βtest- GridCom-TC v1.0

Grid Computing Training Courseware

Grid Computing Training Courseware *v*-1.0

Designed for Testing, Benchmarking & Performance Activities

Document Title	**Grid Computing Training Courseware (βtest-GridCom-TC-*v*1.0)**
Users	Betatesting Group Members of NPSF, C-DAC,Pune
Source	Parallel Computing /Grid Computing conducted by Betatesting group, C-DAC, Pune & Author's Experience on Processing and Grid Computing Projects
Date:	August 01, 2006
No. of Pages	41
Author	VCV.Rao
Contact	betatest@cdac.in

β-Testing Group, C-DAC

Betatesting Group, National PARAM Supercomputing Facility,
Centre for Development of Advanced Computing (C-DAC)

Non-Disclosure

This material is developed and provided by the Centre for Development of Advanced Computing (C-DAC), Pune, India. Licensor grants to Licensee, an Indian Government agency or a non-profit educational institution ("Licensee"), a royalty free, non-transferable license to use the copy of the product as per terms and conditions of this License Agreement. Licensee is permitted to use the product solely for Licensee's own internal research purposes, and alter, modify, adapt the product for Licensee's own internal research purposes.

1. Objective

The aim is to design Grid Computing courseware and we focus on many of the issues that an architect or developer or tester needs to be aware of when designing a grid-enabled application as well as performance of a grid-enabled application.

The training courseware objective is to build expertise for execution of *Test plan and Benchmarks for grid infrastructure* with focus on the open source Globus Toolkit 2.x/4.x. The foundation is focused on quickly adapt to developments in enabling applications to grid environment.

The courseware is useful for beginners, middle level and advanced level grid users. The course has been classified into two-tier level, focusing on various aspects of Grid infrastructure and each tier module gives an overview of topics, which benefit the on-going project activities.

The complete courseware forms a single concentrated course on grid computing, which is a continuously evolving resource at C-DAC. The courseware can be easily tailored to the developer, testing group, user community to extract performance of large scale applications and gives a strong foundation on programming models for Benchmarking computing systems of grid infrastructure.

The philosophy is to introduce new functionality and concepts to solve a design, implementation or analysis of enabling applications for Grid Computing with Globus in this courseware. Various parts of the courseware and details of modules are summarized below.

To understand about the sorts of problems, beginners, developers encounter, when they begin thinking in write grid programming programs using different programming paradigms, the course contents may help and it is designed for short, and long-term duration of time schedule.

Suggested below are one ***short-term*** and ***long-term*** courses or preliminary/exhaustive training programe of grid programmers. Some modules may focus on theory and laboratory session as per requirements of project activities. However, some groups in C-DAC may require detailed contents of modules, pertaining to the on-going research projects.

For specific project work, it is necessary for the course participant to refer advanced books on Grid Computing or visit important web sites.

2. Various part of βtest-GridCom-TC *v*1.0 Courseware

The ***β*test**-GridCom-TC *v*1.0 courseware can be grouped into *twelve* parts and each part has sufficient number of modules. Each module contents can be covered in classroom lectures and the Hands-on Session can be done on grid laboratory. A brief summary of various parts and flow of several modules of courseware is discussed below.

Part I - Introduction to Grid Computing: Reasons for Computational Grids, Some definitions, Grid Applications Classes An overview of Using Grids – Grid Developers, Tool Developers, Grid Programming Environment, Application Developers, End Users, System Administrators, Major Grid Projects

Part II - Application Requirements & Characteristics: Category of applications - First-in-time, or an on-demand computing, Data-Intensive Computing, Collaborative Computing, High Throughput Computing, Distributed Supercomputing, A Grid Systems Taxonomy - Design Objectives of Applications - (Computational Grid, Data Grid, Service Grid); Classification of Grid Applications – Class-I(Loosely Coupled), Class-II (pipelined), Class-III (Tightly Synchronized), Class-IV (Widely Distributed); Performance Metrics; Reproducibility of results, Measurement of Turnaround time and Throughput, Quantification of Grid middleware Overheads, The ratio of communication and comp., Use of numerical libraries on computing systems of different sites with architectural diversity, Algorithms to handle latency and bandwidth requirements of application

Part III-Globus 2.4 Services: An Overview of Globus Toolkit- 2.4; Resource Management, Layered Grid Architecture- Protocols and Services and API; The Hour Glass Model of Globus Architecture, Configure Globus Toolkit; An overview of GRAM, Installation and Set-up of Globus 2.4, Data Management GRIDFTP, Monitoring and Discovery (MDS), Grid Security Principles (GSI)

Part IV-Globus 4.0 Services: GT4 Architecture Overview, Layered Grid Architecture- Protocols and Services and API Installation and set-up of Globus 4.0, GT4- Distributed systems and Web Services, OGSI Implementation, Open grid Service Architecture (OGSA), GT4 Service Oriented Applications and Infrastructures, GT4 Web Services Implementation, GT4 Web Service Specifications, GT4 Service Oriented Architecture; Grid Security Principles (GSI)

Part V-Application Architecture Considerations: The Characteristics of grid application, Understand Globus 2.4 Components- GSI, GRAM, MDS, GridFTP, GSI, GASS, Grid Programming language considerations; Characteristics of application flow and job flow; Job Dependencies on System environment; Job topology, Job Criteria (Standard application, Parallel Applications); Passing of data input/output; Qualification scheme for grid applications, Knock-out criteria for gird applications

Part VI-Grid Data Management Considerations: Grid Programming Model – Shared data and shared nothing, Data Characteristics, Data Management techniques and solutions -Shared file system, Replication, Caching effects, Data Access Control Systems, Database solutions for grids, Distributed Data approaches, Replication, Global Parallel File System (GPFS) Data Management techniques

Part VII-Grid Programming Environment -I: The Programming Problem; Compilers languages and Libraries; Grid Global Compilation Systems, Grid Programming Model – Shared data and shared nothing, Object Oriented-based applications; Examples of Grid Programming Model, Grid Web Services and Peer to Peer (P2P) Grid Concepts, scripting languages (Javascripts, Perl, Python, Tcl/Tk, Unix Shells)

Part VIII- Grid Programming Environment -II: Middleware & Tools: Network Enabled servers, Frameworks and Component based technology, Grid enabled RPC Systems, Middleware Systems – Legion, and Object Oriented Technology, XML technology, Services & Protocols (Web Services/XML), Scripting language, Problem Solving Environments, Portals, Application Specific tools

Part IX- Enabling Applications for Grid Computing: Write C/C++, Perl COG programs using Globus APIs *globusjobrun* command and associated APIs, and Resource Specification Language (RSL) scripts for submitting jobs, Write JAVA CoG programs using Globus APIs *globusjobrun* commands and associated APIs, Example programs using MPICH-G2 & C/C++ language; programs on Perl CoG, Python CoG using Globus APIs; Distributed Applications using Common Component Architecture technologies (XML, Web-services) and Grid Services

Part X- Grid Integration Test Scripts: Automatic Test scripts for Grid with Globus Toolkit – Interoperability Test Suites – RSL-Hello World, Stage, RSL-Shell, BatchJob Submitt, BatchJob Query, GASS, GridFTP, GSIssh, GSIscp tests; Grid MDS Tests – GRIS, Job managers, Support of *globusrun* job commands

Part XI- Low Level Grid Benchmarks (Grid Probes): Estimate Basic authentication for all grid nodes, query MDS, Ping, Ping-Pong, Sleep, Data Transfer, Circular, and Gather Probes, Compute and Communication Benchmarks

Part XII- Application Grid Benchmarks: NAS Grid Benchmarks; GridBench (CrossGRID) - Micro, Macro and Application Benchmarks and Performance Evaluation on Grid (Resources, Site Configuration), Benchmark Metrics, and Grid Bench Definition Language.

Most importantly, the courseware has sufficient number of programming assignments, which should play a central role to make strong foundations on grid programming in order to solve particular applications. Several modules defined in the courseware can be grouped together as per requirements of members who opt for short, and long term time duration.

Figure 1 illustrates the various parts of the courseware and Figure 2 illustrates the various modules of each part and its relations with other modules. The solid arrow from Module *A* to Module *B* indicates Module *B* depends heavily upon material presented in Module *A*.

For long-term courseware schedule, Module *A* and Module *B* have mutual relations, while performance of application programs with programming environment are considered on target architecture.

All course contents given in each part is covered in numerical order, you will satisfy all requirements for Testing, Benchmarks execution, and enabling applications for grid computing environment with Globus. However, you would like your members to start programming in *C* or *C++*, or *Perl* or *Phyton*, or *Java* or with Globus as quickly as possible, you may wish to skip some modules in Part IX or cover only two or three modules of the courseware. Definitely there is a weak dependency across several modules of different parts and judicious choice should be taken up, which merely depends upon the duration of courseware i.e. Short, or Long term.

If you wish to focus on expertise of Globus computing infrastructure, you must work on Part-I, III, IV and you may skip Part IX.

If you would like to start by having your members programming examples with focus on Benchmarks and performance, you can jump to Part IX, XI, XII) after covering important modules in remaining parts.

To get exposure to new functionality ‘just in time’ and enabling applications to Grid Computing with Globus ToolKit, one can jump to various modules in Part I, III, IV, VII, VIII & IX with strong foundation on Part III & IX. A pre-requisite for this is to go through all modules as defined in short term course.

Most importantly, if you wish to get expertise on Testing of Grid Infrastructure with Globus Toolkit, you can work on Part XI immediately after completion of necessary modules in other parts. Special emphasis on solving complex real-life applications on Grid infrastructure and enabling applications for Grid Computing with Globus Toolkit is discussed, with focus on application requirements & Characteristics as described in Part-II & Part V. The topics in the modules of Part-II may assist the grid programmer to identify application requirements on grid infrastructure.

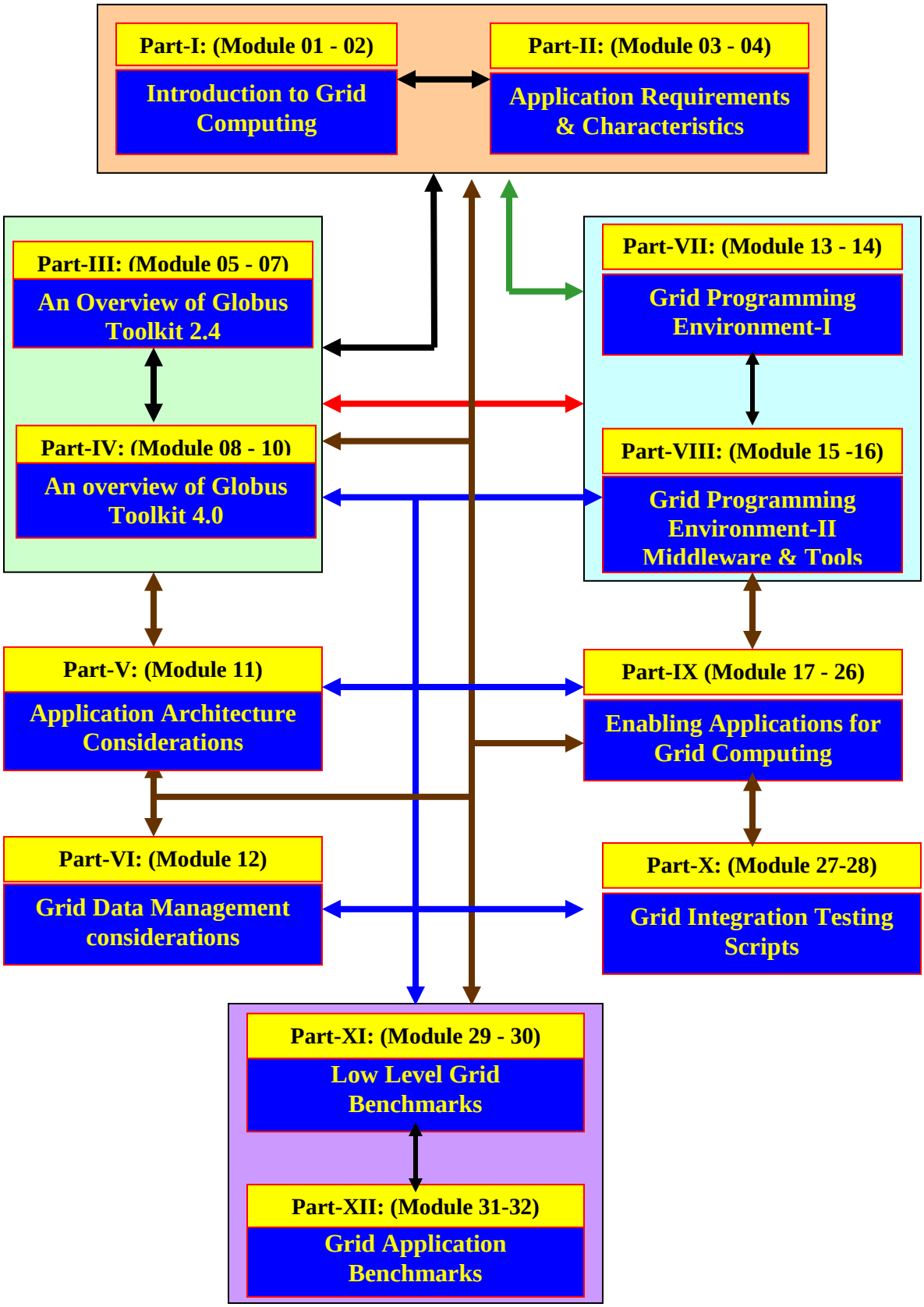

Fig 1. Flow representation of Grid Computing courseware Parts/Modules

3. Time Duration

Suggested below is time duration for courseware program and weightage may vary as per the requirements of project. The Hands-on on some modules is compulsory and it can be conducted on grid laboratory.

- Tier - 1: Short term course and the time duration is 60 Calendar days
- Tier - 2: Long term course and the time duration is 90 Calendar days.

The **short-term** course is focused on identifying suitable modules from Part-I to Part-X and quickly learns to write grid programs with focus on different programming languages using Globus Toolkit. The programming samples developed in this courseware provide the basic techniques required to get you started in the application development for grid environments. The applicability of grid environment to the particular application is important.

The **long-term** course provides complete one semester course in grid programming, focusing on turnkey projects. The topics such as grid infrastructure considerations, application specific considerations, and data management considerations, choice of appropriate grid programming environment and tools are considered. Enabling applications for grid computing with Globus Toolkit require in depth study of these topics and this can be done in an incremental fashion.