

TARTU ÜLIKOOL
Füüsika-keemiateaduskond
Infotehnoloogia eriala

Raido Türk

OGCE
(Open Grid Computing Environments)

Bakalaureusetöö (4 AP)

Juhendaja: prof. Eero Vainikko

Autor: „.....” mai 2006

Juhendaja: „.....” mai 2006

Tartu 2006

Sisukord

Sissejuhatus	4
Mõisted.....	6
1 OGCE olemus.....	8
2 OGCE koostisosad	9
2.1. JSR 168 portletid.....	9
2.2 Portaali raamistikud.....	11
2.2.1 uPortal.....	11
2.2.2 GridSphere.....	11
2.3. Apache Tomcat	11
2.4. Sakai portletid	12
2.5. GPIR sirvimise portlet	12
3 OGCE installeerimine	14
4 Portletite API.....	15
5 OGCE pluginid.....	17
5.1. Maven GridPort plugin	17
5.2. JSR 168 portleti Maveni plugin	17
6 OGCE abivahendid.....	18
6.1. PURSe kasutajakonto registreerimise portletid	18
6.2. GridPorti Ühine Portleti Moodul	18
7 Seotud teenused.....	19
7.1. GPIR	19
7.2. CFT	19
7.3. Globus Toolkit	20
7.4. Java CoGKit.....	20
7.5. SRB	20
7.6. Condor.....	22
7.7 NCSA abivahendid	22

7.8. Tupelo	22
8 Tuntumad OGCE portaalid.....	24
Kokkuvõte.....	26
Resüme (inglise keeles)	27
Kasutatud kirjandus.....	28

Sissejuhatus

1970-ndate aastate alguses tekkis koos kohtvõrkude loomisega idee vaba arvutusressursi ära kasutamiseks. 1990-ndate aastate lõpupoole kasvas paljast ideest välja globaalne projekt: GRID, tänu millele on võimalik teaduslikes arvutusteks ära kasutada ühisesse arvutivõrku ühendatud arvutite vaba arvutusressurssi.

Veel paar aastat tagasi oli GRID müstiline sõnadelühend, mille tähendust ja tähtsust mõistsid vähesed inimesed, kuid viimastel aastatel on GRID-i tehnoloogia tuntus kõvasti kasvanud ja seda tänu paljudele projektidele. Maailma kõige tuntum GRID-i projekt on arvatavasti SETI@home, mis otsib intelligentset elu väljaspool maad, analüüsides selleks kosmosest tulnud signaale, otsides raadioülekandeid. Selline analüüs nõuab tohutut arvutusressurssi, mis saavutatakse läbi Gridvõrku ühendatud kasutajate.

Koos GRID-i tehnoloogia arenemise ja GRID-i süsteemide arenguga kasvab vajadus ka paremaks GRID-i juurdepääsuks. Üha rohkem hakkavad GRID süsteeme kasutama tavalised inimesed, kes ei ole väga kogenud GRID-i alal. Neile on tähtis ainult see, et nad saaksid oma tööd GRID-is arvutatud. Kuna enamik neist ei taha tegemist teha ei käsurea, ega Linuxiga, siis suureneb järjest vajadus kasutajasõbralike GRID-ide vastu. Portaalid pakuvad ligipääsu GRID-i tehnoloogiale läbi jagatud ja taaskasutatavate komponentide, võimaldades teha standardseid GRID-i toiminguid ja jälgida GRID-i tööde seisu. Portaalide eeliseks on see, et nad on tavakasutaja jaoks kasutatavad ükskõik mis platvormil, kuna kasutajatel on vaja ainult internetilehitsejat ja internetiühendust.

Kõnealune töö on suunatud inimestele, kes omavad vähemalt baastadmisi GRID-st ja programmeerimisest. Käesoleva uurimistöö eesmärgiks on anda ülevaade eelkõige OGCE portaalide olemusest ja otstarbekusest ning selgitada välja OGCE otstarbekus. Ülevaade antakse tuginedes OGCE 2 Release 2.0 versioonile, mis ilmus 1. mail 2006 aastal (lihtsuse huvides kasutatakse edaspidi OGCE lühendit). Seejuures analüüsitakse OGCE koostisosi, antakse ülevaade OGCE kasutamisest ning tutvutakse erinevate pluginate ja

abivahendite võimalustega. Samuti antakse ülevaade tuntumatest ja suurematest OGCE portaalidest.

Töö baseerub täielikult internetis leidunud ingliskeelsetel materjalidel, mis peaks lihtsustama lisainformatsiooni leidmist teemast lähemalt huvitatutele ning keelt valdavatele inimestele. Põhiline osa töö materjalist on pärit OGCE ametlikult kodulehelt [1]. Lisaks kodulehel olevale informatsioonile on väga palju kasutatud ka OGCE meeskonna poolt tehtud esitluste materjale. Eestikeelsetele nimetustele ja terminitele järgnevad sulgudes toodud vastavad ingliskeelsed nimetused. Töös on kasutatud mitmeid jooniseid eesmärgiga anda parem ülevaade puudutatud teemadest. Selguse huvides ei tõlgitud võõrkeelset teksti joonisel eesti keelde.

Töö struktuurist

Töö koosneb 8 peatükist. Lisaks on veel eraldi peatükk mõistetele.

Esimeses peatükis antakse ülevaade OGCE olemusest ning kasutamisevõimalustest [1-3].

Teine peatükk kirjeldab lähemalt OGCE koostisosasid ning nende omadusi [4-14].

Kolmandas peatükis antakse ülevaade OGCE installeerimisnõuetest [2].

Neljandas peatükis tutvustatakse portletide API võimalusi [16].

Viiendas peatükis räägitakse OGCE-d puudutavatest pluginitest [1,17].

Kuuendas peatükis räägitakse OGCE-ga seotud abivahenditest [1,18].

Seitsmendas peatükis kirjeldatakse OGCE-ga seotud teenuseid [19-27].

Kaheksandas tutvustatakse tuntumaid OGCE-l põhinevaid portaaale ja nende kasutusalasid [28-35].

Mõisted

- *API* – rakenduse programmeeritav liides, mille peamiseks eesmärgiks on kirjeldada rakenduse funktsioonidele ligipääs.
- *Cluster* – grupp omavahel lähedalt ühendatud arvuteid, mis töötavad üheskoos ja näivad väliselt ühe suure arvutina.
- *Cookie* – tekstipakk, mis saadetakse serverilt veebilehitsejasse ja veebilehitseja poolt muutumatult tagasi iga kord kui veebilehitseja avab selle serveri. Neid kasutatakse kasutajate kindlakstegemisel
- *Credential* – GRID-s kindlakstegemisel kasutatav informatsioon, mille alusel saab GRID-i kasutaja ligipääsu GRID-i süsteemidele.
- *Deployment* – tarkvarasüsteemi kasutusvalmis tegemise protsess.
- *Framework* – defineeritud raamistik, mille peale on võimalik organiseerida ja luua tarkvaraprojekt.
- *GRID job*- GRID-i töö, mis on juba GRID-ile arvutamiseks antud või järjekorda pandud.
- *Fragment* – kild, osa tervikust.
- *JavaServer Pages* – Java tehnoloogia, mis lubab arendajatel dünaamiliselt genereerida HTML, XML või mingit teist tüüpi dokumente vastusena veebikliendi päringutele. Tehnoloogia lubab Java koodi ja kindlaid eeldefineeritud tegevusi kasutada staatilise sisu osana.
- *Java Virtual Machine* – virtuaalmasin, mis käivitab Java baitkoodi. JVM on Java platvormi kõige olulisim osa, mis lubab paljudel tarkvaraplatvormidel funktsioneerida nii vahevarana kui ka iseseisva platvormina.
- *Library* – tarkvaraarenduseks mõeldud alamprogrammide kogum.
- *Listener* – ootab rakenduses mingi tegevuse ilmnemist ja tegevuse ja siis reageerib vastavalt.
- *Middleware* – tarkvarakiht, mis asub operatsioonisüsteemi ja rakenduse vahel.

- *Pattern* – üldine lahendus, mis võimaldab lahendust tihti esinevatele probleemidele tarkvararakendustes. Arendaja valitud tehnoloogiatega kasutatakse mustrit probleemi lahendamisel, näiteks veebikihile suhtlemise võimaldamiseks ärikihiga.
- *Web Application*- veebirakendus, mis on veebilehitsejaga üle võrgu ligipääsetav.
- *Presentation layer* – kasutajatele kuvatav kiht.
- *Provider* – isend, mis pakub teenuseid teistele isenditele.
- *Schema* – dokumendi standardi struktuuri kirjeldus.
- *Servlet* – Java platvormil arendatav aktiivne veebikomponent, mis võimaldab kuvada dünaamilist sisu.
- *Storage*- arvutikomponent, mis hoiab ja säilitab andmeid (kõvakettad, optilised kettad, magnetlindid).
- *Task handler* – GRID tööde haldur.
- *Web Services for Remote Portlets* – veebiportaalide standard kaugserveris asuvatele portletidele ligipääsuks ja kuvamiseks.
- *Workflow* – dokumentide ja ülesannete läbi tööprotsessi liigutamise operatsiooniline ülevaade.

Peatükk 1

OGCE olemus

OGCE (inglise k. *Open Grid Computing Environments*, järgnevalt on kõik ingliskeelsed terminid toodud kursiivis) arendab JSR-168 spetsifikatsiooniga ühilduvaid portlete ja veebiteenuseid, mille abil saab ehitada teadusportaale. OGCE ühendab endas paljusid Põhja-Ameerika GRID-i temaatilisi projekte.

OGCE võimalused :

- Tagab avatud lähtekoodiga laiendatava arhitektuuri, mida saab tänu suurele kasutajaskonnale pidevalt täiendada.
- Võimalus arendada GRID-i kliendipoolseid rakendusi, mis oleksid kasutatavad paljudel GRID-i erinevatel versioonidel.
- võimalus kasutada olemasolevaid portlete, mis on lihtsalt laiendatavad ja sobitatavad kasutaja enda oludele.

OGCE pakub kasutajatele:

- Interaktiivseid isiklikke ja grupikalendreid.
- Suhtlemiseks foorumeid.
- Otse (*live*) jutustamise teenuseid ja kasutajaliideseid.
- Grupipõhist dokumendijagamist.
- Lehitsejapõhist kasutajaliidest GRID-i tööde käivitamiseks ja jälgimiseks kõikjalt.
- Failide ringiliigutamist
- Arvutiressursi olukorra jälgimise võimalust

Peatükk 2

OGCE koostisosad

2.1. JSR 168 portletid

Portlet (*portlet*) on Java tehnoloogial baseeruv veebikomponent, mida haldab portleti konteiner (*portlet container*), mis hoolitseb portleti elutsükli ja –ea eest, käivitades ja hävitades portlete, kutsudes välja portleti meetodeid. Portlet edastab päringuid ja loob dünaamilist sisu. Portleti poolt loodud sisu nimetatakse fragmendiks (*fragment*), märgendi (näiteks HTML, XHTML) osaks, mis allub kindlatele reeglitele. Erinevalt Java servletitest (*servlets*), mis genereerivad juba valmis dokumente, loovad portletid ainult ühe osa tervikust, mida hiljem saab ühendada üheks tervikuks. Portlete kasutatakse portaalides kasutajaliideselementidena, mis loovad esitluskihi (*presentation layer*) informatsioonisüsteemidele.

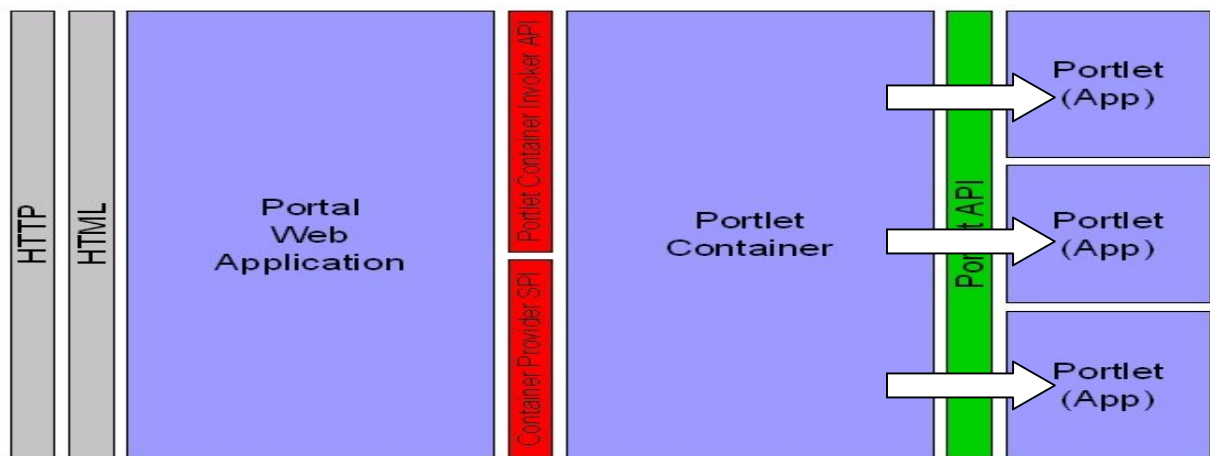
JSR (*Java Specification Request*) 168 on portleti spetsifikatsioon.

JSR-168 eesmärk:

- Luua keskkond (portleti konteiner) portletite jaoks.
- Defineerida API (*Application Program Interface*) portleti ja portleti konteineri vahel.
- Tagada võimalused portletite lühiaegsete ja püsivate andmete hoidmiseks.
- Luua mehhanism, mis lubab portletitel kaasata servletid ja JSP (*JavaServer Pages*) leheküljed.
- Standardiseerida portletite programmeerimisviisid.
- Kergemaks teisaldatavuseks luua portletide pakettidesse panemise reeglistik
- Binaarsete portletite teisaldatavuse lubamine JSR-168 portaalide piires.
- JSR 168 portletite jooksutamine kaugportletitena kasutades WSRP (*Web Services for Remote Portlets*) protokoll.

Portletidel on väga piiratud viis suhtlemiseks enda konteineritega: portletite API on peamiselt ühesuunaline ja portletid ei saa suhelda enda konteineritega. JSR-168 spetsifikatsioonis on portletikonteineriks Pluto. [5]

Joonis 1 kujutab GRID portaali põhilist arhitektuuri. Portaali veebirakendus (*Portal Web Application*) edastab kliendi päringu, otsib üles praegusel kasutaja lehel kuvatavad portletid ja kutsub välja portleti konteineri, et saada kätte iga portleti sisu. Portaal siseneb portleti konteinerisse Portleti Konteineri Käivitamise API (*Portlet Container Invoker API*) kaudu, mis tagab ligipääsu portleti konteineri liidese meetoditele, mille kaudu saab portaal portlete välja kutsuda. Et portaal saaks kätte portletidelt tuleneva informatsiooni, tuleb implementeerida ka Portleti Konteineri Pakkuja SPI (*Container Provider Service Provider Interface*) tagasikutsumis (*callback*) liides. Lõpuks kutsub portleti konteiner kõik portletid välja portleti API kaudu.



Joonis 1. Portaali arhitektuur [5].

JSR-168 portleteid kasutatakse portaalides. Veebiportaal (*web portal*) on dünaamiliste lehtede kogumik, mis võimaldab luua isikustatud (*personalized*) veebisisu, tagab ligipääsu spetsiifilistele võimalustele, nagu näiteks RSS uudiste vood ning mäletab ka hiljem kasutajate eelistusi, kas siis läbi küpsiste (*cookie*) või logimise kaudu. Portaal kuvab portleti konteinerite poolt edasi antud portletite sisu ja vastutab tegeliku portaali kujunduse eest.

GRID portaal on abivahend, mille kaudu saab GRID-i tööülesandeid juhtida veebilehitseja kaudu. GRID portaal võimaldab käivitada ja jälgida töid, liigutada faile, käivitada teenustel baseeruvaid teadusprogramme.

2.2 Portaali raamistikud (*Portal frameworks*)

OGCE portletite peamiseks kasutamiseks on mõeldud 2 vabavaralist portaaliraamistikku: uPortal ja GridSphere. Lisaks võib kasutada ka erinevaid teisi vabavaralisi ja kommerts portleti konteinereid. Ainuke nõue on, et nad oleksid JSR-168 standardiga ühilduvad.

2.2.1 uPortal

uPortal [10] on vabavaraline portaaliraamistik, mis toetab JSR-168 standardiga ühilduvaid portlete. uPortal kasutab portleti kontaineriks Plutot.

2.2.2 GridSphere

GridSphere [11] portaali raamistik pakub avatud lähtekoodi portletitel põhinevat veebiportaali, mis on JSR-168 standardiga ühilduv. GridSphere portleti konteiner võimaldab jooksutada ja administreerida kolmandate osapoolte portleti veebirakendusi.

2.3. Apache Tomcat

Apache Tomcat [13] on Apache Software Foundationi poolt loodud veebiserver Java jaoks, mis toetab servleteid ja JSPsid (*JavaServer Pages*). Kuna Tomcat on loodud Javas, jookseb ta igal platvormil, mida toetab JVM (*Java Virtual Machine*). OGCE portaal jookseb Tomcat serveril.

2.4. Sakai portletid

Sakai projekt (*Sakai Project*) arendab vabavaralist hariduslikku tarkvara. Sakai arendatavad portletid pakuvad ligipääsu sellistele abivahenditele, nagu kalendri ajakavastamine, dokumentide jagamine ja vestluse (*chat*) funktsioonid.

Sakai projekti saab kasutada nii iseseisva süsteemina kui ka integreeritult teiste portleti konteineritega, nagu uPortal, GridSphere ja Pluto sisse, sest Sakai projekt sisaldab veebiteenuseid ja portleti standardiga ühilduvaid osi.

2.5. GPIR sirvimise portlet

GPIR sirvimise portlet pakub kasutajale võimalusi jälgida GPIR (*GridPort Information Repository*) teenuse poolt saadud GRID-i ressursse. GPIR sisaldab nii GRID-i staatilisi kui dünaamilisi ja ka süsteemi andmeid. Staatilistes andmetes sisalduvad andmed ressursi nime, süsteemi tüübi, protsessorite arvu, maksimumjõudluse, mälu ja kõvakettaruumi kohta. Dünaamilistes andmetes on näha ressursi staatus, koormus ja jooksvate tööde arv.

Portlet võimaldab vaadelda GPIR-i informatsioonis kahel viisil:

- Masina vaade: kuvatakse masina spetsiifiline informatsioon. GPIR sirvimise portlet lubab kasutajal ligi pääseda väga laialdasele masinat puudutavale informatsioonile. Kasutaja võib vaadelda masina staatilist informatsiooni, ressursi päeva teadet ja kõiki järjekorras olevaid GRID-i töid.
- Süsteemi vaade: Võetakse kokku kõigi masinate staatiline ja dünaamiline info.

GridPort Demo Portal

http://localhost:8080/gridsphere/gridsphere.jsessionid=69C5DD4FE13868DEC66EEC12C32EDCCF/cid=login

About the GridPort Toolkit - Grid... CSS Background Properties GridPort Demo Portal

GridPort Logout
Welcome, Root User

Grid Information Proxy Management File Management Job Submission Control Job Submission Welcome Administration

GPIR Browser

Grid Information Browser

Refresh

Parallel Computing Resources										
Name	Institution	System	CPUs	Peak GFlops	Memory GBytes	Disk GBytes	Status	Load	Jobs	
TestResource8000	Test University	Test System	555	5555	555	55000	↓		X-Q-O	
Total:			555	5555	555	55000				

High Throughput Computing Resources							
Name	System	Active PCs	Active CPUs	Memory GBytes	Disk GBytes	Resource Details	Jobs
TestResource11000	Test System	0 / 0	0 / 0	0	0	Q	Q
Total:		0 / 0	0 / 0	0	0		

Visualization Resources									
Name	Institution	System	CPUs	Peak GFlops	Memory GBytes	Disk GBytes	Peak Polygons	Graphics HW	Status
TestResource10000	Test University	Test System	5	55	55	555	55	Q	↓
Total:			5	55	55	555	55		

Storage Resources					
Name	Institution	System	Online Storage	Offline Storage	Status
TestResource9000	Test University	Test System	555	555	↓
Total:			555	555	

October 6, 2005

Done

Joonis 2. GPIR sirvimise portleti näide [1]

Peatükk 3

OGCE installeerimine

OGCE 2.0 pakett sisaldab kõike, mida on vaja portaali installeerimiseks, välja arvatud Java. Süsteem on täielikult testitud Java 1.4.2 peal, Java 1.5 toetust on testitud ainult põhiportletitel ja -teenustel. Sakai teenuste kokkuehitamisel võib tekkida Java 1.5-ga probleeme.

OGCE 2.0 on disainitud töötama 2-3 eraldi Tomcat serveril.

1. Portaali server jookseb Apache Tomcat serveril vaikeväärtusena 8080 pordil. See server jooksutab portaali ja portlete.
2. GridPort server jooksutab GridPort's GPIR ja CFT teenuseid. Server on konfigureeritud kasutama porti 7070. Enne 2.0.1 versiooni oli kasutusel port 9090.
3. Sakai server jookseb Sakai teenusel pordil 10010. See on ka konfigureeritud jooksma Tomcatil koos vajalike JVM mälu seadistustega.
Seoses Sakai kõrgete mälunõuetega soovitatakse Sakaid jooksutada eraldi serveril.

Portaali ja veebiteenuste eraldi Tomcati serveritel jooksutamine lihtsustab *Jar* failide konflikte ja lubab jaotada OGCE läbi paljude erinevate serverite.

(Täpsemalt vt. OGCE 2.0 installeerimise juhendist [2])

Peatükk 4

Portletite API

OGCE projekti osana loiid Java CoG Kit arendajad GRID-i abstraktse kihi, et kasutajatel oleks võimalik sooritada tavalisi GRID-i ülesandeid. GRID-i abstraktne kiht peidab erinevate GRID-i abivahendite tootjate ja versioonide erinevused.

Abstraktsioonimoodul baseerub GRID-i abstraktsioonimudelil, eraldades GRID-i mustrid (*patterns*) nende implementatsioonidest. Arendatud rakendused on sõltumatud madalama taseme GRID-i implementatsioonidest. Seega on rakendused kaitstud madalama taseme GRID-i protokollide ja kasutajaliideste implementatsioonidest.

Abstraktsioonide moodul pakub GRID-i rakenduste programmeerijaile järgnevaid võimalusi:

- arendada kliendi rakendusi, mis on kasutatavad paljude GRID-i tagaplaani implementatsioonidega.
- tagab taaskasutatavat koodi, et toetada kiiret GRID-i ligipääsumustrite prototüübistamist.
- tagab vabavaralise ja pidevalt areneva arhitektuuri, mida saab ehitada kollektiivselt vastavalt GRID-i kogukonna tagasisidele.
- võimalus ligi pääseda hulk kasutajaliideseid implementeeritud kardinaalselt erinevates tehnoloogiates.

Java CoG Kit-i abstraktsioonid pakuvad programmeerimismudelit, mis toetab põhilisi GRID-i mustreid, nagu tööde (*jobs*) käivitamine, failide ülekanne ja failide operatsioonid.

Iga GRID-i tööd näidatakse kui GRID-i ülesannet. Kõik tööga seotud detailid näidatakse ülesande spetsifikatsioonina. Kaugkäivitused ja failiserverid näidatakse teenuse (*service*) objektidena. Igal teenusel on varustaja (*provider*) atribuut seotud teenusega, mis näitab, milliste tehnoloogiatega teenus on implementeeritud.

GRID-i töö käivitamiseks on vaja kasutajal luua abstraktne GRID-i ülesanne ja lisada ülesandele spetsifikatsioon ja teenus. Seejärel saadetakse ülesanne ülesande haldajale (*task-handler*), mis loeb spetsifikatsiooni detailid ja sõltuvat varustaja teenuse atribuudist tõlgib need protokollspetsiifilistesse tagaplaani teenusestruktuuridesse.

Seejärel saadetakse ja käivitatakse ülesanne asünkroonses töörežiimis. Seega ei tule kliendil oodata ülesande valmimist ning ta võib jätkata teiste tegevustega. Ülesande valmisest või tõrkest antakse asünkroonselt teada.

Üks kõige tavalisem kasutusjuht Griidiarvutustes on GRID-i ülesande käivitamine (töö saatmine, failide ülekanne või informatsiooni päring). Laiendus tavalisele GRID-i mustriks on GRID-i töövoos (*workflow*) muster, mis lubab kasutajal saata komplekti GRID-i ülesandeid koos käivitamissõltuvustega. Seega abstraktsioonide raamistiku (*framework*) algne disain keskendub elementide pakkumisele, mida on vajava tähtsate kasutusmustrite toetamiseks. Teisi GRID-i mustreid toetatakse laiendades kohanemisvõimelisi abstraktsioonide, tuginedes GRID-i kogukonna tagasisidele.

(Täpsem info CoG Kit-i juhendist [16])

Peatükk 5

OGCE pluginid

Maven on projekti halduse tööriist, mille peamiseks ülesanneteks on:

- Lihtsustada rakenduse ehitamise protsessi
- Tagada ühtne ehitamissüsteem
- Edastada kvaliteetset informatsiooni projekti kohta
- Pakkuda raamistikku paremaks arendustegevuseks
- Tagada uute võimaluste nähtamatu migratsioon

Võrreldes Apache Ant-iga on Mavenis palju rohkem sisseehitatud võimalusi. Maveni peamiseks eeliseks on koodi vähesus, mis tuleb kirjutada, võimaldamaks teha tavalisi ülesandeid, nagu koodi kompileerimine, javadoc-i loomine ning war faili ehitamine.

5.1. Maven GridPort plugin

Maven GridPort plugin sisaldab OGCE ja GridPort portaalide ehitamiseks ja paigaldamiseks (*deploy*) vajalikku funktsionaalsust. Plugin loodi, et mitmed portaalid saaksid kasutada sama ehitamis- ja paigaldamiskoodi. Plugin sisaldab erinevaid käske, mis installeerivad, paigaldavad ja puhatavad portaali ja tema komponente. Lisaks sellele on kaasatud ka funktsionaalsus portaali väljalaskmiseks, veebilehe ja projekti dokumentatsiooni uuendamiseks

5.2. JSR 168 portleti Maveni plugin

Portleti plugin abistab portleti rakenduse kokkupakitud faili (näiteks WAR faili) loomisel sobivale portleti konteinerile. Praegusel hetkel toetab plugin ainult uPortal-i ja GridSphere raamistikku.

Peatükk 6

OGCE abivahendid

6.1. PURS kasutajakonto registreerimise portletid

PURS (*Portal-based User Registration System*) pakub teenuseid, et lubada kasutajal ja portaali administraatoritel hallata GRID-i tunnistuste (*credential*) päringuid läbi veebi. PURS on kasutajate registreerimise süsteem veebirakenduste jaoks, mis kaustavad GRID-i turvalisuse infrastruktuuri (*Grid Security Infrastructure*), mis baseerub PKI-l ja X.509 sertifikaadil. PURS koordineerib uute kasutajate GRID-i turvalisuse sertifikaatide tuvastamise protsessi, logides sisse rakenduse veebilehel. Kasutajad, kellel juba on olemas kehtiv sertifikaat saavad samuti kergelt registreerida portaali rakendustes. PURS varjab veebirakenduste kasutajate eest X.509 sertifikaati komplitseeritud haldamise ja lubab kiiret kasutajate registreerimist, säästes seega kasutajat GRID-i tunnistustega tegelemast.

OGCE PURS portletid on JSR 168-ga ühilduvad pluginid, mida saab lisada GRID-i portaali. [11]

6.2. GridPort-i Ühine Portleti Moodul

Gridport-i Ühine Portleti Moodul (*GridPort Common Portlet Module*) on kasutuses paljude GRID-i portletite poolt, et sooritada harilikke ülesandeid GRID-il. Ülesanded sisaldavad GRID-i kasutajate tunnistuste kätte saamine, tööde haldamine ja failiülekannete id-de portaali andmebaasis säilitamist ja failinimekirjade ja faili üleslaadimiste käivitamine. Moodul sisaldab portletite poolt rakendatud java koodi ja lisaks ka portletite kasutatavaid veebirakenduse faile.

Peatükk 7

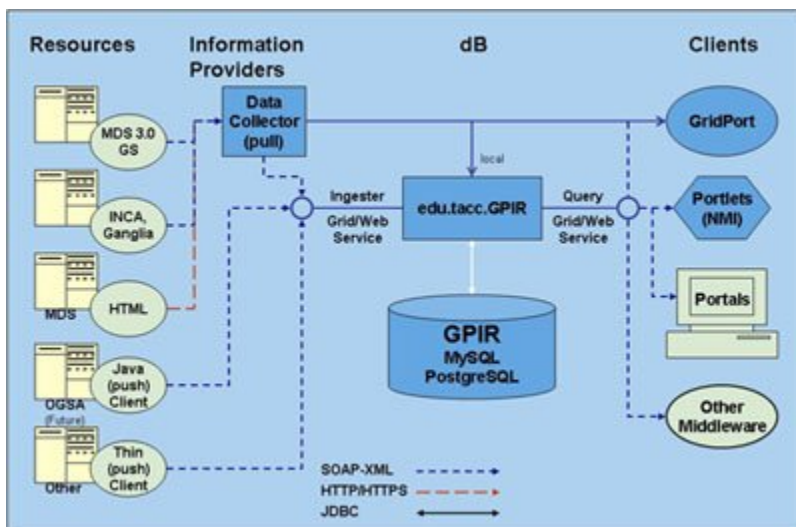
Seotud teenused

7.1. GPIR

GPIR (*The Grid Portal Information Repository*) on koht, kus hoitakse kergesti loetavat informatsiooni griidisüsteemi kohta.

GPIRis sisaldub kahte tüüpi informatsiooni:

- dünaamilised andmed, mida uuendatakse läbi GPIRIngester veebiteenuse ja mille lugemine toimub läbi GPIRQuery veebiteenuse (vt joonis 3).
- inimesekesksed andmed, mis on hallatavad läbi GPIR administratsiooni kliendi.



Joonis 3. GPIR arhitektuur [16]

7.2. CFT

CFT (*The Comprehensive File Transfer*) teenus võimaldab kliendirakendustel saata korraga mitmeid faile üksteisest eemal asuvate ressursside vahel. Failid saadetakse ühe suure hunnikuna ja CFT teenus hõlbustab väga suurte failide ülekannet, hoides alles

erinevat informatsiooni ülekannete kohta, sealhulgas staatust, ülekandekiirusi ja ülekandeaegu. CFT teenus on loodud veebiteenusena, seega väga paljud kliendirakendused saavad seda kasutada, sealhulgas OGCE. CFT teenus kasutab Globus GridFTP-d failide ülekannete tegemiseks.

7.3. Globus Toolkit

Globuse Tööriistakomplekt (*Globus Toolkit*) on avatud lähtekoodiga komplekt, mis sisaldab tarkvara teenuseid ja raamatukogusid (*libraries*) GRID-i arvutite ressursi jälgimiseks ja haldamiseks ning turvalisuse ja failide haldamiseks. . Globuses olevaid komponente saab kasutada nii üksi kui ka üheskoos.

Globuse Tööriistakomplekt on väga paljudes projektides aluseks, mille peale ehitatakse nii vabavaralisi kui ka kommertstooteid. Globuse põhiteenused, liidesed (*interfaces*) ja protokollid lubavad kasutajatele ligipääsu kaugressurssidele, mis paistavad kasutajatele kui kohaliku võrgu ressursid.

7.4. Java CoGKit

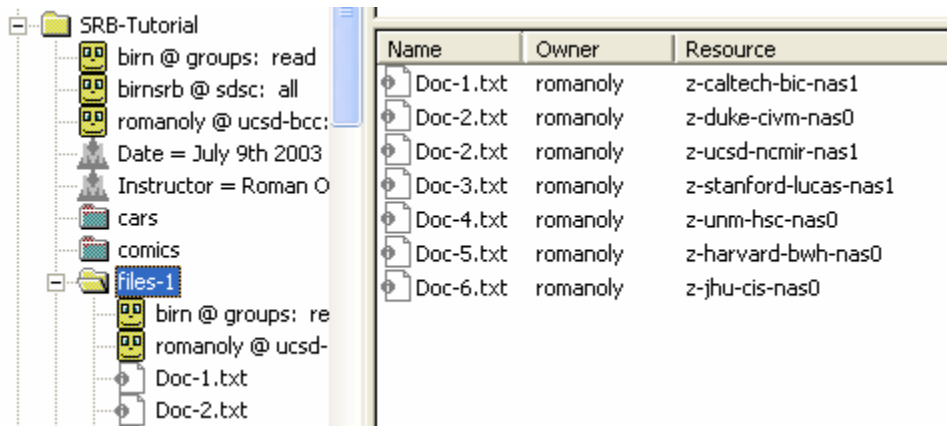
CoG komplekte (*Commodity Grid Kits*) kasutatakse portaalide ehitamiseks. CoG komplektid lubavad GRID-i kasutajatel, arendajatel ja administraatoritel kasutada, programmeerida ja hallata GRID-e kõrgema taseme raamistikust. Nad julgustavad kollaboratiivset koodi taaskasutust ja hoiavad ära mõttetu koodi dubleerimise probleemilahendamise keskkondades, teadusportaalides, GRID-i vahevaras.

7.5. SRB

SRB (*The SDSC Storage Resource Broker*) on klient-server vahevara (*middleware*), mis pakub üle võrgu ligipääsu erinevatele andmeladudele (*storage*) ning haldab metaandmeid iga salvestatud objekti kohta. SRB, koos MCAT-ga (*Metadata Catalog*) pakub võimaluse andmeobjektidele ligipääsuks läbi objektide parameetrite, hoolimata nende

füüsilisest asukohast või nimest. SRB esitab kogu süsteemi ühe loogilise failihierarhiana. MCAT andmebaas on metaandmete hoidla, mis hoiab infomatsiooni, mida kasutab SRB süsteem. Informatsioon hõlmab nii süsteemisisesid kui ka SRB poolt vahendatavaid andmeid.

Joonisel 4 on kujutatud Windowsis tehtud ekraanilõige, mis kujutab SRB näidisstruktuuri. Vasakul on näha failistruktuur, koos metaandmete ja õigustega iga kataloogi jaoks. Kuigi failid asuvad füüsiliselt erinevates ladustamissüsteemides, nagu näha ressursi (*Resource*) tulbast, ühendab SRB siiski failid süsteemselt üheks loogiliseks alamkataloogiks.



Joonis 4. SRB näidisstruktuur [21]

Suurte failide ülekannetel on SRB kiirus palju suurem kui FTP, SCP või NFS pakkuda suudavad, tänu SRB mitmelõimelisusele (mitu lõime saadavad korraga andmeid võrku). Väiksemate failide ülekanded võivad olla natuke aeglasemad, kuna SRB kulutab MCAT-ga suhtlemiseks lisa-aega.

SRB portletid pakuvad kasutajale järgnevad võimalused:

- Failide/dokumentide sisu vaatamine.
- Kataloogistruktuuri haldamine (struktuuris liikumine, kataloogide loomine, kataloogide kustutamine).
- Failide toimingud (ümbenimetamine, kustutamine, kopeerimine)
- Faile üleslaadida enda arvutist SRB serverisse.
- Faile allalaadida SRB serverist enda arvutisse.

7.6. Condor

Condor on töökoormuse haldamise tarkvarasüsteem väga suurt arvutusressurssi nõudvate tööde jaoks. Condor rakendab efektiivselt üle võrgu tööjaamade arvutusvõimsust. Condor suudab hallata tervet tööjaamade klastrit (*cluster*). Süsteem pakub tööde järjestamise mehhanismi, ajakava, prioriteetsuse skeemi, ressursi jälgimist ja ressursi haldamist.

Condor suudab efektiivselt rakendada ka tegevusetus (*idle*) seisus olevate lauaarvutite protsessori arvutamisevõimsust. Näiteks on võimalik Condorit konfigurida kasutama ainult neid lauaarvuteid, kus klaviatuur ja hiir on tegevusetus seisundis. Kui avastatakse, et klaviatuur või hiir pole enam tegevusetus seisus, siis suudab Condor luua tihti märkamatu kontrollpunkti ja suunata kogu töö mõnda teise tegevusetus olekus olevasse masinasse.

7.7 NCSA abivahendid

7.7.1 ELF

ELF on robustne konteiner, mis on disainitud toetama skriptitavaid rakendusi. ELF-i eesmärk on olla veakindel, ta ei tohi ootamatult ja vaikselt tööd lõpetada, samuti peab ta olema suuteline leidma vea põhjuse. ELF on ka jälgitav, kuna ta saadab pidevalt kuularile (*listener*) informatsiooni enda seisundi kohta. Samuti on võimalik ELF-i funktsionaalsust laiendada ilma uuesti kompileerimiseta.

7.7.2 OgreScript

OgreScript on OGRE edasiarendus. OGRE on Apache Ant-l põhinev XML skriptimiskeel. OgreScript on Javale ehitatud XML skriptimiskeel, mis on mõeldud jooksma ELF-i sees.

7.8. Tupelo

Tupleo on andmete ja metaandmete arhiveerimise süsteem, mis toetab RDF-OWL-l põhinevaid objektorienteeritud metaandmete raamistikke (*schemas*), loogilist failide

nimetamist, versiooni- ja ligipääsukontrolli ja GRID-i teenuseid. RDF-OWL veebi ontoloogia keel (*Web Ontology Language*) on disainitud rakenduste jaoks, mis peavad ka informatsiooni sisu töötleva, mitte lihtsalt kasutajale kuvama.

Tupelo sisaldab arvukalt unikaalseid võimalusi, sealhulgas:

- GRID-i teenuse kasutajaliides metaandmete ja andmeteenustega
- Esimese klassi metaandmete raamistikud, iseennast kirjeldavate metaandmete jaoks.
- RDF-OWL raamistike toetus.
- Versioonikontroll
- GSI autentimine ja objektitasemel ligipääsukontroll
- API, lisamaks uusi võimalusi

Peatükk 8

Tuntumad OGCE portaalid

- LEAD (*Linked Environments for Atmospheric Discovery* [28]) on 1. oktoobril 2003 alanud 5 aastane projekt, mille eesmärk on luua integreeritud raamistik meteoroloogiliste andmete kogumiseks ja ilnamudelite väljatöötamiseks, et oleks võimalik visuaalselt simuleerida erinevaid ilmastsenaariume.
- Põhja-Carolina Bioportaal (*North Carolina Bioportal* [29]) pakub ligipääsu bioinformaatika abivahenditele. Portaalil on laiendatav tööriistakomplekt (*toolkit*) ja infrastruktuur.
- TeraGridi Kasutajate Portaal (*TeraGrid User Portal* [30]) on portaal, mille kaudu TeraGridi kasutajad saavad luua ja kasutada oma kasutajakontosid lihtsalt, ilma liigsete komplikatsioonideta. TeraGrid on avatud teaduslik infrastruktuur, milles osaleb 9 partnerit ja mille tulemusel on tekkinud maailma suurim jagatud küberinfrastruktuur teaduslikuks uurimustööks. Süsteem hõlmab rohkem kui 102 teraflopsi arvutusvõimsust ja rohkem kui 15 petabaiti andmemahtu.
- CIMA Kristallograafia Portaal (*CIMA Crystallography Portal* [31]) hõlmab mitmeid projekte, mille eesmärgiks on tõhustada ligipääsu teaduslikele instrumentidele ja sensoritele.
- TIGRE teadusportali (*TIGRE Science Portal* [32,33]) eesmärgiks on ehitada arvutuslik GRID, mis integreerib arvutussüsteeme, andmesüsteeme ja –baase, visualisatsiooni laboreid, instrumente ja sensoreid üle Texase. TIGRE

kasutajaportaal on keskseks kohaks, kust kaudu saab informatsiooni ja ligipääsu TIGRE ressurssidele.

- Maaväriinate Simulatsiooni Portaal (*QuakeSim Portal* [34]) on portaal, mille eesmärk on luua raamistik, et paremini mõista ja ennustada tektoonilisi ja maaväriinate protsesse, ning luua toimiv süsteem nende protsesside tundmaõppimiseks. Samuti tagatakse ligipääs maaväriinate andmebaasi.

Kokkuvõte

Käesoleva töö eesmärgiks oli anda ülevaade OGCE portletitest ja veebiteenustest ning populariseerida OGCE ja üldse GRID-i kasutamist. Töö käigus töötati läbi sadu lehekülgi võõrkeelset materjali, mille põhjal kirjeldati OGCE koostisosasid ning anti ülevaade OGCE üldisest struktuurist. Erilist tähelepanu pöörati ingliskeelsete mõistete paralleelsele välja toomisele, mis lubab töö lugejatel teksti siduda erialase ingliskeelse materjaliga. Samal põhjusel esitati ka mitmeid viiteid allikamaterjalidele, mis aitavad asjast huvitatutel lisainformatsiooni hankida.

Töö käigus saadi ülevaade OGCE-l põhinevate portaalide arhitektuurist ning võimalustest. Selgus, et OGCE portlete ja teenuseid võib kasutada suvalisel JSR-168 standardiga ühilduval portleti konteineril. Samuti jõuti järeldusele, et portleti API tagab laiendatava arhitektuuri, mille peale korralike veebiportaale ehitada. Lisaks uuriti OGCE-ga seotud teenuseid ning anti kiire ülevaade OGCE-d kasutavatest teadusasutustest.

Töö tulemustest järeldub, et GRID tehnoloogial põhinevad OGCE portaalid on vajalikud ning ka Eesti avalikkusele võiks teha ettepaneku luua OGCE tehnoloogial põhinev teaduslik portaal. Loodav portaali koondaks teadusasutuste arvutiressursse ning võimaldaks teadlastel seda enda huvides kasutada ning üksteisega suhelda.

OGCE

(Open Grid Computing Environments)

Bachelor thesis

Raido Türk

ABSTRACT

The purpose of this given bachelor work is to give an overview of OGCE (Open Grid Computing Environments) and popularise the usage of Grid based portals and Grid technology itself in Estonia.

The topics covered in current work are as follows: OGCE main architecture with detailed description of JSR-168 standard portlets, with brief overview, what are the requirements to install OGCE. Also the main concepts of portlets API are explained. In addition to those, OGCE related Maven plugins are discussed and OGCE tools are. Also an overview of related services, that are used in OGCE system, is given. The final chapter of the work gives overview of most famous OGCE based portals.

The graduation paper was created based on information found in internet. Therefore a great deal of different web pages containing hundreds of pages of information has been worked through and all the sources of information used in current work has been listed accordingly.

Kasutatud kirjandus

1. Open Grid Computing Environments – OGCE. 2006 [WWW]
<http://www.collab-ogce.org/ogce2/> (28.05.2006)
2. OGCE Release 2 – OGCE . 2006 [WWW]
<http://www.collab-ogce.org/nmi/ogce2-docs/index.html> (28.05.2006)
3. Grid Computing – Wikipedia, the free encyclopedia. 2006 [WWW]
http://en.wikipedia.org/wiki/Grid_computing (28.05.2006)
4. Sun Microsystems – JSRs: Java Specification Requests. 2006 [WWW]
<http://www.jcp.org/en/jsr/detail?id=168> (28.05.2006)
5. Sun Microsystems – Introduction to JSR 168 – The Portlet Specification . 2003 [WWW]
<http://www.servogrid.org/slide/GEM/NMI/OGCE2TutorialMaterial/JSR168Tutorial.ppt> (28.05.2006)
6. Architecture Overview – Apache Software Foundation. 2006 [WWW]
<http://portals.apache.org/pluto/v101/developer/arch.html> (28.05.2006)
7. Alejandro Abdelnur and Stefan Hepper – Java™ Portlet Specification Version 1.0. 2006 [PDF]
http://sdhc-esd.sun.com/ESD6/JSCDL/portlet/1.0-fr/portlet-1_0-fr-spec.pdf?AuthParam=1148109702_db27034bbb52bd5e333d2989d676b147&TUrl=an1npDpbKod7kSYrROhENTonIeQ1W0D1Lc4nXz+pGFFranixdCdgxDTPbW4=&TicketId=dld/NwJKNu0+/w==&GroupName=SDLC&BHost=sdhc5i.sun.com&FilePath=/ESD6/JSCDL/portlet/1.0-fr/portlet-1_0-fr-spec.pdf&File=portlet-1_0-fr-spec.pdf (28.05.2006)
8. Stefan Hepper and Stephan Hesmer – Introduction to Portlet Specification, Part 1. August 1 , 2003 [WWW]
<http://www.javaworld.com/javaworld/jw-08-2003/jw-0801-portlet.html>
(28.05.2006)

9. Marlon Pierce – Developing Grid Portals Using Portlets. April 3, 2003 [PPT]
<http://www.servogrid.org/slide/GEM/NMI/OGCE2TutorialMaterial/PortalOverview.ppt> (28.05.2006)
10. uPortal - The JA-SIG Collaborative. 2006 [WWW]
<http://www.ja-sig.org/wiki/display/UPC/Home> (28.05.2006)
11. GridSphere – GridSphere. 2006 [WWW]
<http://www.gridsphere.org/gridsphere/gridsphere> (28.05.2006)
12. Apache Tomcat – Wikipedia, the encyclopaedia. 2006 [WWW]
http://en.wikipedia.org/wiki/Apache_Tomcat (28.05.2006)
13. Apache Tomcat – The Apache Software Foundation. 2006 [WWW]
<http://tomcat.apache.org/> (28.05.2006)
14. NSF Releases NMI-R9 – Grid today. May 8, 2006 [WWW]
<http://www.gridtoday.com/grid/649667.html> (28.05.2006)
15. The GridPort Toolkit – GridPort. 2006 [WWW]
<http://gridport.net/main/> (28.05.2006)
16. Gregor von Laszewski and Kaizar Amin – Java CoG Kit Abstraction Guide. January 17, 2006 [WWW]
http://wiki.cogkit.org/index.php/Java_CoG_Kit_Abstraction_Guide (28.05.2006)
17. Portlet plugin – Marcus Christie. 2005 [WWW]
<http://www.cs.indiana.edu/~machrist/portlet-plugin/> (28.05.2006)
18. PURSe Portlets – OGCE [WWW]
<http://www.extreme.indiana.edu/portals/purse-portlets/index.html> (28.05.2006)
19. GPIR – GridPort. 2006 [WWW]
<http://gridport.net/services/gpir/> (28.05.2006)
20. Comprehensive File Transfer Service – GridPort. 2006 [WWW]
<http://www.gridport.net/services/cft/> (28.05.2006)
21. What is the SRB– San Diego Supercomputer Center. 2006 [WWW]
<http://www.sdsc.edu/srb/index.php/SRB>About> (28.05.2006)
22. Storage Resource Broker – CCLRC. 2004 [WWW]
http://www.e-science.clrc.ac.uk/web/projects/storage_resource_broker
(28.05.2006)

23. SRB Portlet Development for the Grid Portals – Mary Thomas. February 3, 2006
[WWW]
http://www.sdsc.edu/srb/Workshop/Talks/Talks4/SDSU_SRB_Portlets.ppt
(28.05.2006)
24. What is Condor ? – Condor Team [WWW]
<http://www.cs.wisc.edu/condor/description.html> (28.05.2006)
25. Tupelo Wiki – May 1, 2006 [WWW]
http://dlt.ncsa.uiuc.edu/wiki/index.php/Main_Page (28.05.2006)
26. OWL Web Ontology Language Overview – W3C. February 10, 2004 [WWW]
<http://www.w3.org/TR/owl-features/> (28.05.2006)
27. Tupelo: a semantic content repository – Joe Futrelle [PDF]
<http://dlt.ncsa.uiuc.edu/presentations/ggf15-futrelle.pdf> (28.05.2006)
28. Linked Environments for Atmospheric Discovery - LEAD portal [WWW]
<https://lead.extreme.indiana.edu:8443/gridsphere> (28.05.2006)
29. Project Information. 2006 [WWW]
<http://www.ncbiogrid.org/> (28.05.2006)
30. Documentation – NSF. 2006 [WWW]
<https://portal.teragrid.org/> (28.05.2006)
31. CIMA Crystallography Portal – CIMA. 2006 [WWW]
<http://iri164.informatics.indiana.edu:8080/gridsphere> (28.05.2006)
32. TIGRE. 2006 [WWW]
<http://www.hipcat.net/Projects/tigre> (28.05.2006)
33. TIGRE portal. 2006 [WWW]
<http://tigreportal.hipcat.net/> (28.05.2006)
34. QuakeSim. 2006 [WWW]
<http://complexity.ucs.indiana.edu:8282/> (28.05.2006)
35. QuakeSim – NASA. 2006. [WWW]
<http://quakesim.jpl.nasa.gov/> (28.05.2006)