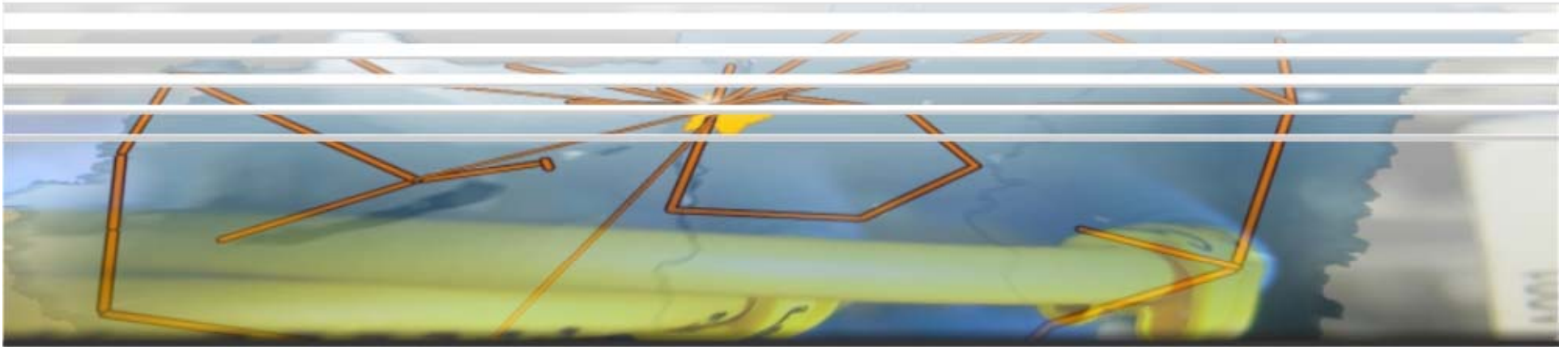


Az ARC új generációs bróker rendszere



2009.04.16.

Szeged, Networkshop 2009

Rőczei Gábor

roczei@niif.hu



Rövid összefoglaló

- KnowARC projekt
- Mi is a bróker, miért van rá szükség?!
- ARC bróker
- Bróker modulok
 - RandomBroker
 - FastestCPUBroker
 - FastestQueueBroker
 - DataBroker
- Saját bróker modul fejlesztése: C++, Python nyelveken

KnowARC projekt

- 2006-ban indult
- Kiemelt Európai Unió projekt
- NorduGrid köztesrétege: Advanced Resource Connector (ARC)
- Web service technológia alkalmazása
- Gridek közti együttműködés: Unicore, gLite, ARC
- Szabványok használata (JSDL, OGSA BES, GLUE2, stb.)



Miért van szükség brókerre?

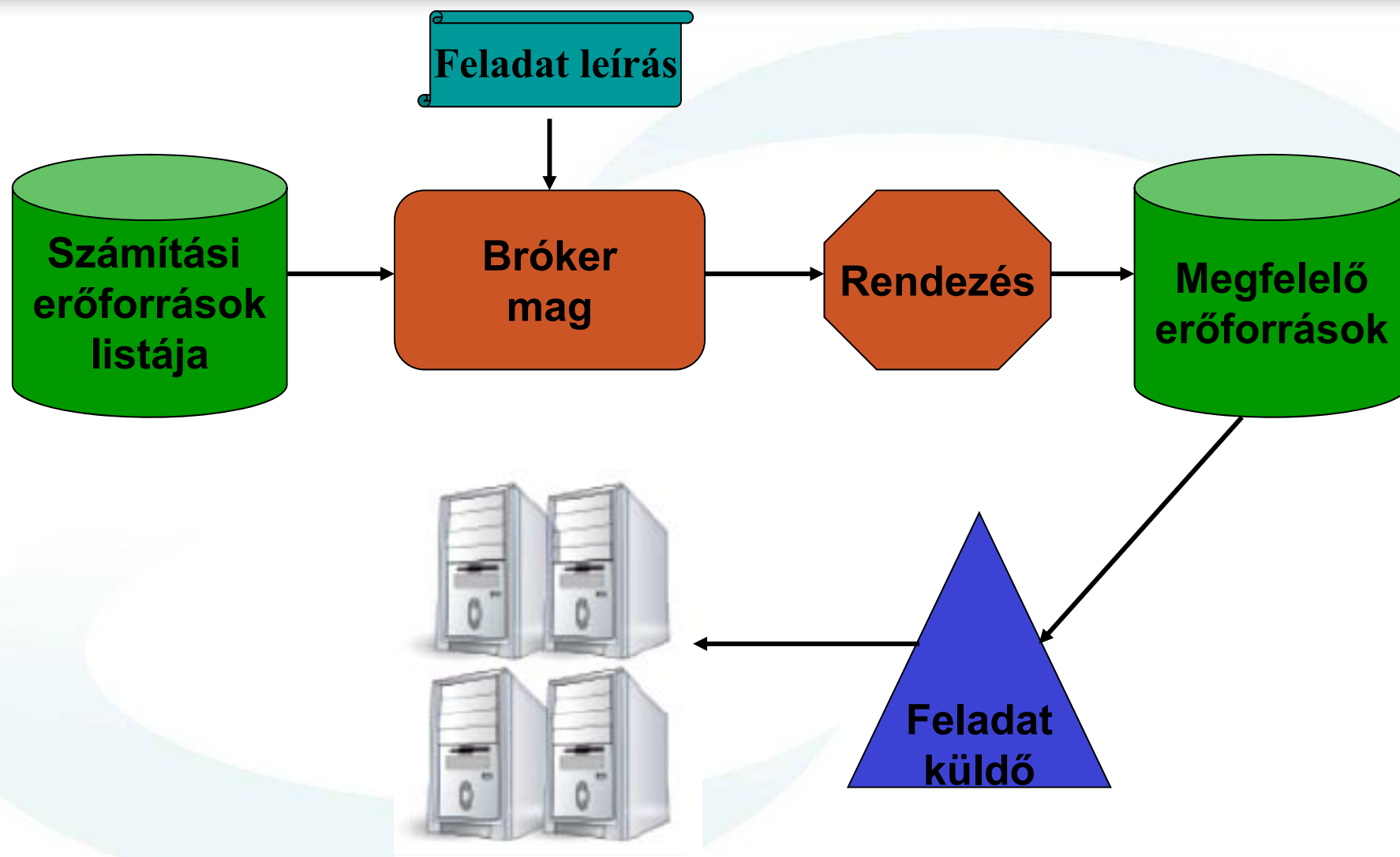
- A felhasználó szeretné olyan gépen futtatni a programját, ami minden tekintetben megfelelő számára: memória, helyterület, futtatási környezet, stb.
- Intelligens döntés hozás valamilyen algoritmus alapján



ARC bróker áttekintése (2/1.)

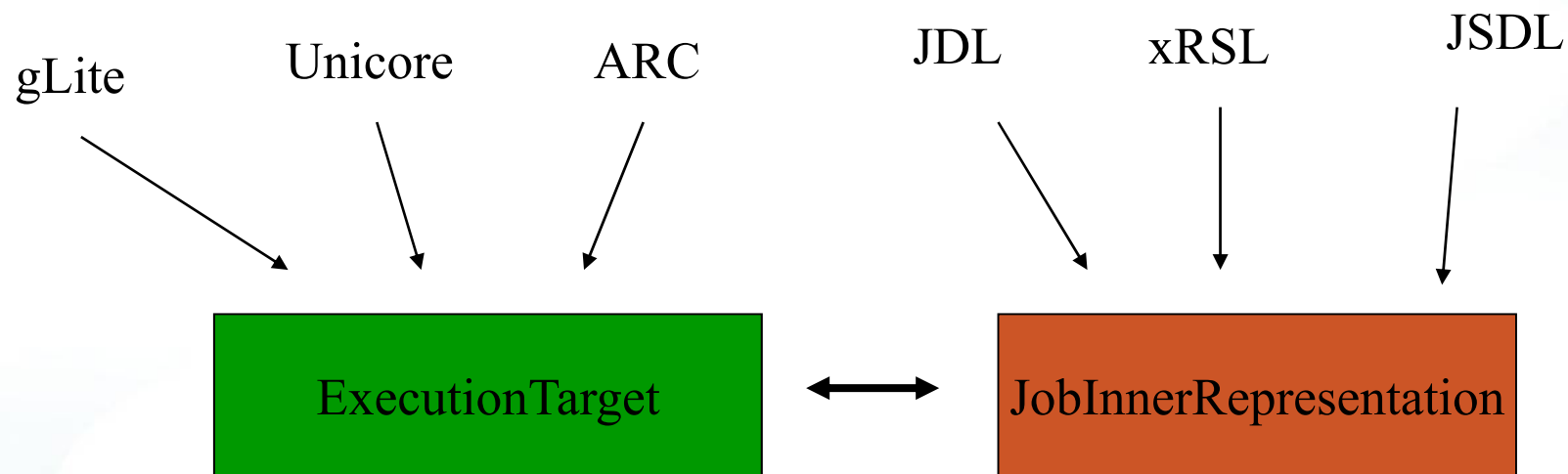
- Az ARC bróker a kliens oldalon helyezkedik el
- Kihasználja az ARC kliens könyvtár által adott eszközöket
- Köztesréteg független brókerezés: ARC, gLite, Unicore
- Betölthető modul
- Kiegészítő modul is fejleszthető hozzá
- Az broker mag végzi a felhasználó programja által igényelt és a rendelkezésre álló erőforrások összehasonlítását
- A bróker kiegészítő modulok már csak a kapott lista rendezését végzik valamilyen algoritmus szerint (pl. futás idejű adatok)

ARC bróker áttekintése (2/2.)



Belső reprezentációk

- ExecutionTarget (GLUE2)
- JobInnerRepresentation (GIN/PGI JSDL)



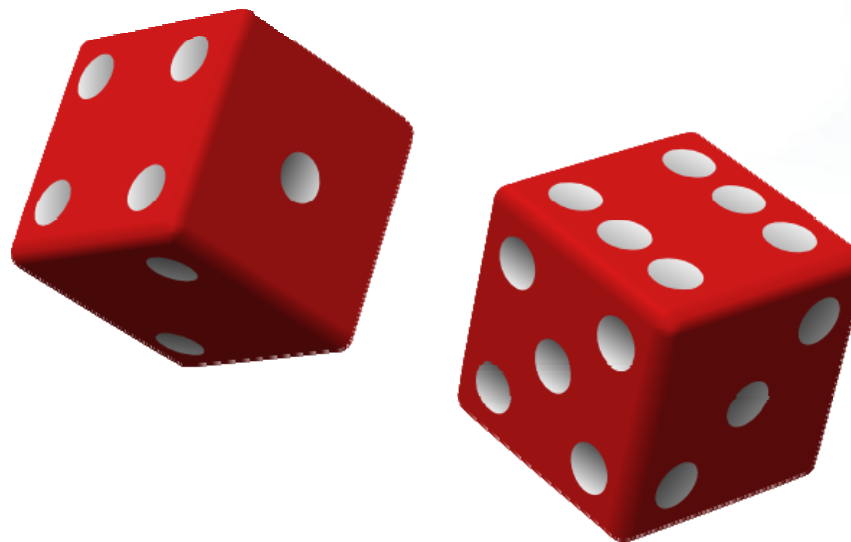
ApplicationEnvironment	RuntimeEnvironment
MinCPUTime, MaxCPUTime	TotalCPUTime

Broker modulok - RandomBroker

- Véletlenszerű sorrendet csinál

Előtte: A, B, C, D

Utána: C, A, D, B



Broker modulok - FastestCPUBroker

- CPU sebességnek megfelelően rendez
- Integer Component of SPEC CPU2000 (CINT2000)

Működési elv:

- Első lépésként eltávolítja azokat a klasztereket, amelyek nem publikálják a a specint2000-es benchmark értéket
- Olyan rendezést végez, hogy a lista elején lesz a legnagyobb benchmark értékű és a végén a legkisebb



Broker modulok - FastestQueueBroker

- A várakozó sorok hossza szerint végzi el a rendezést

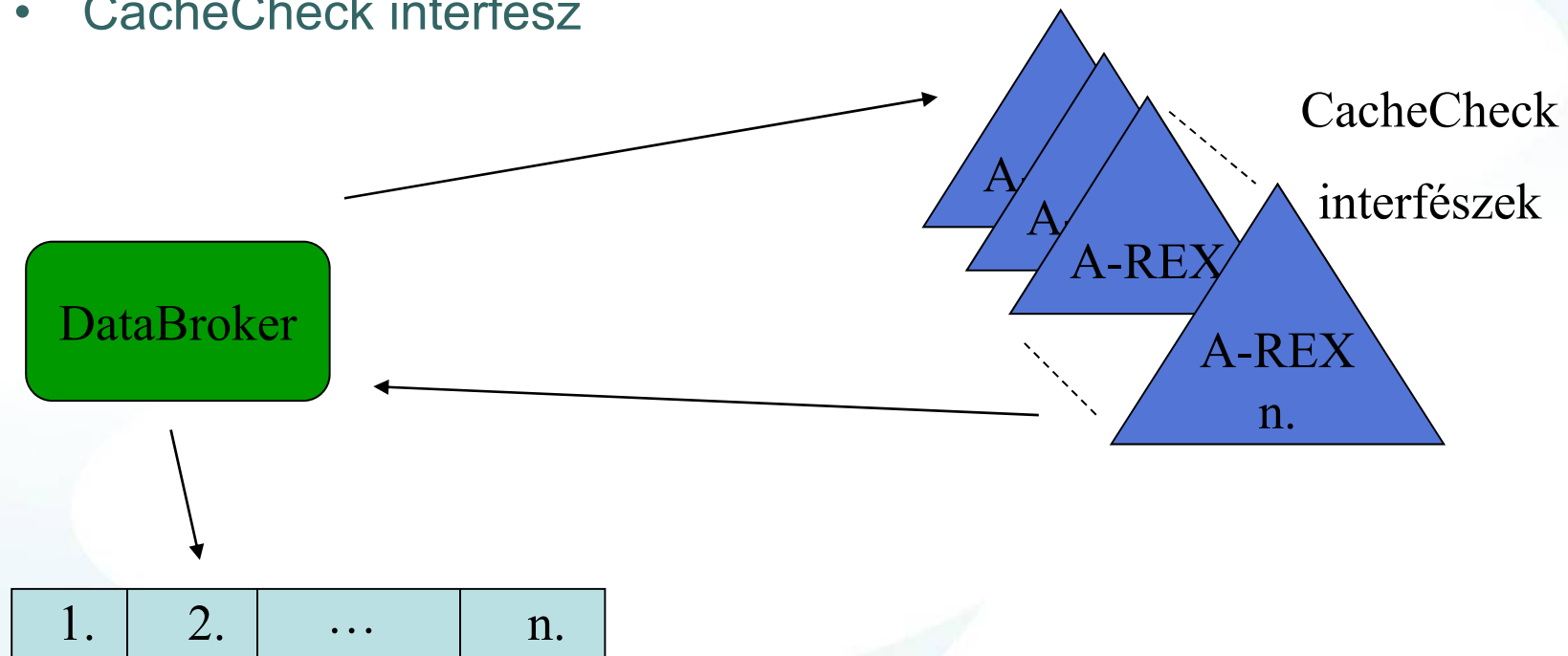
Működési elve:

- Első lépésként eltávolítja azokat a, amelyek nem publikálják a TotalSlots, FreeSlots, WaitingJobs értékeket
- A lista elejére az fog kerülni, ahol a WaitingJobs/TotalJobs érték a legkisebb, minél nagyobb, annál hátrébb fog kerülni
- Terhelés elosztás több üres várakozó sor esetén: FreeSlots értéket illetve a véletlenszám generátort használjuk fel.
- Például:
 - A.WaitingJobs = 0 és B.WaitingJobs = 0
 - A.FreeSlots = 90 és B.FreeSlots= 10
 - „A” várakozó sor lesz a nyertes, 90% eséllyel



Bróker modulok – DataBroker 2/1.

- Adat központú döntés (bemeneti fájlok)
- A-REX szolgáltatás (számítási erőforrás)
- CacheCheck interfész



Broker modulok – DataBroker 2/2.

Működési elv:

- A-REX-es várakozó soron kívül minden más eltávolításra kerül
- Ahonnan nem sikerül lekérdezni, ott nulla értéket kapunk vissza
- A lista elejére az kerül, amelyik legtöbb adattal rendelkezik.

Például:

alma.png (3300 byte), korte.gif (400 byte)

A-REX1: alma.png, korte.gif

A-REX2: korte.gif



arcsub – kliens program

- Ennek a segítségével lehet jobot küdeni

```
arcsub -b FastestCPUBroker job.xrsl
```

- Alap beállítás: RandomBroker

```
arcsub job.xrsl
```

Saját bróker modul C++ nyelven

- Broker őssosztályból kell leszármazatni
- SortTargets függvény implementálása
- Modul leíró fájl létrehozása (ARC plugin descriptor)

Fordítás:

```
g++ -I /usr/local/include -L /usr/local/lib `pkg-config --cflags glibmm-2.4 libxml-2.0` -o libaccnewbroker.so -shared NewBroker.cpp
```

Használat:

```
arcsb -b NewBroker ~/simplejob.jsdl
```

Saját bróker modul Python nyelven

- PythonBroker
- Ezeket az adatokat lehet elérni:
 - Kliens konfiguráció (pl. tanúsítvány elérési útvonala, bróker paraméterek)
 - Job leírás
 - Lehetséges erőforrások, ami rendezést lehet végezni
- Két darab python függvényt kell implementálni:

```
def __init__(self, cfg)
def SortTargets(self, PossibleTargets, job)
```

Használat:

```
arcsb -b PythonBroker:SampleBroker.MyBroker
arcsb -b PythonBroker:SampleBroker.MyBroker:10
```

Hasznos linkek

<http://svn.nordugrid.org>

<http://www.knowarc.eu>

<http://www.nordugrid.org>

Köszönöm a figyelmet!



Róczei Gábor

roczei@niif.hu