Performance Assessment of Computational Grid on Weather Indices from HOAPS Data

Madhuri Bhavsar, Anupam K Singh, Shrikant Pradhan

Abstract—Long term rainfall analysis and prediction is a challenging task especially in the modern world where the impact of global warming is creating complications in environmental issues. These factors which are data intensive require high performance computational modeling for accurate prediction. This research paper describes a prototype which is designed and developed on grid environment using a number of coupled software infrastructural building blocks. This grid enabled system provides the demanding computational power, efficiency, resources, user-friendly interface, secured job submission and high throughput. The results obtained using sequential execution and grid enabled execution shows that computational performance has enhanced among 36% to 75%, for decade of climate parameters. Large variation in performance can be attributed to varying degree of computational resources available for job execution.

Grid Computing enables the dynamic runtime selection, sharing and aggregation of distributed and autonomous resources which plays an important role not only in business, but also in scientific implications and social surroundings. This research paper attempts to explore the grid enabled computing capabilities on weather indices from HOAPS data for climate impact modeling and change detection.

Keywords—Climate model, Computational Grid, Grid Application, Heterogeneous Grid

I. INTRODUCTION

COMPUTATIONAL grids are large scale distributed computing systems which involve the coordinated sharing of heterogeneous resources across administrative domains[6]. Grid computing is not only interesting to business world, but offers less expensive computing power for complex computational models of various engineering and natural science disciplines. The computational power is composed by harnessing the unused CPU cycles of desktop computers available in the network on the campus. The grid computing model accelerated the performance of computation of weather indices from global climate data acquired from

Madhuri Bhavsar and Shrikant Pradhan are with the Computer Engineering Department, Nirma University of Science and Technology, Ahemadabad, India (phone number +91-02717-241911-17 Ext 262, Email mdb.p_itcomp.it@nirmauni.ac.in & snp.it@nirmauni.ac.in   )

Anupam K Singh is with Civil Engineering Department, Nirma University of Science & Technology, Ahemadabad, India. (Email - anupam.singh@gmx.net)

Hamburg Ocean-Atmosphere Parameters and Fluxes from Satellite Data (HOAPS). Aggregation of the existing resources provides a cohesive computational platform.

II. COMPOSITION OF GRID

The grid based computational model has two objectives which utilize large amount of computing resources harnessed on the campus and reduce execution time of raster based climate model. Because desktop resources are pervasive, desktop grid systems can aggregate large number of machines even within a single organization. The policy involves exploiting the availability of the idle desktop workstations or PCs often idle as 85% during the day. [3] [4]. To compose a grid, the machines which we have coordinated together are heterogeneous in architecture and OS. Hence in order to achieve a high degree of utility, desktop grid systems must be able to put an application on the platform easily [2]. Further to make the data secure, Grid proxy is developed, which is discussed further in the same section.

The climate model is executed on the grid which is developed using globus toolkit and third party schedulers like OpenPBS and Condor. Globus Toolkit 4 is an open source toolkit organized as a collection of loosely coupled components [5]. These components consist of services, programming libraries and development tools designed for building grid-based applications.

Major functional components of GT4 contributing areas of GT4 are:

1. Security- GT4 security components collectively referred to as GSI facilitate secure communication through the policies defined for authentication and authorization, delegation and credential management.
2. Delegation and simple CA toolkit- It includes a service that delegates credentials to a container.
3. Data Management- GridFTP, Reliable File Transfer, Data Replication Services, OGSA-Data Access and Integration are the components which provide data for discovery, transfer and access to large data.
4. Execution Management- These components handle the deployment, scheduling and monitoring of jobs. Grid Resource Allocation and Management provide services to deploy and monitor jobs on the grid.

5. Monitoring and Discovery System- It is a part of information services that monitors and discovers resources in a Virtual Organization (VO).
6. Java and GT4 container which handles client requests includes standalone containers.[10]

In addition to the above-mentioned components, proxy generated for creation and submission of jobs as shown in figure 1, provides authenticated and secured access to the grid. This certificate is generated in X.509 format. An X.509 certificate is a file which includes a lot of information including *distinguished names.*

User is facilitated for acquiring flexibility in computation by either Grid-proxy-init or Globusrun.

If the user opts for Grid-proxy-init and wants single-sign-on facility, then the user will be prompted for lifetime of the proxy certificate, the strength of the key and password to access the private key of the user. The more the strength, the harder would be the effort to guess the key. For accessing the private key, pass phrase is required.
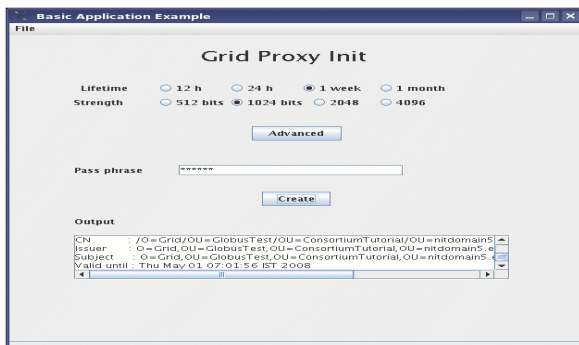

Fig. 1 Grid proxy generated for creating the proxy of head node

III. DESCRIPTION OF COMPUTATIONAL CLIMATE MODEL

This topic describes analysis, design, functional description and results obtained after the deployment of the climate model which facilitates the computations of various parameters for rainfall such as mean (μ), standard deviation (σ), anomalies for the years 1987 to 2005, which in turn requires parameters for climate change signals and rainfall prediction.

Data Analysis

To derive a climatology from satellite data major demands on the used instruments are that they must deliver accurate and stable measurements over a long time which is only achievable with an on board algorithms that consider the aging of a certain radiometer[7]. To analyze the huge amount of acquired data which was in the netCDF format, is transferred to an ASCII format and supplied to the grid for further computations which are requisite parameters for rainfall prediction.

Data Source

Hamburg Ocean Atmosphere Parameters and Fluxes from Satellite Data - HOAPS-3 - daily twice composite
At $01^0 \times 01^0$ grid cells

Data Source Description

The Hamburg Ocean Atmosphere Parameters and Fluxes from Satellite Data (HOAPS) set is a completely satellite based climatology of precipitation, turbulent heat fluxes and freshwater budget (evaporation minus precipitation) as well as related atmospheric state variables over the global ice free oceans[8]. It has been developed by Meteorological Institute, University of Hamburg and Max Planck Institute for Meteorology, Hamburg.

Dataset Description

This dataset contains 1 degree, daily twice, globally gridded multi-satellite rain values falling in time series of years from 1987 to 2005, providing high temporal resolution. The fields are stored for 0-12 and 12-24 UTC. Time steps in the data files are at 0 UTC (0-12 UTC overpasses) and 12 UTC (12-24 UTC overpasses). Each grid-cell contains the average of data from the satellite that passed this gridbox closest to 12 and 24 UTC, respectively. The dataset is in netCDF format.

Processing on Dataset

The dataset was processed to extract rain values for a predefined window out of the whole global grid.
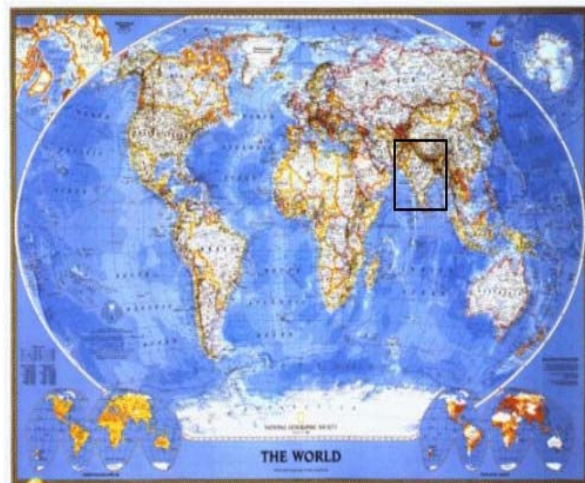

Fig. 2 Grid proxy generated for creating the proxy of head node

This window corresponds to the area covering India in the global grid shown in figure 2 with a rectangle, was defined by following coordinates:

5°N 40°E
35°N 95°E

The extraction of data was done as follows:

- The netCDF dataset contained 228 files, each corresponding to a month from January, 1987 to December, 2005.
- Each file was converted into readable ASCII format using a software code for format conversion.
- Data in the ASCII files was in following structure:
  1. Header providing information about data
  2. Rain values separated by ',' with time steps mentioned above.
- All the files of totalling size of 2.7 GB were read word by word for data extraction.
- The rain values whose coordinates accommodated in the window were extracted on 12 hourly bases i.e., for every period of 12 hours from January 1, 1987 to December 31, 2005; one window was extracted and stored in a two dimensional tabular format in a central database system.
- The format of each table was "rain_<year>_<month>_<day_of_month>_<binary_value _representing_half_of_the_day>". More than 13780 such tables were created each having 1736 records.
- Each table had three scaled fields corresponding to latitude, longitude and rain value respectively.
-

This way, there were large data sets generated. These data sets are to be processed and statistical analysis to be performed on the computational grid platform designed for this purpose.

IV. DESIGN

To illustrate in more detail the way climate model works, a functional diagram and class diagrams are shown in figure 3 and 4 respectively. The grid architecture which refers pool of resources is modeled with Globus, PBS, Sun SGE and Condor and thus forms heterogeneous environment. Figure 5 shows functional prototype of climate model on this heterogeneous platform.
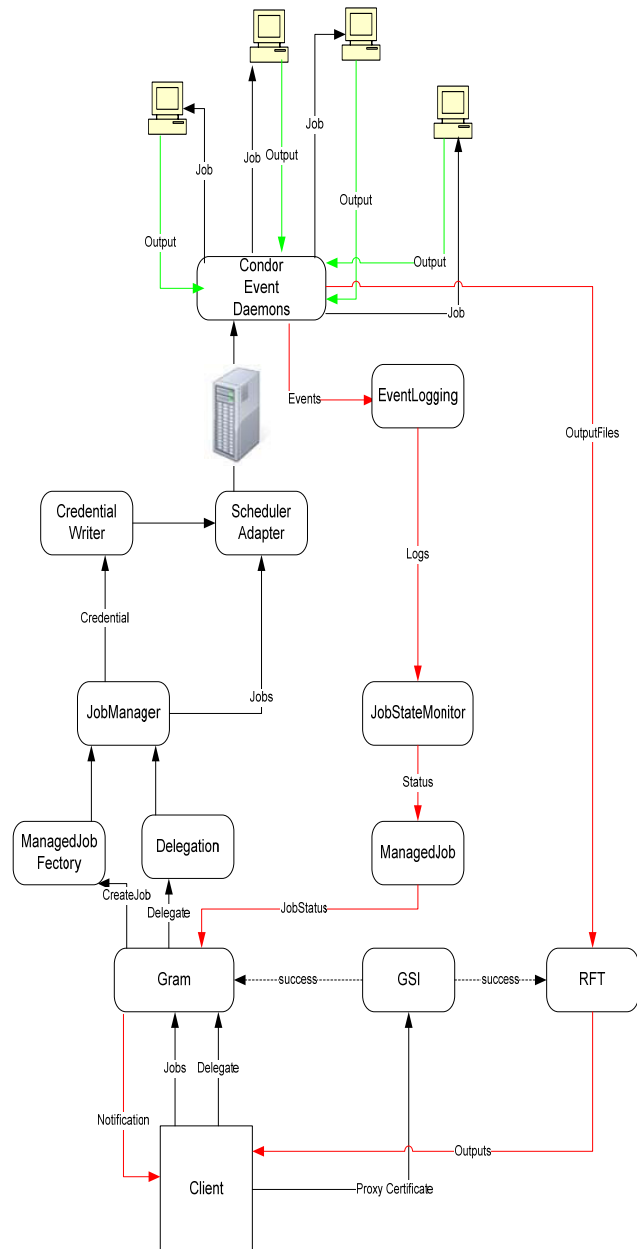

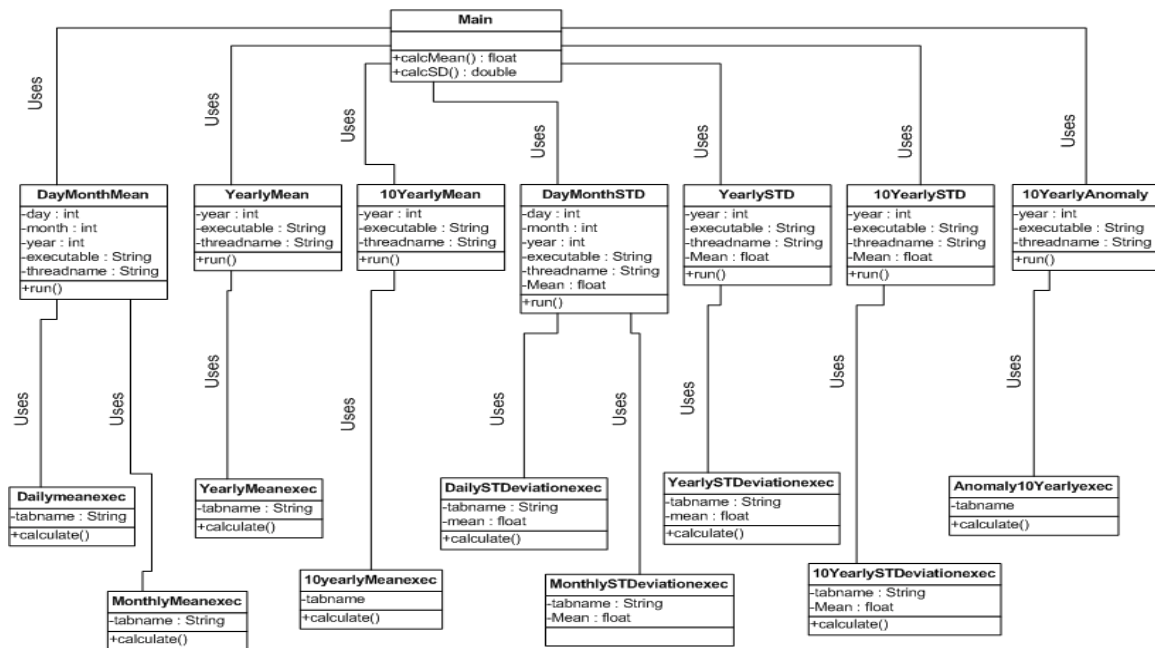Fig. 3 Data flow diagram of climate model deployed on grid
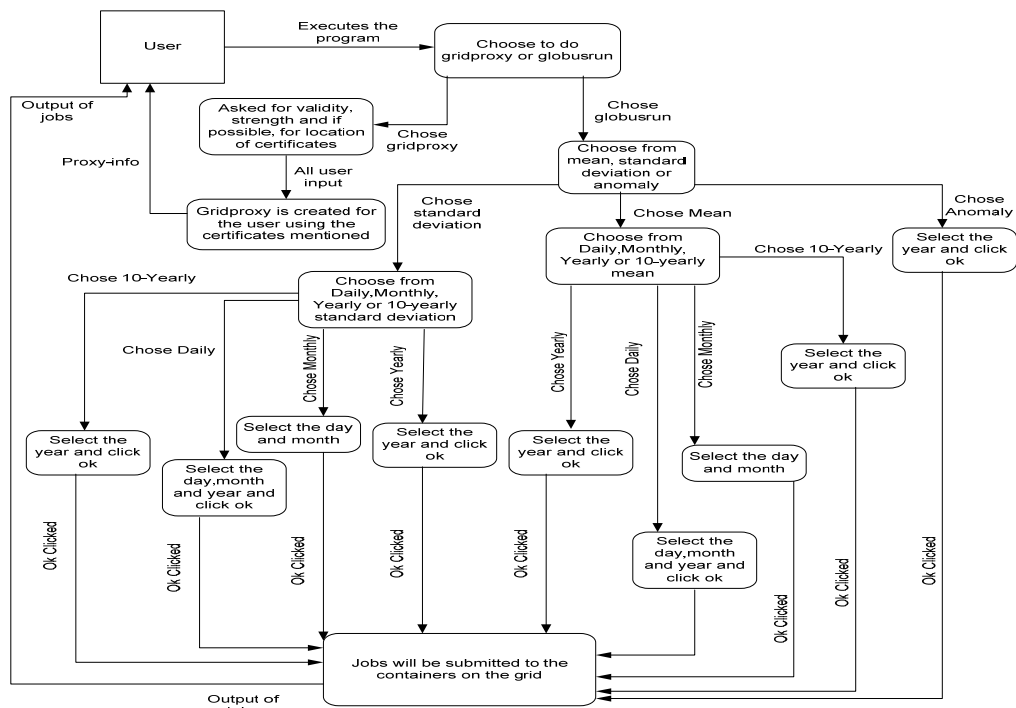
Fig. 4 Class diagram of design model

Fig. 5 Functional Prototype of climate model

Design diagram shows the flow of the data in the grid infrastructure after the jobs are submitted to the container. The Class diagram shows the taxonomy of classes used in the application.

Major classes are:

DayMonthMean: Used to generate RSL file for PBS and Condor and computes mean for a day or a month.

Main: Abstract class used to call multiple threads of different classes.

Anomaly10yearlyexec: Finds out anomaly for the period of 10 years.

V. IMPLEMENTATION AND RESULTS

Upon completion of grid programming using the globus programming components [1][9] and deploying climate model, many valuable results are obtained.

The following Screenshot represents options provided to the user for selection of parameter and the frame for finding out the Standard Deviation and anomaly of the rain for a given day/month or year respectively. This also facilitates in turn, the computation for the required number of years. Different computed parameters based on year wise HOAPS data has shown in figure 11 and hence exhibit that grid enabled execution offers an enhancement in execution.

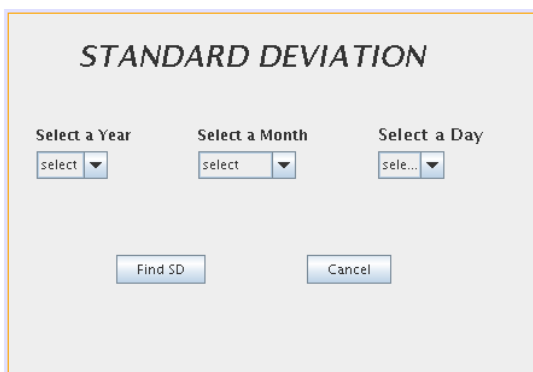

Fig. 6 Parameter selections for computation

Fig. 7 Screen provides computation facility on the grid

Fig. 8 Frame for finding out the anomaly for the given year

VI. MODEL PERFORMANCE ANALYSIS

Grid exploits an idle computational cycles of the available resources. The application when developed and installed on the grid, the results obtained are shown in table 1, table 2 and table 3. Average time for finding mean on the grid constituted with 10 machines is 92 seconds whereas sequential execution consumes 255 seconds. Standard deviation computation on grid acquires 175 seconds whereas sequential execution for the same could consume 415 seconds. Computation of these parameters as shown below in the expressions, leads to the calculation of anomalies for the required time period (1). Result shows the gain in execution time achieved on the grid platform. Graph shows results obtained for the computation of anomalies with respect to time. These results are submitted further to civil engineering department for further predictions needed for meteorological study.

$$\mu = \sum_{i=i}^{N} x_i \times \frac{1}{N}$$

$$\sigma = \sqrt{\sum_{i=1}^{N} (x_i - \mu)^2 \times \frac{1}{N}}$$

$$\{x \mid x = (x_i - \mu)/\sigma\} \tag{1}$$

TABLE 1 RESULTS OF GRID ENABLED COMPUTATION

GRID ENABLED EXECUTION OF THE JOB FOR FINDING MEAN AND STANDARD DEVIATION FOR 10 YEARS' DATA OF RAIN				
YEARS (FROM – TO)	MEAN [MM]	TIME (MIN:SEC)	STANDARD DEVIATION	TIME (MIN:SEC)
1987-1996	69.64	1:28	48.378	2:41
1990-1999	74.81	1:34	49.272	2:55
1995-2004	83.42	1:55	52.603	3:43
1993-2002	53.66	1:15	42.134	2:30
1989-1998	54.81	1:28	42.245	2:48

TABLE II RESULTS OF ANOMALIES OBTAINED FROM MEAN AND STANDARD DEVIATION FOR THE DEMANDED PERIOD

Grid enabled execution of the job for finding out the anomaly for 10 years' data of rain	
Year	Time (min:sec)
1987-1996	16:38(worst case)
1993-2002	15:28

TABLE III SEQUENTIAL EXECUTION OF THE JOB FOR FINDING OUT THE MEAN AND STANDARD DEVIATION FOR 10 YEARS' DATA OF RAIN

Years (From-to)	Mean	Time (min:sec)	Standard Deviation	Time (min:sec)
1987-1996	69.67	2:18	48.377	3:30
1990-1999	74.86	5:00	49.271	8:43
1995-2004	83.34	3:58	52.603	5:31
1993-2002	53.62	5:10	42.132	8:39

TABLE IV COMPARATIVE ANALYSIS OF EXECUTION

Years (From-to)	Sequential execution (min:sec)	Grid enabled execution (min:sec)	% Improvement
1987-1996	2.18	1.28	36.5
1990-1999	5.00	1.34	68.8
1995-2004	3.58	1.55	67.25
1993-2002	5.10	1.15	75.7
1989-1998	2.20	1.28	37.5

Fig. 11 Performance assessments and improvement of grid enabled execution over sequential execution
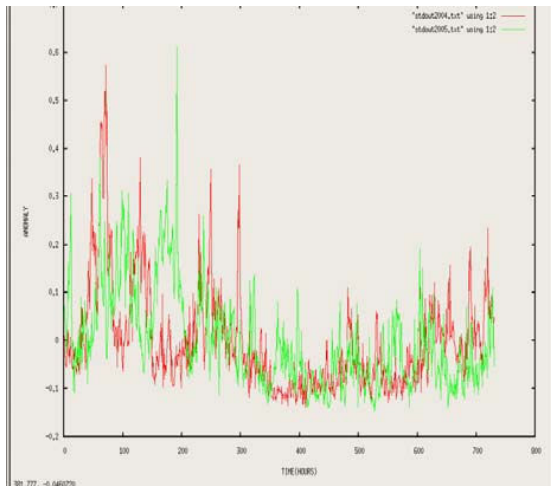
Fig. 9 Comparison of anomaly values for 2004 and 2005
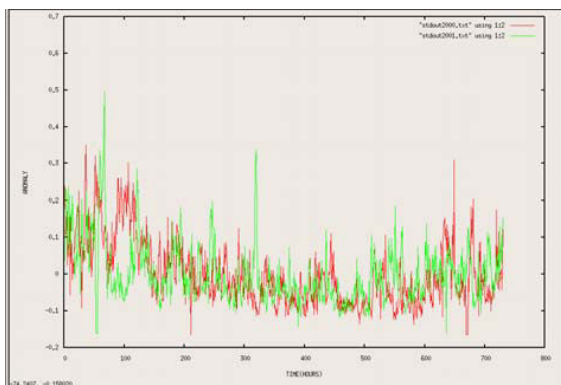
Fig. 10 Comparison of anomalies for 2000 and 2001

VII. CONCLUSION

The heterogeneous grid setup on the campus with various software mechanisms such as middleware, schedulers and job submission components like Globus, PBS, SGE and Condor is exigent. The large computing power generated by exploiting an idle computers, helped in achieving gain in the execution time required for the computation and classification of weather indices of climate model.

ACKNOWLEDGMENT

REFERENCES

[1] Bart Jacob, Luis Ferreira, "Enabling Applications for Grid Computing with Globus", Redbooks IBM, June 2008.
[2] I. Foster and C Kesselman, "The Grid2: Blueprint for a New Computing Infrastructure." Morgan Kaufmann publisher, 2004.
[3] Litzkow,M., and Livny, M. Experience with the Condor distributed batch system, IEEE Workshop on Experimental Distributed Systems, IEEE Computer Society Press, Los Alamitos, 1990.
[4] Litzkow,M., and Livny, M and Mukta, M. W., Condor, - A hunter of idle workstations, in 8th International Conference on Distributed ComputingSystems, 1988.
[5] Luis Ferreira, Viktor Berstis, "Introduction to Grid Computing with Globus", Redbooks IBM, October 2003.

[6]   Daniel Minoli,"A Networking Approach to Grid Computing",Wiley-Interscience, 2004.
[7]   H. Grassl, V.Jost, J.Schulz, M.R. Ramesh Kumar, P. Bauer, and P. Schhluessel, " The Hamburg Ocean-Atmostphere Parameters and Fluxex from Satellite Data: A CLimatological Atlas of Satellite derived AirSea Interatcion Parameters over the World Oceans", Max-Planck Institute,hamburg, Germany, November 2000.
[8]   http:// http://cera-www.dkrz.de/WDCC/ui/
[9]   http://www.globus.org/toolkit/docs/4.0/Exution/wsgram
[10] Sotomayor Borja, Childers Lisa,"Globus Toolkit4, Programming java Services", Morgan Kaufmann publisher, 2006