

# **Ipari szoftver rendszerek variabilitás menedzsmentjének javítása**

*Daniel Dan*

*II. évf. Programtervező Informatikus MSc.*

*Témavezető: Dr. Ferenc Rudolf adjunktus*

*SZTE TTIK Szoftverfejlesztés Tanszék*

A mai nagyobb szoftverrendszerek nagy része több verzióval és különböző felhasználói közösségekkel rendelkezik. Ezek a verziók különbözhetnek például egyes alkalmazási lehetőségekben vagy az alkalmazás nyelvében. A fejlesztőknek találniuk kellett egy megoldást arra, hogy bizonyos variabilitási opciókat adjanak a szoftverhez anélkül, hogy mindig újra kelljen azt fejleszteni.

Jelen dolgozat célja a fent említett szoftverek konfigurálásának, személyre szabásának könnyítése. Ilyen szoftverrendszerek megvizsgálása során azt tapasztaltuk, hogy e szoftverrendszerek nagy része (pl. Linux) sok esetben cpp (C preprocessor) utasításokkal (pl. #define, #ifdef, #if parancsokkal) oldják meg a szoftver variabilitásának megalapozását.

A kutatás során egy a fent említett cpp-t kiegészítő cpp+ elnevezésű nyelvet és ezt értelmező és feldolgozó eszköz fejlesztettünk, mely segítségével a szoftverrendszerek variabilitásának kezelése generikusabbá, fejlettebbé és érthetőbbé válik. A cpp+ nyelvfüggetlen, azaz használhatjuk bármilyen szöveges forráskódra, vagy akár dokumentumokra is. A nyelv nagy előnyei közé tartozik, hogy a cpp+-t használó fejlesztők definiálhatnak generikus sablonokat (template) egyes, hasonló programstruktúrák leírására, mint például hasonló osztályok, metódusok, kódrészek. E fejlesztés segítségével drasztikusan csökkenhet a variációs pontok száma a szoftverrendszerben, ezzel segítve a program logikai részének megértését, fejlesztését.

A cpp+ elméleti alapjait a National University of Singapore (NUS) egyetemen fejlesztett XVCL (XML-based Variation Configuration Language) adta. Az XVCL-hez képest a cpp+ független lett az XML nyelv által megszabott korlátoktól, flexibilis szintaxissal rendelkezik, és számos új funkciót, lehetőséget is bevezetett.

A fejlesztett eszközt volt alkalmunk hasznosítani egy esettanulmány során. A Linux operációs rendszer forráskódjának egy kisebb részét – mely az egyes eszközök, perifériák működtetését biztosító forráskódokat tartalmazza – kiegészítettük cpp+ parancsokkal a benne szereplő cpp parancsok nagy részének felülírása segítségével. Eredményképpen a vizsgált kódok mérete drasztikusan csökkent és a rendszer azon része sokkal könnyebben átlátható lett a közös részek, logika kiemelése révén. Egy új, hasonló célú rész fejlesztése során, egy cpp+ reprezentáció segítségével a fejlesztőnek elég azokra a részekre koncentrálnia, amik eltérnek a közös logikától, így a fejlesztés egyszerűbb és átláthatóbb lesz.