

# Erlang/OTP

## Magas rendelkezésre állású, elosztott rendszerek fejlesztése

### *annotáció*

Készítette: Czinkos Zsolt, Gazdaságinformatikus Szak (BSc)

Konzulens: Dr. Fodor Szabina

2012

Az Erlang nyelvet az Ericsson-nál hozták létre rendkívül magas rendelkezésre állású telefonrendszerek fejlesztéséhez 1986-tól kezdődően. Az 1998-ban nyílt forráskódúvá vált Erlang nyelvet – és a platformba foglalt Open Telecom Platform (OTP) keretrendszert – ma már a telekom iparon kívül is használják, elsősorban skálázható, magas rendelkezésre állású rendszerek építéséhez. Az Erlang megalkotásánál az elsődleges cél magas rendelkezésre állású, hibatűrő rendszerek építése volt (a folyamatos működést nem szakíthatja meg szoftverfrissítés sem). A megfelelő redundancia csak egynél több géppel biztosítható, ezért olyan megoldásra volt szükség, amely lehetővé teszi, hogy magas szintű, erre a célra kialakított nyelven lehessen a megszokottnál egyszerűbben párhuzamos (konkurens) programokat fejleszteni. Az Erlang két alapra építve éri ezt el:

- Egyrészt funkcionális nyelv, nincs megosztott állapot-változó a rendszerben – *no shared state*, minden függvény megkap paraméterként minden adatot, ami a feladata elvégzéséhez szükséges. Ugyanazokkal a paraméterekkel meghívva mindig ugyanazt az eredményt adja vissza, akárcsak például az  $f(x) = x + 1$  függvény (*referential transparency*).
- Másrészt az Erlang folyamatok (*process*) közti kommunikációt aszinkron üzenetküldés teszi lehetővé; az aktor modellnek megfelelően a rendszert aktorok alkotják, amelyek bizonyos viselkedésmintával rendelkeznek, és képesek üzenetet küldeni, fogadni, további aktorokat létrehozni.

A dolgozat bemutatja az Erlang nyelv alapvető elemeit (példákkal segítve a megértést) és az Open Telecom Platform szoftverkönyvtár alapelveit, működési mechanizmusát. Az alapozás után egy egyszerű webes alkalmazás megépítését követi végig lépésről lépésre, kódrészletekkel bemutatva a gyakorlatot. A dolgozat végére összeálló

---

rendszer tőzsdei kereskedési adatokat továbbít a böngésző előtt ülő felhasználóknak, azonnal (*real-time*) vagy a beállított késleltetéssel. A szolgáltatást Erlang node-okból álló klaszter biztosítja, amely a terhelés növekedésével egyszerűen, pusztán egy újabb node hozzákapcsolásával bővíthető.

A példaprogram felépítése során a dolgozat bemutatja az Erlang azon tulajdonságait, amelyek lehetővé teszik a folyamatos működést biztosító rendszerek fejlesztését.