

Párhuzamos JavaScript programok memóriafogyasztásának mérése

Horváth Zoltán

Programtervező Informatikus BSc

Napjaink legdinamikusabban fejlődő programozási területét a webes technológiák jelentik. Emberek millió használják az internetet és ezáltal azokat a technológiákat melyek nélkülözhetetlenek a mindennapjaikban mind asztali-számítógépes, mind mobilkommunikációs környezetben. Ilyen internetes szolgáltatások például az online-kapcsolattartás, vásárlás és hírolvasás. Az említett szolgáltatások és minden egyéb weboldal forrásának elengedhetetlen komponensei azok a JavaScript kódok, amelyek a kliens oldali végrehajtásával számtalan lehetőséget biztosítanak a webprogramozó számára, továbbá a felhasználónak kényelmesebbé teszik a szolgáltatások használatát. A JavaScript kód futtathatósága utáni igény magával vonja, hogy minden komolyabb böngésző alapszolgáltatásként biztosítsa azt. A mobiltelefon fejlesztő cégek - például a Nokia is -, igyekeznek a legtöbbet profitálni a web alapú szolgáltatások mobil környezetben való kiaknázásából, melyek futtathatósága nem lehetne lehetséges a webböngészők használata és ezáltal a JavaScript programok végrehajtása nélkül.

A technológia fejlődése miatt napjaink asztali hardware-rei szinte kivétel nélkül többmagos processzorral és tág korlátokat adó memóriával rendelkeznek. Ezzel szemben a beágyazott rendszerekben - például a mobiltelefonokon -, mind a processzormagok száma, mind a memória és rendelkezésre állási idő is korlátozott. A hardware nyújtotta szűk keretek ellenére a felhasználók elvárásai mégis hasonlóak az asztali környezetben megszokotthoz. Felmerül a kérdés: kielégíthetők-e ezek az igények? Amennyiben a rendelkezésre álló memória elfogy, a böngészőprogram lefagyhat, mely hatására elvesznek az abban tárolt adatok, megnyitott weboldalak, befejezetlen levelek, beszélgetések. Ez a szituáció egy olyan telefonban melyben a telefonon futó programok egy részét JavaScript alapú technológiával biztosítják – mint például a WebRunTime -, kritikus adatvesztéshez, az eszköz rendszerének összeomlásához is vezethet, így nem megengedhető.

Dolgozatomban bemutatom a JavaScript alapú programok processz szintű futtatását, mely technika segítségével minimalizálható a memória elfogyás esetén fellépő adatvesztés és elkerülhető a rendszer teljes összeomlása. A programok processzekbe szervezése azonban költséges lehet futásidő vagy memória használat tekintetében. Míg ez a költség asztali környezetben a tág korlátok miatt nem jelent gondot, beágyazott környezetben annál nagyobb problémát jelenthet. A dolgozatom részeként kidolgozok egy több processz párhuzamos futtatásának memória és sebesség mérésére alkalmas keretrendszert, mely által produkált eredmények alapján válaszokat kaphatunk és további kérdéseket vethetünk fel a témában.

Dolgozatomban az Apple vezetése alatt fejlesztett szabad forráskódú, WebKit kódnéven futó többcélú böngésző motorját választottam, melyet asztali és beágyazott környezetben is több böngésző használ. A WebKit portjai közül a Qt C++ függvénykönyvtár és platformfüggetlen GUI keretrendszer portját választottam, melyet Linux operációs rendszeren fordítok és használok. A memóriahasználat mérésére a Linux kernel által biztosított proc pseudo-filerendszert használom fel, amelyet a Linux::Smaps Perl modul segítségével alakítok szemléletes és felhasználható információvá.