehetőségüket is az adott technológiai területeken.

A korábban említett probléma a C++ programozási nyelvben is megjelenik. A C++-ban új szabvány több mint nyolc éven keresztül nem jelent meg. Azonban a 2011-ben kiadott frissítés nem csak egy egyszerű frissítés volt, hanem akár egy új nyelvnek is tekinthetjük, mert annyi új változást és nyelvi elemet adott hozzá a C++-hoz. C++11-et maga Bjarne Stroustrup, a C++ atyja is egy új nyelvnek tekinti ezen okok miatt. A fejlesztők pedig szeretnék kiaknázni a C++11-ben rejlő lehetőségeket, de az előbbieket felsorolt problémák miatt még a 2011-es verzió megjelenése után öt évvel sem tudják használni az új nyelvi elemeket egyes vállalatok közötti együttműködés miatt. Ezért a C++ programozási nyelven fejlesztett rendszereket vizsgáltunk. Ezen probléma megoldására kidolgoztunk és megvalósítottunk egy olyan forráskód transzformációs keretrendszert, amely lehetővé teszi a C++ 2011-ben kiadott frissítésében (C++11 szabvány) megjelenő számos új nyelvi elem átalakítását úgy, hogy az átalakított kód a C++ 2003-as verziójával (C++03 szabvány) kompatibilis legyen. A keretrendszer nem csak egy-egy forrásfájl átalakítására alkalmas, hanem különböző összetett rendszerek transzformálására is, ezért a fejlesztési folyamatokba történő integrálás segítésére számos egyéb alkalmazást fejlesztettünk, mint például a forráskód könyvtárstruktúra duplikálás, inkrementális transzformálás, és az transzformációk nyomon követése az eredeti forráskódban. Az általunk megvalósított keretrendszer segítségével a fejlesztők használhatják a 2011-es szabványban megjelenő új nyelvi elemek egy részét úgy, hogy továbbra is biztosítva marad a 2003-as szabvánnyal való kompatibilitás.

Az alkalmazás tervezése során különböző megvalósítási lehetőségeket vizsgáltunk meg, melynek eredményeképpen úgy döntöttünk, hogy az LLVM clang fordítórendszer által biztosított infrastruktúrát használjuk fel az eszközünk megvalósításához. A megvalósított alkalmazást különböző méretű rendszereken teszteltük, melyek között megtalálhatók kereskedelmi termékek (1,5 millió, illetve 0,5 millió sor), valamint több nyílt forráskódú alkalmazás is. Továbbá, a tesztelést különböző platformokon is elvégeztük, mert a megvalósítandó alkalmazásnál kiemelt cél volt a platformfüggetlenség biztosítása.