

Saleve: párhuzamos grid-alkalmazások fejlesztőeszköze

DÓBÉ PÉTER, KÁPOLNAI RICHÁRD, SZEBERÉNYI IMRE

BME Irányítástechnika és Informatika Tanszék
{dobe, kapolnai, szebi}@iit.bme.hu

Lektorált

Kulcsszavak: Saleve, grid alkalmazásfejlesztés, paraméterelemzés, EGEE

A Saleve rendszer egy párhuzamosan futó, úgynevezett paraméterelemző alkalmazások fejlesztését és futtatását segítő eszköz. A Saleve-vel fejlesztett programokat változtatás nélkül integrálhatjuk különböző köztesrétegeket alkalmazó gridekbe, így az alkalmazás fejlesztőjének nem szükséges a köztesréteg technikai részleteit ismernie.

1. Bevezetés

Napjainkban a tudományos számítások futtatására már számos számítógépes erőforrás rendelkezésre áll, melyek használatára azonban még nincs mindenütt megfelelő támogatás. Bár sok fejlesztés célozta meg ennek segítségét, egy kutatónak még ma is számos akadállyal kell szembenéznie, ha igénybe kíván venni egy elosztott, párhuzamos számítási rendszert, így például egy gridet.

A fenti nehézségeket kívánjuk áthidalni a Saleve keretrendszerrel, amely elfedi a párhuzamos alkalmazás fejlesztője elől a mögöttes infrastruktúra technikai részleteit. A Saleve egy speciális, könnyen párhuzamosítható feladatcsoport, a paraméterelemző algoritmusok implementálására összpontosít. Segítségével olyan párhuzamos alkalmazások készíthetők, melyek akár többféle infrastruktúrán is végrehajthatóak változtatás nélkül.

A továbbiakban részletesen ismertetjük a Saleve által támogatott feladatokat és bemutatjuk Európa legkiterjedtebb grides projektjét, az EGEE projektet. Ezt követően felvázoljuk a Saleve fejlesztésének főbb mozgatórugóit és célkitűzésit, majd a 4. szakaszban bemutatjuk a rendszer működését, végül pedig összefoglaljuk eredményeinket és a továbbfejlesztési lehetőségeket.

2. Grid alkalmazása paraméterelemző feladatokhoz

2.1. Paraméterelemző feladatok

A gyakorlatban sokszor felmerül az igény arra, hogy egy adott algoritmust több száz, vagy akár sok ezer különböző bemeneti paraméterértékkel lefuttassunk: az ilyen feladatok a paraméterelemzések (Parameter Study/Parameter Scan).

A várt megoldás bizonyos esetekben az egyes paraméterekkel kapott kimenetek összessége, de gyakran egy végső, összesítő lépés révén kapjuk ezekből a végeredményt. Egy kimerítő optimumkeresés során például ez az utolsó lépés egyetlen paraméter kikeresését jelenti. Egy másik egyszerű példa egy nem analitikus

függvény numerikus integrálása egy adott tartományon. A tartományt feloszthatjuk egymást nem fedő résztartományokra, ezek lesznek az integrálást elvégző eljárás bemeneti paraméterei, a végső lépés pedig a részintegrálokhoz az összegzése.

A paraméterelemzés problémaköre felbukkan számos kísérleti tudományágban, különösen fizikai szimulációknál. Ide tartozik továbbá a nagyenergiájú részecskefizika, az asztrofizika, génkutatás, gyógyszerkutatás, földrengés-kutatás. A paraméterelemzéshez hasonlóan részfeladatokra osztható feladat általában minden olyan mérnöki feladat, probléma, amely közönséges differenciálegyenlet-rendszerrel írható le. Ilyenek például a statikai feladatok, mely területen megemlítendő a BME-n végzett kutatás, melynek keretében vasbeton hídgerendák tervezési feladatainál alkalmaznak jól párhuzamosítható, a paraméterelemzéshez részben hasonló algoritmust [1].

A rendkívül nagy számú különböző bemenettel történő futtatásokat időben egymás után végrehajtani azonban igen sok időt venne igénybe. Megállapítható viszont, hogy az egyes lefutások között semmiféle csatolás nincs, így ezek futási sorrendje tetszőleges, ráadásul egymással párhuzamosan is történhet a Single Program, Multiple Data (SPMD) modellnek megfelelően [2]. Több processzor jelenléte tehát jól kihasználható, akár egy multiprocesszoros számítógépről, akár egy sok gépből álló fűrtről van szó – sőt, legjobb esetben egy grid infrastruktúra szolgáltatásait is igénybe vehetjük erre a célra.

2.2. A Grid-rendszerek jelene

A rohamosan növekvő számítási és adattárolási igény kielégítése érdekében már több mint egy évtizede felmerült az elképzelés egy földrajzilag elosztott erőforrás-hálózat, grid kiépítésére [3]. Ennek megvalósítására azóta számos kezdeményezés indult a világ különböző területein.

Ezek közül Európában egyik legnagyobb az Enabling Grids for E-sciencE (EGEE) [4]. A svájci CERN kutatóközpontnál levő Large Hadron Collider (LHC) részecskegyorsító érzékelőiből származó adatok feldolgozására kezdték kialakítani a rendszert, mára azonban szá-

mos szerteágazó tudományterületen vannak alkalmazásai: többek között asztrofizikában, bioinformatikában és geofizikában. A projekt 240 intézményt fog össze a világ 45 országából, köztük Magyarországról is. A grid jelenleg körülbelül 41 ezer processzort tartalmaz, 5 petabájt adatot képes tárolni és 100 ezer feladatot dolgoz fel egyidejűleg.

A BME is részt vesz az EGEE-ben, egyrészt szervező tevékenységekkel, másrészt erőforrások rendelkezésre bocsátásával. Az általunk kialakított erőforrás-készletben, amely a BMEGrid nevet kapta, jelenleg 8 darab négymagos szerver futtatja a gridbe beküldött programokat. A rendszerhez csatlakoztattunk ezen kívül egy nagy kapacitású és hatékony párhuzamos elérési adattároló eszközt, a Scalable File Share-t, amely mintegy 3 terabájt adat tárolására képes. Erőforrásainkat többségében a részecskefizikával foglalkozó Atlas kutatócsoport tagjai, valamint biomedikus kutatók használják.

A grid projektek hangsúlyt fektetnek az új alkalmazások fejlesztésének könnyítésére, hogy növeljék a felhasználók táborát. Erre megoldást nyújtanak a portálrendszerek [1,10], helyenként kiegészítve alkalmazásfejlesztési és munkafolyamat-kezelő eszközökkel [11], vagy egyéb, összetett funkciókat is támogató környezetek [12]. A Saleve megközelítése ezektől eltér: segítségével olyan párhuzamos alkalmazások hozhatók létre, amelyek saját magukat képesek a futtató környezethez eljuttatni külön eszköz nélkül és emellett könnyűsúlyúak és egy személyi számítógépen is futtathatóak. A rendszer – mint ahogy az említett környezetek is – független az alkalmazási területtől.

3. A Saleve rendszer áttekintése

3.1 Motiváció

A kutatók, mérnökök nagy része rendelkezik programozási ismeretekkel, különösen a C és Fortran nyelv te-

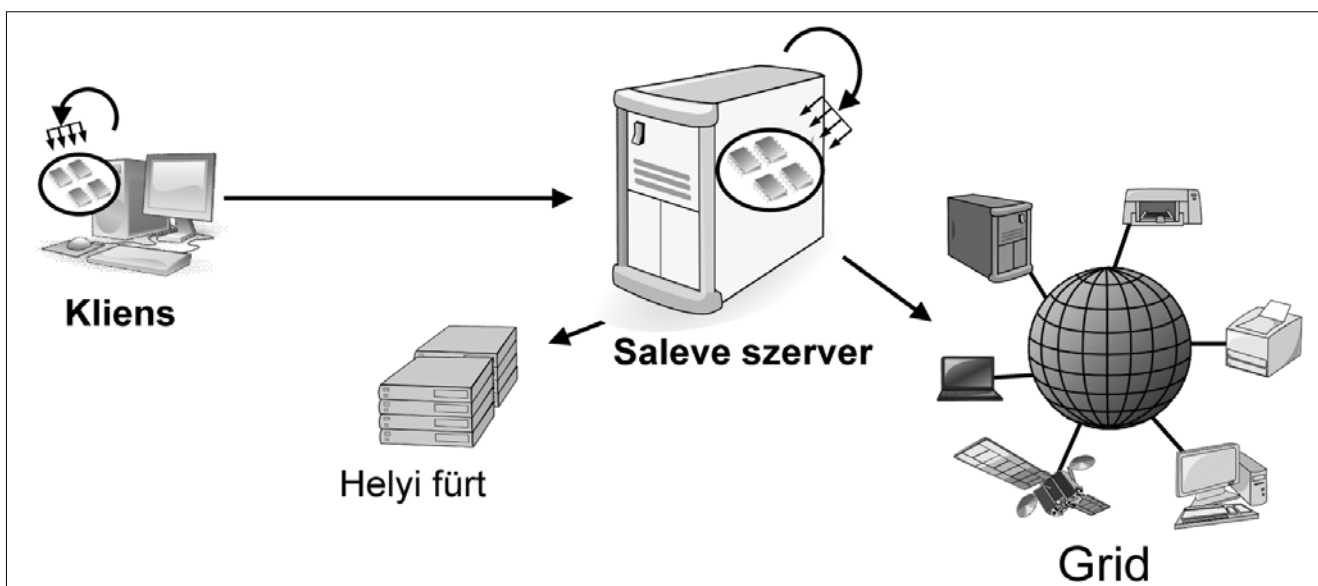
rén, így a tudományos számításokat végző hagyományos, soros feldolgozású programok elkészítése nem okoz nekik gondot. Párhuzamosan futó, elosztott programok fejlesztése azonban nagyobb programozási tudást és gyakorlatot igényel. A helyzetet nehezíti, hogy számos eltérő, nagy léptekkel változó technológia van használatban. Ide tartoznak a különböző hosszú távú ütemezők: a PBS, az LSF és a Condor [5] is.

Igaz ugyan, hogy a grid fejlesztés végső célja egy szabványosított módon elérhető világméretű szolgáltatás kialakítása, ennek megvalósulásáig azonban várhatóan még sok évet kell várni. Jelenleg az egyes grid rendszerek különböző, egymással inkompatibilis köztesréteg-szoftvert használnak. Mindezeket a technológiákat megismerni, használatukat megtanulni ahhoz, hogy egy általános problémát, például paraméterelemzést megoldhassunk a segítségükkel, felesleges energiákat von el a kutatás igazi feladataitól. Gyakori eset továbbá, hogy egy már megírt programot gyökeres átírás nélkül szeretnének használni a háttérben levő infrastruktúra változása, fejlődése után is. Ilyen változás lehet az, amikor egyetlen számítógépről áttérnek egy helyi fürt használatára, vagy pedig a fürről térnek át valamely griden történő futtatásra.

3.2 Megoldási javaslat

Ezen nehézségek áthidalására hivatott a Saleve rendszer, egy nyílt forráskódú segédeszköz a C nyelvű, párhuzamos futásra is képes programok fejlesztésére. A Saleve egyaránt használható új programok készítésére és már meglévő, de egyszerre csak egy processzoron futni képes kódok gyors átírására. Az alapvető cél elfedni a háttérben megbújó különféle számítási technológiákat és egy ezektől független, könnyen elsajátítható metodológiát nyújtani olyan paraméterelemző alkalmazások gyártására, amelyek az egyszerű szekvenciális futáson kívül képesek a párhuzamos rendszereket is kihasználni.

1. ábra A Saleve használati lehetőségei



4. A Saleve működése

4.1. Kliens-szerver architektúra

A Saleve rendszer működésének megértéséhez tekintünk egy C nyelven megírt, szekvenciális parameter study (PS) alkalmazást. A felhasználó számára az egyetlen újdonság az, hogy kis mértékben átalakítva az eredeti alkalmazását, el kell készítenie a Saleve klienst. A szükséges változtatás mindössze abból áll, hogy szét kell választani a programban a paramétertartomány felbontását, a részeredmények kiszámítását és a részeredmények összegzését, majd fordításkor össze kell szerkeszteni a Saleve fejlesztői könyvtárral.

A Saleve kliens futtatásakor kiszámítja az összes résztartományhoz tartozó részeredményt és összegzi azokat. Alapértelmezésben – az eredeti alkalmazáshoz hasonlóan – sorban egymás után indítja a részfeladatokat, de lehetőség van arra is, hogy párhuzamosan több részfeladatot indítson. Azonban a kliens legfontosabb képessége az, hogy el tudja küldeni saját bináris kódját és a bemeneti állományokat egy megadott Saleve szervernek.

Miután a kliens kérése, vagyis a program és az adatfájlok megérkeznek a Saleve szerverhez, a szerver minden résztartományhoz egy külön folyamatot indít el, amelyeket vagy helyileg hajt végre, vagy továbbküld egy gridbe vagy egy fűrtbe a felhasználó számára tökéletesen transzparens módon (1. ábra).

A szerver gondoskodik a továbbküldött feladatok felügyeletéről, hiba esetén újraküldésről és a részeredmények ideiglenes tárolásáról. Ebben a fázisban kliens a szerverrel való kapcsolatot megszakíthatja és később,

akár egy másik kliens példány az azonosítást követően újra felépítheti.

A szerver a kiszámított részeredményeket folyamatosan visszajuttatja a vele kapcsolatban álló kliensnek. Amikor minden részeredmény megérkezett a klienshez, akkor a felhasználó által megadott módon kiszámításra kerül a végeredmény. A feladat végrehajtásának folyamatát a 2. ábra foglalja össze.

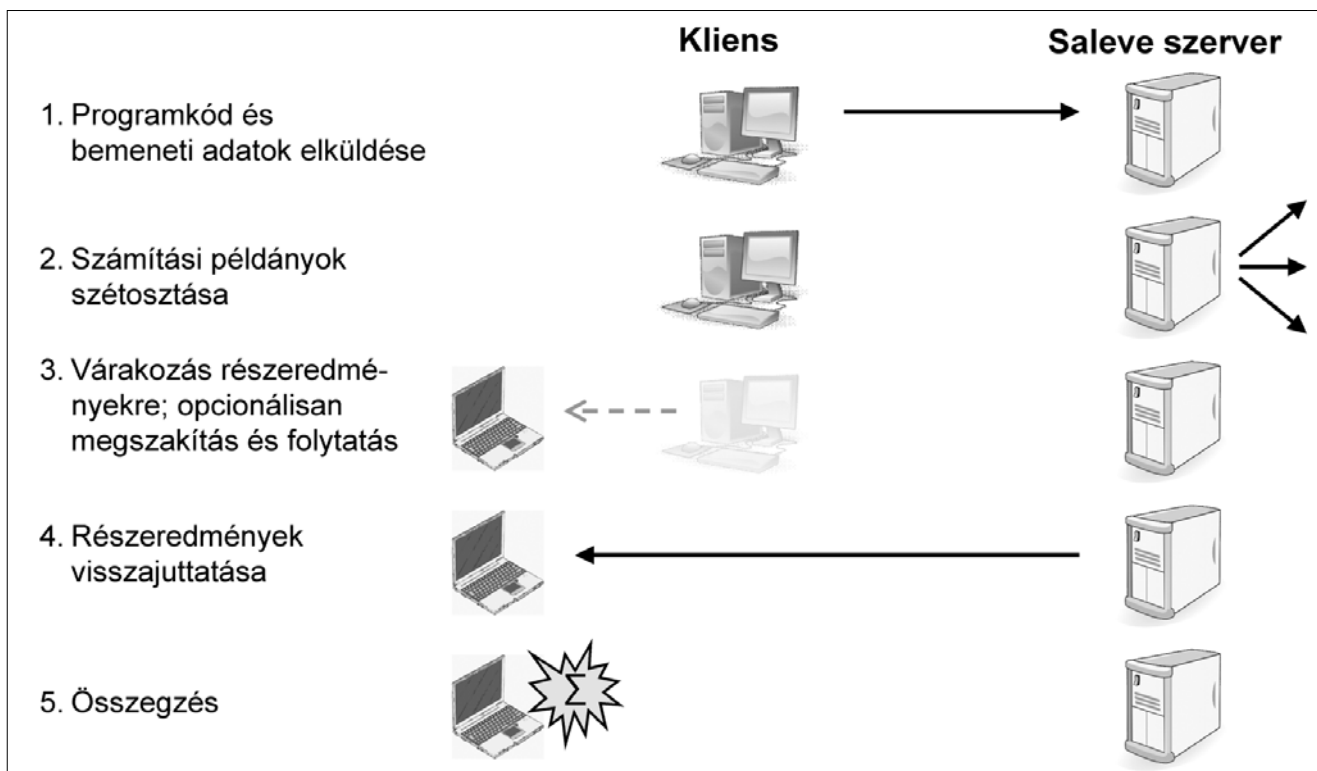
4.2. A szerver felépítése

A szerverrel szemben támasztott legfőbb követelményünk, hogy minél több elterjedt elosztott számítási környezetet támogasson, és emellett könnyen adaptálható legyen egy új ütemezőhöz vagy grides köztesréteghez. E szemléletet tükrözi a szerver komponenseinek két csoportra való felosztása: egyik típusú komponensek az általános, számítási környezettől független feladatokat végzik, a másik csoportot a plugin komponensek (kiterjesztések) képezik.

Az általános csoportba tartozik a kliensekkel való SOAP alapú kommunikációt végző komponens. A szerver tehát webszolgáltatásokat nyújt a kliensek felé, ezeken keresztül történik a feladat és a paraméterek feltöltése a szerverhez és a részeredmények visszajuttatása a klienshez. A webszolgáltatások a gSoap [6] implementációját használják. Az általános csoportba sorolandó továbbá a felhasználókat kezelő és a feladatkezelő komponens, melyeket részletesen [7,8] ismertet.

A kiterjesztések csoportja, a pluginok teszik lehetővé, hogy a szerver többféle, eltérő környezetekkel lehessen kapcsolatban, de egy új környezethez történő illesztés során ne legyen szükség a működés újratervezésé-

2. ábra A feladat végrehajtása



re. Minden egyes elosztott környezethez fejleszteni kell egy plugint, amely a környezetspecifikus kommunikációt kezeli (3. ábra).

Eddig az alábbi környezetekhez készült Saleve plugin:

- a Saleve szerveret befogadó gépen történő végrehajtás, ami a szervergépen párhuzamosan indítja el a feladatokat,
- Condor ütemezőnek [5] történő feladatbeküldés, mely a fürtöknél gyakran használatos feladatütemező,
- az EGEE grid infrastruktúrába továbbítás a gLite köztesrétegen keresztül.

4.3. Saleve és az EGEE infrastruktúra

A Saleve rendszer az előző szakaszban említett pluginnal támogatja az EGEE gridbe történő feladatbeküldést is. Egy Saleve plugin elkészítése elsősorban a megfelelő köztesréteg vagy ütemező interfészének ismeretét követeli meg és nem igényel nagy felkészültséget a Saleve architektúrájának terén. A plugin fejlesztésének folyamata egy absztrakt interfész-osztály implementálásából áll, ahol az adatkezelést segíti a Saleve programkönyvtár. A kihívást inkább a grid felé történő hitelesítésben és a grides feladatok menedzselésében találtuk. A feladatok gondozását a futtatást végző infrastruktúra megbízhatatlansága teszi szükségessé: bizonyos részfeladatok végrehajtása meghiúsulhat, ezeket újra be kell küldeni.

A gLite köztesréteg, az EGEE infrastruktúra szoftvermotorja olyan tanúsítványalapú hitelesítést és erőforrás-kiosztást alkalmaz, amelyben a felhasználókat és az erőforrásokat virtuális szervezetekbe (VO) csoportosítja. Ha egy felhasználó el kíván érni egy erőforrást, például

feladatot szeretne beküldeni, akkor a tanúsítványából egy rövidlejárátú, úgynevezett proxy tanúsítványt kell generálnia, ezt csatolnia kell a beküldött feladathoz és rendszeresen megújítania. Ez az eljárás segít megóvni a hosszú távú tanúsítványt abban az esetben, ha a proxy tanúsítvány kompromittálódna.

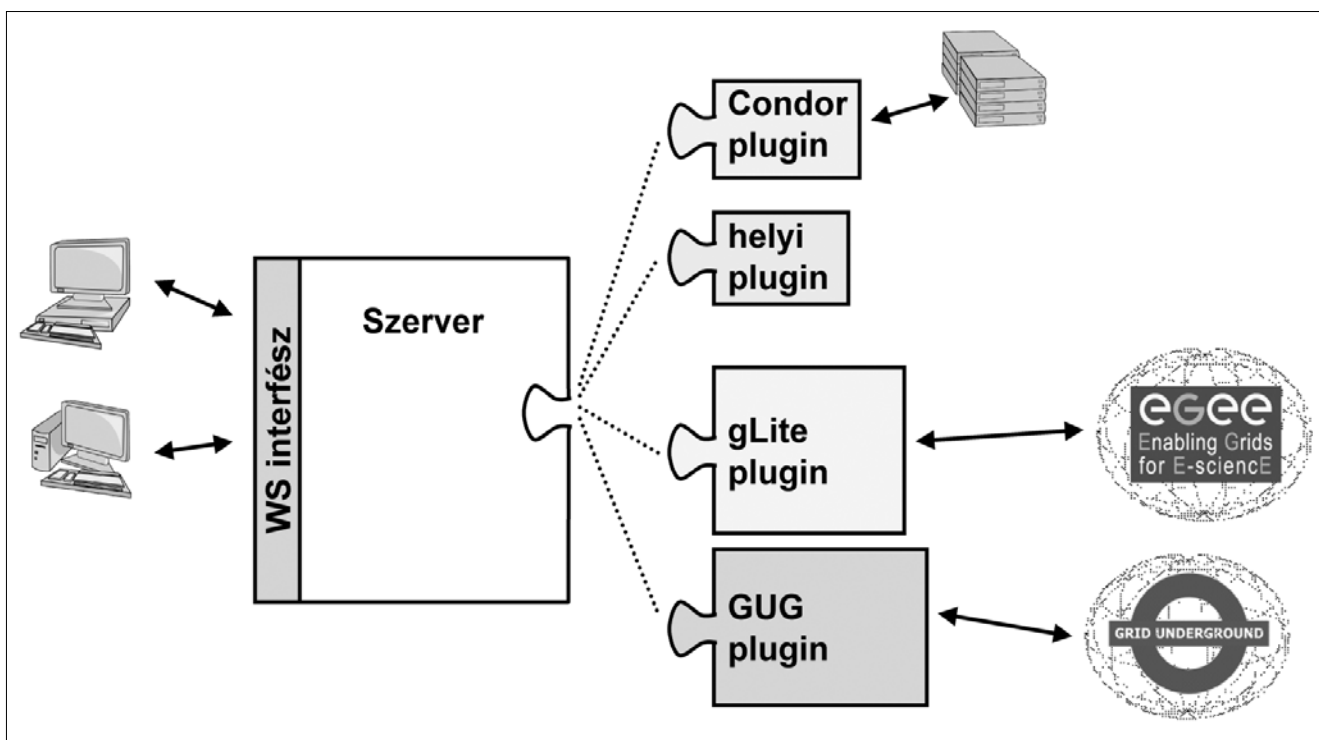
Jelenlegi megoldásunkban a Saleve szerver saját tanúsítványt tart birtokában, vagyis közvetlenül hozzáfér a grides erőforrásokhoz, hiszen tagja valamelyik virtuális szervezetnek. Így a proxy generálása és periodikus megújítása teljesen elrejtethető a felhasználó elől, aki így nem észleli, hogy a feladata az EGEE gridben, vagy egy helyi Condor-fürtön futott-e le.

5. Összefoglalás

A bemutatott Saleve rendszer átlátszó absztrakciós réteget képezve a különféle elosztott környezetek köztesrétege és ütemezője felett megkönnyíti a parameter study típusú párhuzamos alkalmazások fejlesztését. Legfőbb előnye, hogy az eredeti alkalmazás enyhén módosított példánya, a Saleve kliens többféle környezetben vagy akár a helyi gépen is futtatható változtatás nélkül, így technikai részletek ismerete nélkül is könnyen fejleszthetünk alkalmazásokat akár Európa legnagyobb infrastruktúrájába: az EGEE gridbe.

A továbbfejlesztési lehetőségek közül a közeli jövőben hangsúlyt helyezünk a pluginok dinamikus cseréjének fejlesztésére és a gridbe küldött feladatok jobb menedzselésére. Terveink között szerepel a kliens-szerver kommunikáció rugalmasabbá tétele webstream-ek segítségével. A Saleve projekt jelenlegi állapotáról [9] weblap ad tájékoztatást.

3. ábra A Saleve architektúrája



Köszönetnyilvánítás

E munka részben a Nemzeti Kutatási és Technológiai Hivatal Pázmány Péter programjának (RET-06/2005) támogatásával jött létre. A szerzők szeretnék kifejezni a köszönetüket az Európai Unió által támogatott EGEE projektnek (EU INFSORI-031688), valamint az NKFP MEGA (2_009_04) projektnek.

Irodalom

- [1] D. Pasztuhov, A. Sipos, I. Szeberényi: Calculating Spatial Deformations of Reinforced Concrete Bars Using Grid Systems, MIPRO 2007 – Hypermedia and Grid Systems, Opatija, Croatia, 2007., pp.189–194.
- [2] P. E. Black: Algorithms and Theory of Computation Handbook, CRC Press LLC, U.S. National Institute of Standards and Technology, 1999.
- [3] I. Foster, C. Kesselman: The Grid 2: Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishers Inc., 2003.
- [4] EGEE-II Information Sheet, 2007. <http://www.eu-egee.org/sheets/uk/egee-ii.pdf>
- [5] D. Thain, T. Tannenbaum, M. Livny: Distributed Computing in Practice: the Condor Experience, Concurrency and Computation: Practice and Experience, 2005., pp.323–356.
- [6] R. A. van Engelen, K. Gallivan: The gSOAP Toolkit for Web Services and Peer-To-Peer Computing Networks, In Proc. of the 2nd IEEE International Symposium on Cluster Computing and the Grid (CCGrid2002), Berlin, Germany, 2002., pp.128–135.
- [7] Zs. Molnár, I. Szeberényi: Saleve: simple web-services based environment for parameter study applications. In Proc. of the 6th IEEE/ACM International Workshop on Grid Computing, 2005., pp.292–295.
- [8] P. Dóbbé, R. Kápolnai, Szeberényi: Simple grid access for parameter study applications. In 6th International Conference on Large-Scale Scientific Computations, Szopopol, 2007. (nyomtatás alatt)
- [9] Saleve projekt, <http://egee.ik.bme.hu/saleve/>
- [10] P. Kacsuk, Z. Farkas, G. Hermann: Workflow-level parameter study support for production Grids, In Proc. of the ICCSA'2007, Kuala Lumpur, 2007. Springer LNCS 4707, pp.872–885.
- [11] P. Kacsuk, G. Dózsa, J. Kovács, R. Lovas, N. Podhorszki, Z. Balaton, G. Gombás: P-GRADE: A Grid Programming Environment. Journal of Grid Computing, Vol. 1, No.2, 2004. pp.171–197.
- [12] T. Fahringer, A. Jugravu, S. Pillana, R. Prodan, C. Seragiotto, Jr., H.-L. Truong: ASKALON: a tool set for cluster and Grid computing, Concurrency and Computation: Practice and Experience, Vol. 17, 2005., pp.143–169.

Hírek

A biometrikus azonosító eszközök – ujjlenyomat-, íriszleolvasók, arcfelismerők – piaca jelentős fejlődésnek indult az utóbbi időben, Magyarországon is egyre több cég használ ilyen eszközöket. **FaceReading** elnevezéssel figyelemre méltó hazai fejlesztésű képfelismerő programot mutattak be december elején, amely képes arra, hogy akár egy egyszerű webkamerával, valós időben készített képeken is 95%-os pontossággal azonosítsa azokat a kulcspontokat, melyek segítségével az arc elemezhetővé válik és másodpercek alatt megállapíthatóak az illető alapvető személyiségjegyei, melyekből számos tulajdonságára lehet következtetni.

A FaceReading Kft. az arcelemzés sok évszázados tudományát ötvözte a mesterséges intelligenciára épülő technológiával, melynek nyomán az Arcolvasó neurális hálózatokból felépített algoritmusai a relatíve rossz minőségű, alacsony felbontású mozgóképen is képesek az arc kulcs-pontjait követni és azonosítani. A cégvezetők szerint a technológia forradalmasíthatja a munkaerő-kiválasztást, illetve annak hatékonyságát. A kulcspontok adataiból Nagy Judit ügyvezető, a cég humán erőforrás menedzsmenttel foglalkozó pszichológusa szerint olyan személyiségjegyekre lehet következtetni, mely egy adott munkakörre vonatkozó alkalmassági vizsgálatban is segíteni tudja a HR-esek munkáját.

Galambos József cégvezető a sajtótájékoztatón elmondta, hogy az alacsony hardverigényű, költséghatékony rendszer további alkalmazási területekkel is rendelkezik, ugyanis megfelelő "tanítás" után képes beazonosítani az arcon kívül más tárgyakat is, így például a járművek rendszámán túl felismeri karosszériájukat, vagy akár az azokon végzett módosításokat is. A technológia természetesen felhasználható az ellenőrzött be- és kiléptetés meggyorsítására, célszemélyek kiszűrésére civil és speciális területeken egyaránt.

Az Arcolvasó iránt főként az algoritmusok skálázhatósága miatt komoly a nemzetközi érdeklődés, a cég szakemberei egyelőre csak annyit árultak el, hogy jelenleg is több új alkalmazást fejlesztenek.