



NORDUGRID

*Grid Solution for Wide Area
Computing and Data Handling*

Pagrindiniai Grid principai ir Nordugrid ARC pristatymas

Aleksandr Konstantinov
MTMI, Vilniaus Universitetas/Lietuva ir
Oslo universitetas/Norvegija

Turinys

- Kas yra “Grid”?
- Kas keičiasi?
- Nesusipratimai.
- Aplikacijos
- “Middleware”
 - Legion
 - Unicore
 - Globus
- “Grid” vartotojai
 - Identifikavimas
 - Autorizavimas
- NorduGrid
- ARC
 - Kaip veikia
 - Užduoties aprašymas
 - Pasiruošimas
 - Paprasta užduotis
 - Sudėtinga užduotis

Kas yra “Grid”?

- Naujas terminas – prasmė dar nenusistovėjo.
 - Kiek žmonių, tiek ir apibrėžimų
 - Dažnus nesusipratimai.
 - Standartizavimas tik prasidėjo <http://www.globalgridforum.org/>.
- Idėja sena:
 - Vienodas ir saugus priėjimas prie kompiuterinių resursu, kurie yra geografiškai nutolę ir labai skirtingi.
 - Ir resursai ir naudotojai pastoviai keičiasi.
 - Decentralizuotas valdymas.
 - Sistema apjungia resursus priklausančius skirtingiems savininkams.
 - Žmogiškasis bendradarbiavimas ne mažiau svarbus negu technologijos
- “Grid” aplinka paprastai kuria programinis sluoksnis įdiegtas kompiuteriniuose, duomenų saugojimo ar kituose resursuose.
 - Prigijęs terminas - “**middleware**”.
- *I. Foster and C. Kesselmann, The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufman Publishers, 1998*

Kas keičiasi?

- Nebesvarbu kur fiziškai randasi kompiuteris ir saugomi duomenys.
- Svarbi tampa unifikacija:
 - Vartotojų atvaizdavimui nebenaudojami kompiuteriniai vardai ir slaptažodžiai
 - Resursų suvienodinimas: virtualizavimas arba unifikuotos sąsajos.
 - Labai daug neišspręstų klausimų.
 - Standartizacijos procesai tik prasidėjo.
 - Virtualus kompiuteriai gali padėti.
 - Kas turi prisitaikyti, vartotojo programa ar “Grid” aplinka?

- “Grid” padidina resursus
 - Vienas skaičiavimo resursas (kompiuteris) negalės paskaičiuoti daugiau.
 - Resursai gali būti išnaudojami racionaliau.
- “Grid” yra juoda dėžė ir viskas viduje vyksta automatiškai
 - Kol kas tik ateities planuose
 - Realiai naudotojai vis dar turi daug išmokti apie konkrečios realizacijos ypatybes. Galimybė kontroliuoti užduoti atliekama “Grid” aplinkoje tampa svarbi.

Programos “Grid” aplinkai

- Ar programos turi būti pakeistos tam, kad veiktų “Grid” aplinkoje?
 - 1) Be pakeitimu. “Grid” yra kaip didelis klasteris ir “Grid” sluoksnis perkels programa ir duomenys ten kur jie bus vykdomi. Programa negalės pasinaudoti “Grid” privalumais.
 - 2) Programa iš anksto arba pagal reikalavimą įdiegiama resursuose ir vartotojas bendrauja su ją per “Grid” sąsają. Daugiausiai problemų iškyla, kai programos reikalauja pastovaus bendravimo su vartotoju.
 - 3) Programa pakeičiama taip, kad galėtų išnaudoti “Grid” galimybes. Tai vienintelis būdas, kai “Grid” turi savo OS tipo sąsają. Ypač naudinga programoms, kurie gali atliekamą darbą padalinti į beveik nepriklausomas dalis.

Yra ne taip jau ir daug bazinių “middleware” paketų. Ir dauguma kitų “Grid” sprendimų padaryti jų pagrindu.

Tarp plačiausiai naudojamų:

- **Legion** – Worldwide Virtual Computer - <http://legion.virginia.edu>
 - Klasikinis pavyzdys “middleware”, kuri pati sukuria operacinę sistemą.
 - Vartotojas mato visą sistemą kaip vieną kompiuterį
 - Viskas yra objektas – kompiuterinė įranga, programa, prietaisas.
 - Naujas resurso tipas- nauja objekto klasė.
 - Apibrėžtos sąsajos ir funkcionalumai, bet ne realizacijos. Baziniai objektai realizuoti.
 - Vartotojai gali sukurti savo klases ir netgi savo realizacijas jau egzistuojančių klasių.
 - Komeracinė.

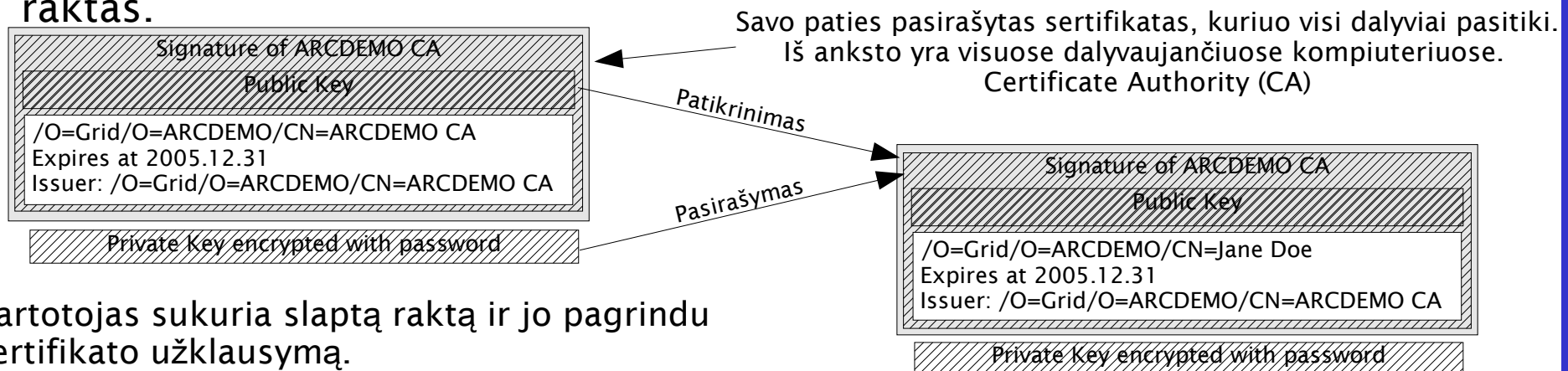
- **UNICORE** - “Uniform Interface to Computing Resources” - <http://www.unicore.org>
 - Paslepia kompiuterinių resursų ypatybes už unifikuotomis sąsajomis.
 - Lengvai naudotis nes turi vartotojo grafinę sąsają uždavinių kūrimui ir valdymui
 - Už duotis yra abstrakti ir sudėtinga. Užduotys gali tarpusavyje priklausyti.
 - Kuriamos vartotojo sąsajos kiekvienai svarbiai programai
 - Egzistuoja sąsaja bendros užduoties sukūrimui (“legacy jobs”), bet specializuotos sąsajos yra pageidautinos.
 - Neseniai tapo “Open Source”.

- **Globus Toolkit™** - “fundamental technologies needed to build computational grids” - <http://www.globus.org>.
 - Iš pradžių servisų ir programavimo bibliotekų rinkinys.
 - Dabar prisidėjo standartai, sąsajų apibrėžimai – “Web Service Resource Framework” (WSRF) pagrindu <http://www.globus.org/wsrf/>.
 - Tai nėra vieninga sistema.
 - Atskirus servusus galima naudoti ar nenaudoti.
 - Galima kurti savo servusus.
 - Yra daug “Grid” projektų kuriančių savo sprendimus Globus pagrindu.
 - “Open Source“

Kai vartotojų daug ($10^3, 10^6, \dots$) ir resursų daug (10, 100, ...) ir jie tvarkomi skirtingų organizacijų tampa sunku kurti ir sinchronizuoti visų ir visur vardus ir slaptažodžius. Reikia kito būdo:

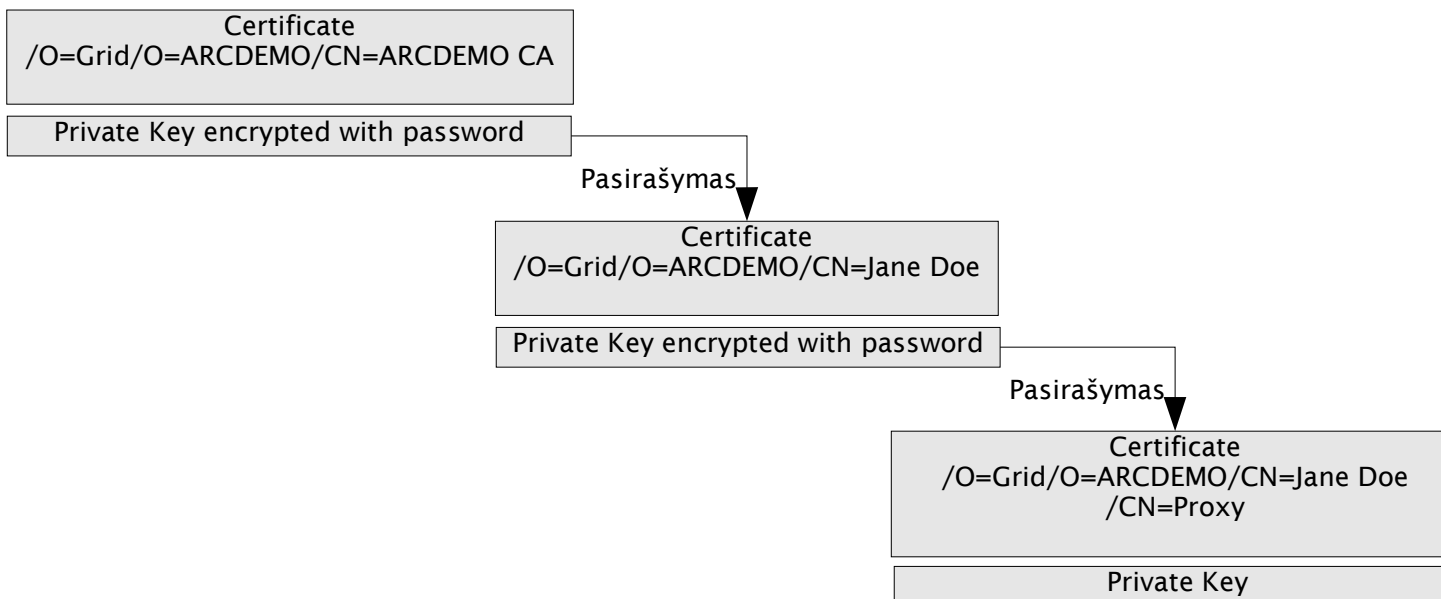
- Saugiai ir patikimai atpažinti vartotoja - identifikacija
- Kontroliuoti vartotojo galimybes - autorizavimas

- Daug “Grid” projektų naudoja “Public Key Infrastructure” (PKI) visų dalyvaujančių vienetų atpažinimui. Paprastai X.509 atmaina.
 - Raktas – slapta, pakankamai ilga atsitiktinių bitų seka (paprastai nuo 1024 iki 4096 bitų).
 - Sertifikatas – laisvai platinamas raktas susietas su slaptu raktu specialiu algoritmu, papildytas dalyvį aprašančiais atributais ir skaitmeniškai pasirašytas aukštesnio lygio dalyvio raktu.
 - Nesimetrinis kodavimas – informacijos pakeitimas panaudojant koki nors kodavimo algoritmą. Algoritme dalyvauja ir vienas ir raktu. Informacija atkurama panaudojant atvirkštinį algoritmą, kuriame dalyvauja kitas raktas.



- Vartotojas sukuria slaptą raktą ir jo pagrindu sertifikato užklausimą.
- Už klausimą siunčiamas į CA.
- CA savo metodais patikrina vartotojo autentiškumą ir suteikia jam sertifikatą.
- Visi “Grid” dalyviai pasitiki CA naudojamais vartotojų patikrinimo metodais.

- “Proxy”
 - Globus naudoja papildomą sertifikatą ir raktą, kurie yra sukuriami automatiškai.
 - Nė ra apsaugoti slaptažodžių.
 - Apsaugoti tik operacinės sistemos priemonėmis.
 - Turi trumpą galiojimo laiką. Paprastai 12 valandų nuo sukūrimo.
 - Leidžia išvengti slaptažodžio įvedimo kiekvieną kartą, kai programai reikia bendrauti su koku nors servisu. “Single log in”.



- Resursų savininkams būtų labai sunku spręsti ar kiekvienas vartotojas gali naudotis jų resursais
- Vartotojai suburiami į grupes vadinamas virtualiomis organizacijomis (VO).
- VO viduje vartotojams galimybes paskiria VO administratoriai
- Resursai pasitiki VO ir skiria jai tam tikras galimybes.
- Resursai taip pat gali priklausyti VO ir deleguoti sprendimus VO administratoriams.
- Globus naudoja paprastus tekstines rinkmenas su vartotojų sąrašais.
 - Yra programų galinčių dinamiškai kurti tas rinkmenas.
 - Atsiranda daug lankstesnių sprendimų.

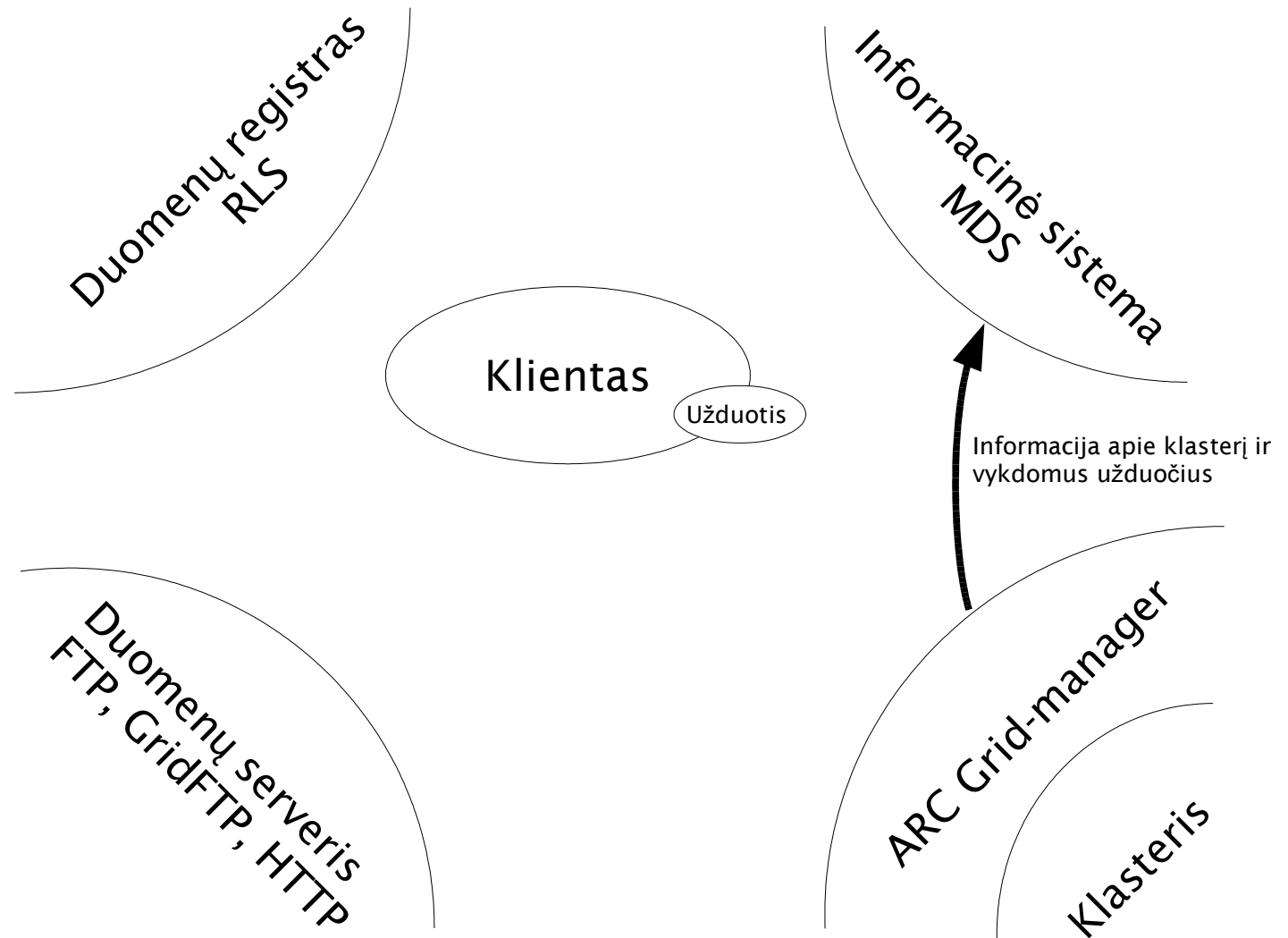
- NorduGrid, tai mokslinė kolaboracija kurią sukūrė Danijos, Estijos, Norvegijos, Suomijos ir Švedijos universitetai.
 - Bando sukurti ir palaikyti “Grid middleware“ tinkama kasdieniniam vartojimui.
 - Šiu metu apjungia resursus iš 10 šalių:
 - ~4000 procesorių
 - ~30 TB atminties



- **ARC** (Advanced Resource Connector) tai “Grid middleware“ sukurta **NorduGrid** projekto.
 - Ja sudaro servisu ir vartotojo sąsajos rinkmenų rinkinys
 - Sukurtas Globus Toolkit™ 2 pagrindu
- Pagrindiniai kūrimo principai
 - Paprastumas
 - Stabilumas
 - Nereikalauja pakeitimų dalyvaujančiuose resursuose

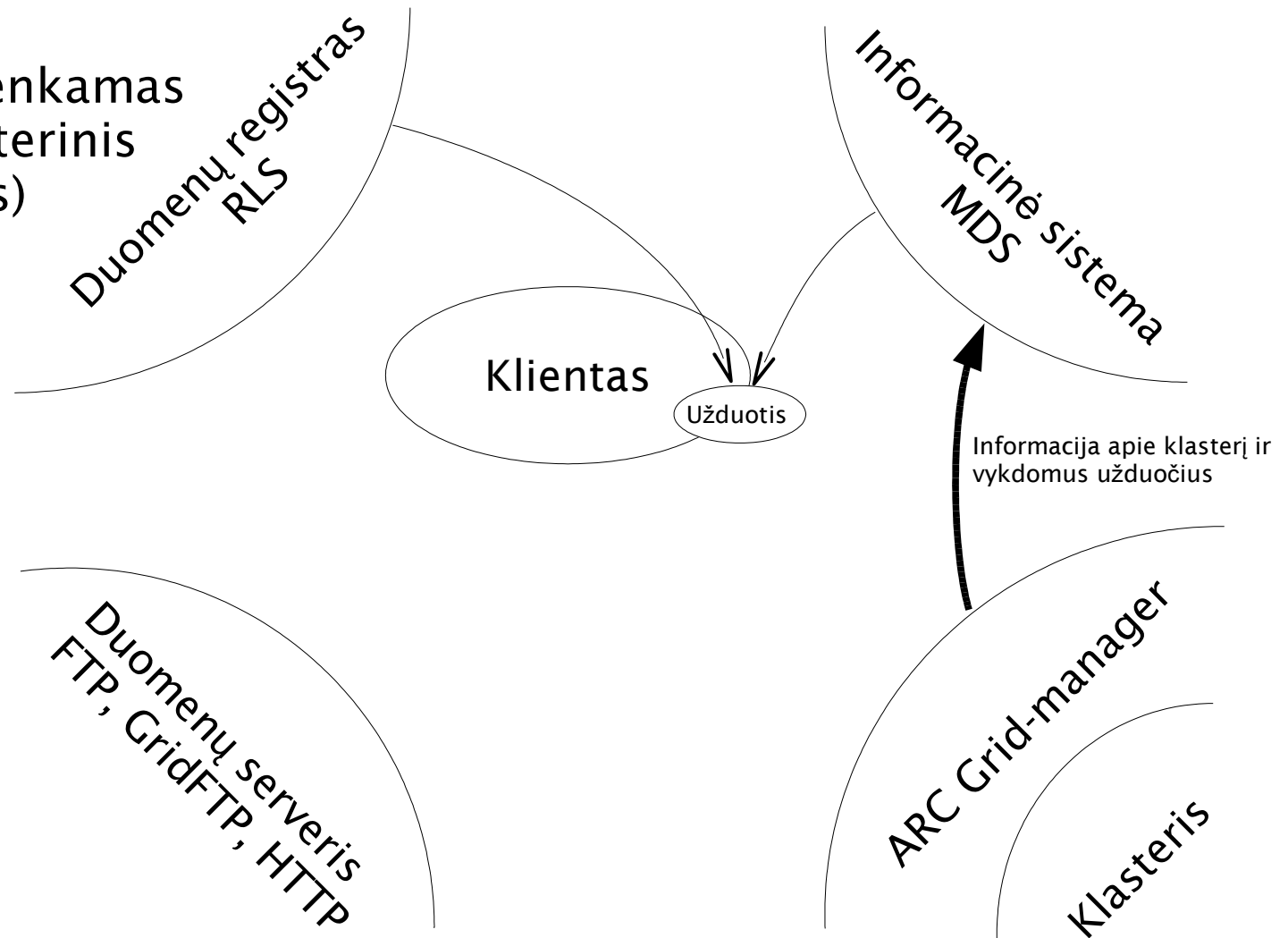
ARC - Kaip veikia

- Vartotojas suformuoja užduotį ir pateikia ją klientinei programai



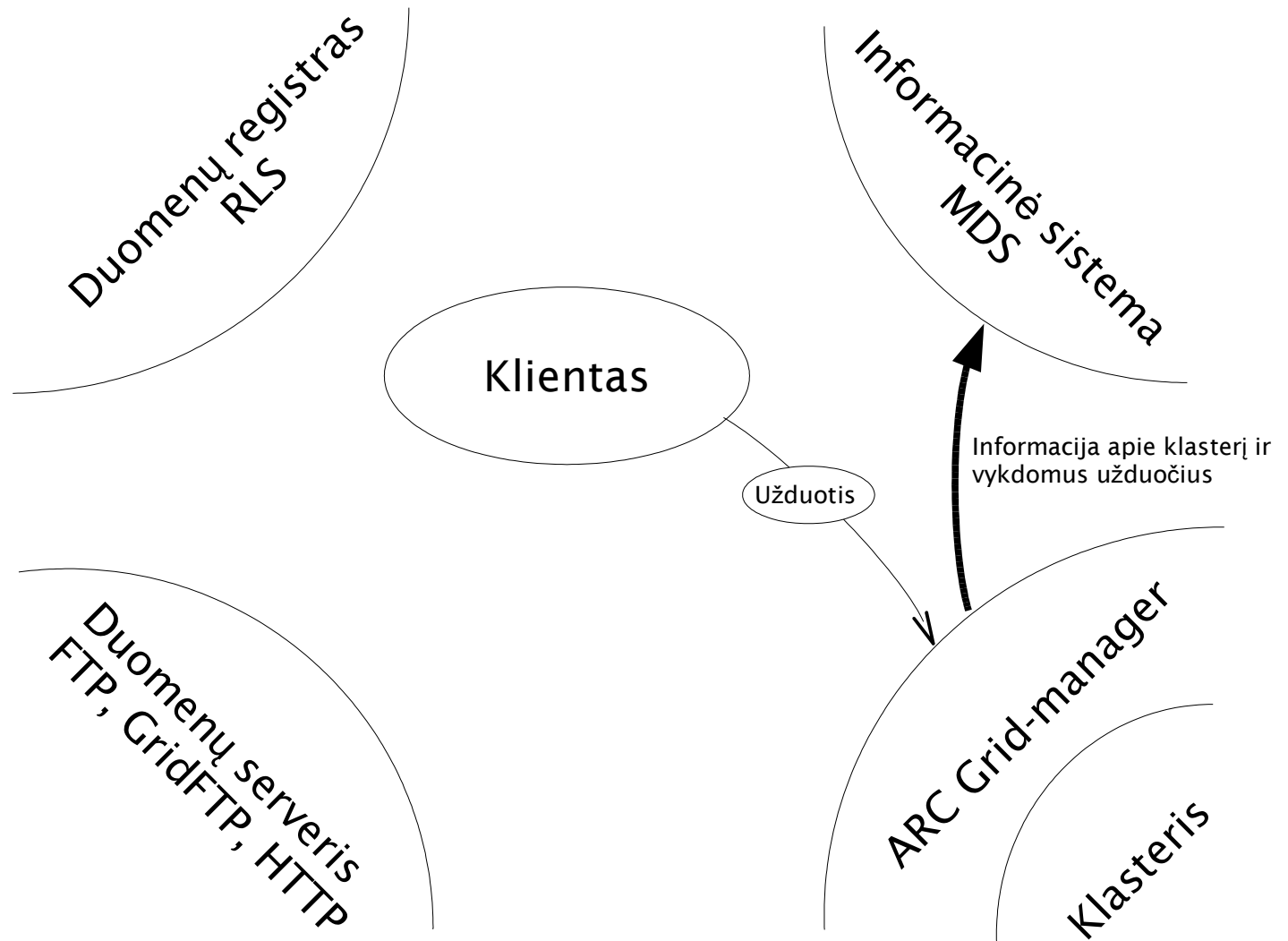
ARC - Kaip veikia

- Klientas surenka informaciją apie resursus ir duomenų saugojimo vietas
- Pagal surinktą informaciją pasirenkamas tinkamas kompiuterinis resursas (klasteris)



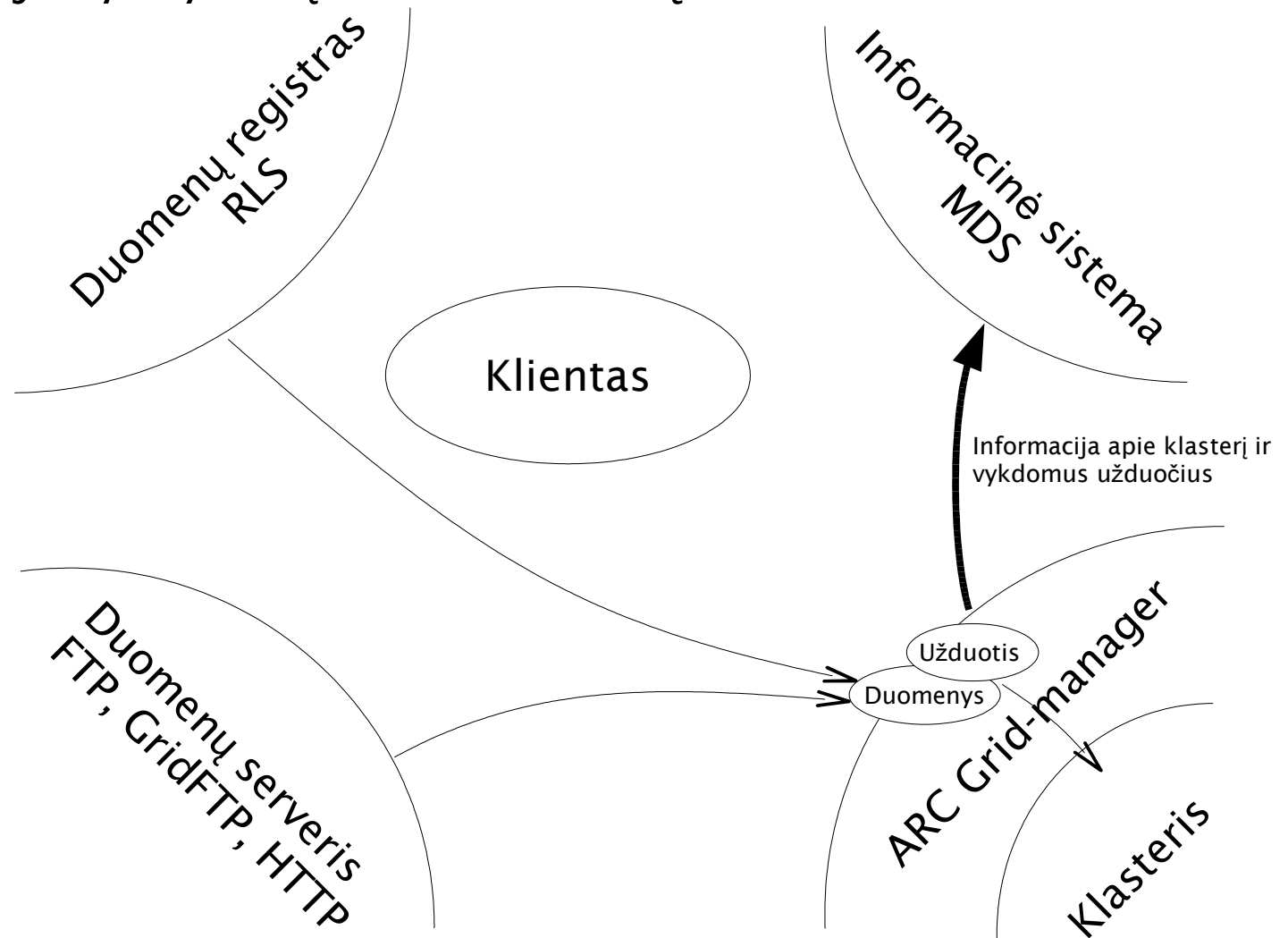
ARC - Kaip veikia

- Už duotis nukreipiama į klasterį



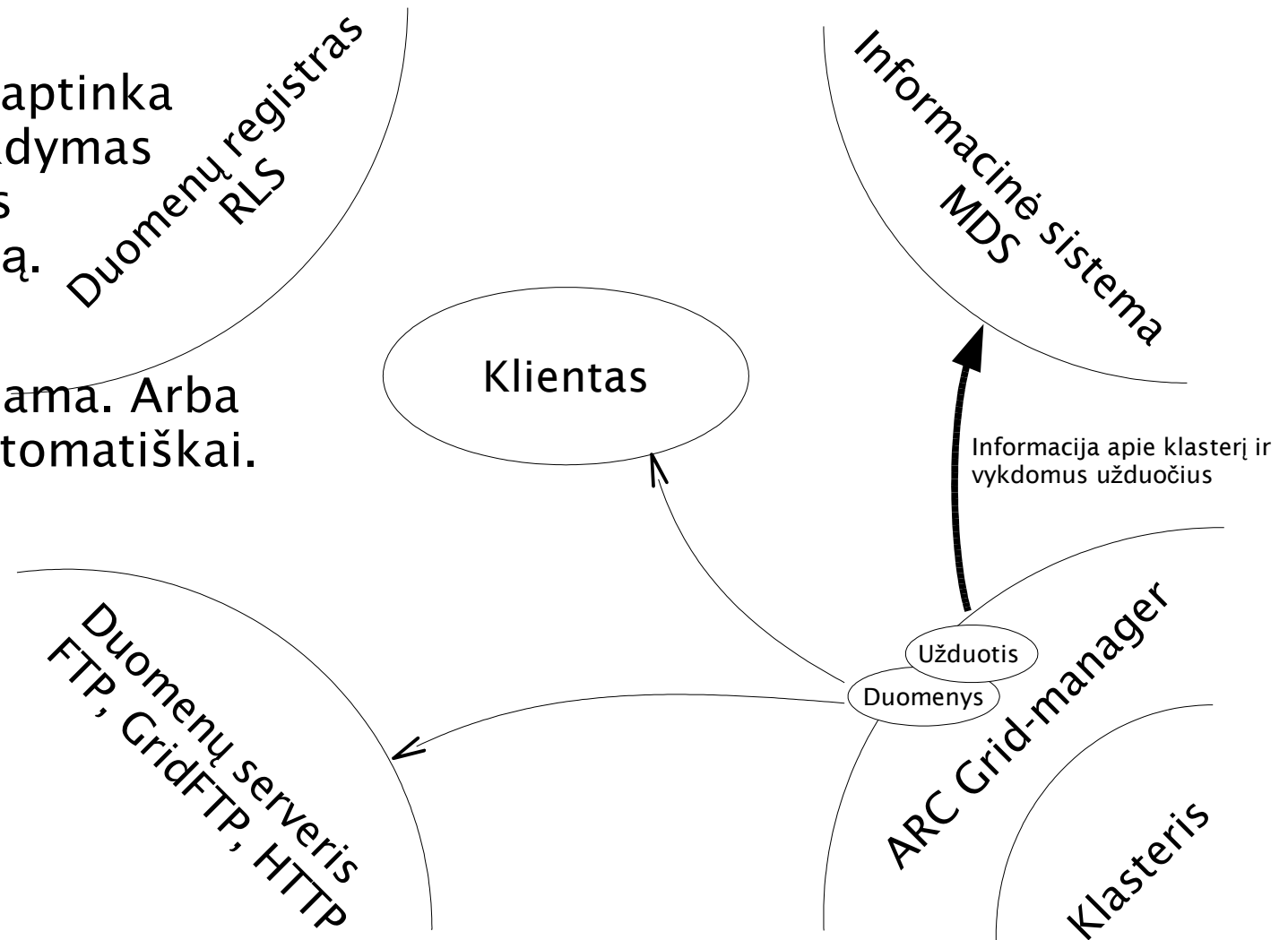
ARC - Kaip veikia

- Duomenys surenkami prie užduoties
- Už duoties nukeliauja vykdymui į klasterio vidinę aplinką



ARC - Kaip veikia

- Po vykdymo užduoties sukurti duomenys dalinai perkeliama į duomenų serverį ir dalinai paliekami klasteriye.
- Po to kai klientas aptinka kad užduoties vykdymas baigtas, duomenis perkeliama į klientą.
- Užduotis panaikinama. Arba bus panaikinta automatiškai.



ARC priima užduočius parašytus xRSL (Resource Specification Language) kalba.

&(atributas=verte)(atributas=verte)...

Dažniausiai naudojami atributai:

- **executable** – pagrindinė vykdomoji rinkmena
- **arguments** – vykdymo parametrai
- **cputime** – per kiek laiko užduoties vykdymas pasibaigs
- **memory** – kiek atminties reikia užduočiai
- **jobname** – vardas suteikiamas užduočiai
- **runtimeenvironment** – programa reikalinga užduoties vykdymui (RE)
- **stdin, stdout, stderr** – UNIX standartinių srautų nukreipimas į rinkmenas
- **inputfiles** – įėjimo duomenų sąrašas
- **outputfiles** – išėjimo duomenų sąrašas
-

Kiekvieną rinkmeną aprašo 2 elementai:

- vardas užduoties vykdymo metu
- šaltinis/tikslas

Šis žingsnis jau atliktas

- Vartotojo sąsajos įdiegimas
 - Nepriklausomas vartotojo sąsajos paketas yra prieinamas <http://ftp.nordugrid.org/download/> - *releases - standalone*.
 - Išpakuokite

```
$ tar -zxvf nordugrid-standalone-0.5.24-1.i386.tgz
```

- Inicializuokite

```
$ cd nordugrid-standalone-0.5.24
```

```
$ source setup.sh
```

```
$ cd ..
```

Programinė aplinka paruošta naudojimui. Bet vartotojas turi būti identifikuojamas.

Šis žingsnis jau atliktas

- Vartotojui reikia identifikavimo sertifikato. Paprastai vartotojai turi kreiptis į savo šalies CA.
 - Sukurkite sertifikato reikalavimą
- \$ grid-cert-request -int*
- Atsakykite į užduotus klausimus:
 - slaptažodis
 - organizacija
 - vardas
 - Bus sukurtas katalogas *.globus* su slaptu raktu apsaugotu slaptažodžiu ir sertifikato reikalavimu *.globus/usercert_request.pem* .
 - Nusiųskite sertifikato reikalavimą į Jūsų CA karu su savo duomenimis.
 - Gautą atsakymą išsaugokite *.globus/usercert.pem* . Tai gali užtrukti keletą dienų.

Šis žingsnis jau atliktas

- Vartotojui turi būti leidžiama naudotis resursais
 - Paprasčiausiai tai padaryti prisijungiant prie kokios nors VO
 - Susisiekite su savo VO administratoriumi, kad tapti jos nariu
 - NorduGrid Guest VO gali būti panaudota bandymo tikslams.
 - Į leidžiami beveik visi
 - Tik labai nedaug resursų priima užduočius iš tos VO narių.

- Pradekite darbą nuo aplinkos inicializavimo

```
$ cd nordugrid-standalone-0.5.24
```

```
$ source setup.sh
```

```
$ cd ..
```

- Sukurkite „proxy“ sertifikata ir rakta

```
$ grid-proxy-init
```

- Jeigu jūsų užduoties atlikimui reikia ilgesnio negu 12 valandų laiko

```
$ grid-proxy-init -valid 24:00
```

- Kataloge *examples* yra keletas paruoštų užduočių

```
$ cd examples
```

- Paprasčiausia užduotis

```
$ cat hello.rsl
```

```
& (executable="/bin/echo")
```

```
(arguments="Hello Grid")
```

```
(stdout="out.txt")
```

- Be savo vykdomos rinkmenos. Naudojamas */bin/echo*, kuris yra beveik visose UNIX operacinėse sistemose.
 - Be įėjimo ir išėjimo duomenų. Standartinis srautas nukreipiamas į *out.txt*, kuris automatiškai priskiriamas prie išėjimo duomenų.
- Paleidžiame užduotį

```
$ ngsb -f hello.rsl
```

```
Job submitted with jobid
```

```
gsiftp://farm.hep.lu.se:2811/jobs/162941110665573478034132
```

- Tai gali užimti laiko, todėl kad klientas turi apklausti visus resursus ir rasti tinkamą.
- Pranešime nurodytas užduoties identifikatorius

- Už duoties vykdomą galima sekti pasinaudojus *ngstat* komanda

```
$ ngstat gsiftp://farm.hep.lu.se:2811/jobs/162941110665573478034132  
Job gsiftp://farm.hep.lu.se:2811/jobs/162941110665573478034132  
Status: FINISHED
```

- Galimi rezultatai

- ACCEPTED - užduotis priimta ir laukia
- PREPARING - užduoties įėjimo duomenys apdorojami
- INLRMS - užduotis vykdoma
- FINISHING - užduoties išėjimo duomenys apdorojami
- FINISHED - užduotis baigta

- Kol užduotis vykdoma jos standartinius srautus galima peržiūrėti

```
$ ngcat gsiftp://farm.hep.lu.se:2811/jobs/162941110665573478034132  
Hello Grid
```

- Po užduoties pabaigos visus rezultatus galima atgauti su

```
$ ngget gsiftp://farm.hep.lu.se:2811/jobs/162941110665573478034132  
ngget: downloading files to /home/user/examples/162941110665573478034132  
ngget: download successful - deleting job from gatekeeper.
```

- Pabandykite patobulinti pavyzdį
 - Pabandykite panaudoti `ngsub -d 1 -f hello.rsl`. Tai parodys su kuriais resursais bendrauja `ngsub` ir kaip pasirenka.
 - Pridėkite (`jobname="Test"`) į užduoties aprašymą (hello2.rsl) ir pabandykite panaudoti užduoties pavadinimą vietoj identifikatoriaus.
 - Pridėkite (`gmlog="log"`) į užduoties aprašymą (hello3.rsl) ir pažiūrėkite, kas atsirado užduoties rezultatuose.
 - Išbandykite visas anksčiau minėtas komandas su `-h` opcija.

Pabandykite užduotį, kuri turi praktinę prasmę.

- GAMESS - The General Atomic and Molecular Electronic Structure System - kvantinės chemijos programa.

- RSL - gamess.rls

```
&(executable="$GAMESS_LOCATION/rungms")metu
```

```
(arguments=gamess.inp)
```

```
(stdout="stdout")(stderr="stderr")(gmlog="gmlog")
```

```
(inputfiles=("gamess.inp" ""))
```

```
(runtimeenvironment="APPL/CHEM/GAMESS")
```

```
(cputime="1 h")
```

```
(jobname="GAMESS")
```

- Įėjimo rinkmena gamess.inp turi GAMESS komandų, kurie sudaro standartinį testavimo užduotį.

- Tai testavimo užduotis, todėl mums reikia tik standartinių srautų. Patikrinkite, ar sukurtose rinkmenose nėra pranešimų apie klaidas.

Kelias į vykdomą rinkmeną nurodomas per kintamąją apibrėžta RE inicializavimo metu

Įėjimo rinkmena be šaltinio bus perkelta iš vartotojo kompiuterio į užduoties darbinį katalogą

Užduotis reikalauja programos GAMESS

Reikalaujame vienos valandos. Tai daug daugiau negu reikia. Bet užtikrina, kad užduotis nebus nutraukta.

ARC - Sudetinga užduotis

- Išbandykime užduotį, kurį turi įėjimo ir išėjimo duomenis, savo vykdoma rinkmena ir naudoja programą įdiegtą skaičiavimo resurse.

- Vykdoma rinkmena - *Bash script* povray.sh

```
#!/bin/sh
povray +H600 +W600 +rmonolith.png monolith.pov
```

Paleidžia pagrindinę programos rinkmeną su nurodytais parametrais

- RSL - povray.rsl


```
&(executable=runpov.sh)
(runtimeenvironment=POVRAY)
(inputfiles=
("monolith.pov"
"http://www.nordugrid.org/tutorial/dapsys_tutorial/povraydemo/monolith.pov")
("front.png"
"http://www.nordugrid.org/tutorial/dapsys_tutorial/povraydemo/front.png"))
(outputfiles=("monolith.png" ""))
(stdout=out.txt)(stderr=err.txt)
(gmlog=logs)
(cputime=1h)
(jobname=povray)
```

Vykdoma rinkmena bus paimta iš vartotojo kompiuterio ir perkelta į užduoties darbinį katalogą

Užduotis reikalauja programos POV-Ray

Įėjimo rinkmena bus perkelta iš nurodyto šaltinio į užduoties darbinį katalogą

Išėjimo rinkmena bus saugoma klasteryje užduoties darbiniam kataloge

- Visos sistemos ir atskirų užduočių veikimą galima stebėti per *Web* sąsają.
<http://www.nordugrid.org/monitor/>
- Pagrindinis langas rodo visų skaičiavimo resursų sąrašą ir jų užimtumą
- Raskite save
 - Pasinaudokite *gridcert-info -subject*, kad sužinoti savo vardą.
 - Pasirinkite  tam, kad gauti visų vartotojų sąrašą
 - Raskite savo vardą sąrašė
 - Naujame lange jus matysite prieinamų resursų ir vykdomų užduočių sąrašus.

Paeksperimentuokite su sąsaja. Beveik visus elementus galima spausti.

ARC - Sudetinga užduotis

- Kai užduotis pasibaigs, gausite rezultatus *ngget povray*
- Rezultatus sudaro 3 rinkmenos
- out.txt – standartinis išėjimo srautas
- err.txt – standartinis klaidų srautas
- monolith.png – sukurtas paveikslukas

Šitas pavyzdys yra suprastintas variantas sudėtingesnio pavyzdžio sukurto Leif Nixon. Pastarasis taip pat rodo, kaip gali suskirstyti užduotį į eilę paralelinių užduočių, kurie generuoja paveiksluką dalimis. Jis pasiekiamas

<http://www.nsc.liu.se/~nixon/ng-povray/>