Prerequisites before Cloud Computing
Cluster/Grid Computing

Lenuța Alboaie
adria@info.uaic.ro
Summary

Paradigms

- Cluster Computing
- Grid Computing
  - Definition
  - Architecture
  - Initiatives and Applications
  - Classification
  - Evolution
  - Present & Future -> Cloud Computing
What means *computing*?

• Computing

• The way one thinks

In computer science?

• “*we can define computing to mean any goal-oriented activity requiring, benefiting from, or creating computers.*”
... Computing?

“... computing may someday be organized as a public utility just as the telephone system is a **public utility**... The computer utility could become the basis of a new and important industry.” – John McCarthy (a professor of MIT) 1961.

“As of now, computer networks are still in their infancy, but as they grow up and become sophisticated, we will probably see the spread of **computer utilities** which, like present electric and telephone utilities, will service individual homes and offices across the country.” – L. Kleinrock (one of the chief scientists of the original ARPANET project) 1969” – John McCarthy (a professor of MIT) 1961.
"it was transformed in a model consisting of consumer services (commodity computing) and can be provided in a manner similar to traditional utilities “

**Fifth utility** -> **Utility Computing** or “**Computing as a Utility**”
Computing Power?

Required:
• solving problems involving modeling, simulation and analyzes

Using unoccupied resources:
  – in the 90s almost 90% of a processor power was not used
  – the possibility to solve a wide variety of problems at affordable prices
  – cost/performance report in relation with a super-computer (HPC - high performance computer) => …. 
Trends

Traditional Food Chain

Food Chain of a Computer

Food Chain of Distributed Computing
Grid Computing

- The Grid concept appeared in the 90s
  - In analogy with *electric power grids* ~ 1910
Grid Computing

- Foster and Kesselman (1998): “A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities.”

- “The Grid is an emerging infrastructure that will fundamentally change the way we think – and use – computing. The word Grid is used by analogy with the electric power grid, which provides pervasive access to electricity and, like the computer and a small number of other advances has had a dramatic impact on human capabilities and society. Many believe that by allowing all components of our information technology infrastructure – computational capabilities, databases, sensors, and people – to be shared flexibly as true collaborative tools, the Grid will have a similar transforming effect, allowing new classes of application to emerge.” (Foster and Kesselman 2004)
Grid Computing

- Distributed computing architecture originally designed for scientific projects and then the industrial ones
- Offers the existence of a software and hardware infrastructure which allows: permanent and affordable access in a consistent manner to computing resources
- Offers various mechanisms to process data in a distributed manner
- Allows the execution of tasks on multiple machines that can be viewed as a single computer
- Offers support for searching and retrieving information, regardless of their physical location
- Offers the context to create VO - virtual organizations - which share application, data in an open and heterogeneous environment in order to solve various complex problems
  - Rules for sharing:
    - What is shared?
    - Who can share?
    - Sharing conditions
Grid Computing

- An organization can be involved in one or more VOs
  - Example: Three organizations and two VOs (P and Q)

What is shared: Computing/processing power, Data storage/networked file systems, Communications and bandwidth, Application software, Scientific instruments
Grid Computing

- Terminology:
  - *Grid middleware* – software level providing the required functionalities needed for heterogeneous resources sharing and creating a virtual organization
  - *Grid infrastructure* – refers to the combination of hardware and Grid middleware which transforms disparate and heterogeneous computing resources in a virtual infrastructure that offers to the end user the view of a single machine
  - *Utility computing* – Grid Computing and applications are providing as services (e.g. *hosting* solutions for VO, et. al.)
    - Utility computing is based on business *pay-per-use* model
Grid Computing| Architecture

- Grid Architectures use simultaneously a large number of resources (hardware, software, logical)
- Resource – a sharing entity that can be present in a Grid infrastructure:
  - Computation: PDA, PC, workstation, server, cluster
  - Storage: hard disk, RAID, SAN, ...
  - I/O type: sensors, networks, printers etc.
  - Logical: timers, ...
- Obs. Systems as: scientific instruments or HPC can be part of a Grid
- A Grid architecture focuses on interoperability issues, communication protocols between suppliers and the resource used in order to establish sharing relationships
Grid Computing | Architecture

- Generic Grid architecture

"Coordinating multiple resources": ubiquitous infrastructure services, app-specific distributed services

"Sharing single resources": negotiating access, controlling use

"Talking to things": communication (Internet protocols) & security

"Controlling things locally": Access to, & control of, resources
Grid Computing | Architecture

- Fabric
  - Provides interfaces to physical resources (computing, storage, network, ...) for which the access is mediated by Grid protocols
  - Offers components that implement local operations which are particular to each resource type

Protocols & APIs
- Include protocols & APIs providing access to shared resources
- Offer a logical vision and not a physical one over resources
Grid Computing | Architecture

- **Connectivity**
  - Core of communication protocols and authentication protocols for network transaction within Grid
  - Services for communication: transport (e.g. protocols to transfer data between resources, remote access to a resource), routing and naming
  - Authentication services: *single sign on*, delegation, integration with local security solutions, relationships based on trust

**Protocols & APIs**
- Standard Internet protocols
- Security protocols
  - Grid Security Infrastructure (GSI)
    - Authentication, authorization et.al.
Grid Computing | Architecture

- **Resource**
  - Goal: communication and security protocols (defined in Connectivity) are used for secure negotiations, monitoring, control, accounting and payment of operation per each resource
  - Resource layer is responsible for managing a single resource
  - Protocols:
    - Information protocols: used to obtain information about the structure and the state of a resource (e.g. configuration, load, use policies)
    - Management protocols: used to negotiate access to shared resources and to check the use of resources in accordance with the rules that were shared
Grid Computing | Architecture

- **Resource**
  - Protocols & APIs
    - Protocols for the initiation and control of local resource sharing
    - Management resource allocation: Grid Resource Allocation Management (GRAM)
      - Allocation, reservation, monitoring and resources remote control
    - GridFTP – efficient data access & transport
    - Information service for Grid resources:
      - Grid Resource Information Service (GRIS)
        - Access to the structure and the status of a node’s Grid
Grid Computing | Architecture

- **Collective**
  - Provides global protocols and services relating to grid resources
  - E.g. facilitates interactions between sets of resources
  - Implement various sharing services:
    - *directory*
    - co-allocation, planning and intermediation (brokering services)
    - Monitoring and diagnostic (e.g. overload)
    - Replication and discovery
    - Quantification and payment

- **Application**
  - Includes user applications operating in Grid:
    - Programming environment + high-level libraries
    - Obs. *Gridified applications* – applications designed to run in parallel and use multiple processors in Grid
Grid Computing Initiative

- UniGrid – a grid system that integrates computers in several universities and research institutes in Taiwan
Grid Computing | Initiative

- UniGrid – a grid system that integrates computers in several universities and research institutes in Taiwan
Grid Computing

- Main features offered by Grid middleware (consisting of Collective, Resource, Connectivity)
  - Virtualization and integration of heterogeneous autonomous resources
  - Providing information on resources and their availability
  - A flexible and dynamic management regarding resource allocation
  - Security (authentication and authorization) and trust
  - Management of licenses
  - Billing and payment
  - Providing QoS

- Grid computing provides advantages at two levels
  - IT management
  - Business
Grid Computing

- Advantages of IT management level:
  - Grid integrates heterogeneous resources => higher availability of computing power and efficient use of resources
  - Lowering procurement costs
  - Reducing border between departments => more scalability
  - Efficiency in computing and access to resources due to: the ability of parallel computing, load balancing => increase robustness and reliability
  - In combination with Utility Computing, Grid Computing enables the transformation of capital expenditures or IT infrastructure in operating expenses and enables an increased scalability and flexibility

- Advantages at business level
  - Lower costs and higher income
  - Easier collaboration
  - Ability to create VO with business partners
Grid Computing

- Risks and Challenges:
  - A suitable administration will avoid any 'Sever hugging' (e.g. sharing of resources that should not be shared)
  - Adjusting existing applications to function in a Grid environment
  - Lack of standards in Grid Computing leads to tough decisions on technologies used
  - Although Grid is designed to run on heterogeneous resources, involving high costs in terms of integration, it is worth considering from the perspective of keeping a standard for physical resources => full affecting the IT infrastructure
Grid Computing Initiative

- **GridPP** (UK Computing Grid for Particle Physics) - http://www.gridpp.ac.uk/
  - Contributes with over 40,000 PCs as part of the largest Grid in the world- LCG (LHG Computing Grid)
    - LHG = Large Hadron Collider (CERN, din 2007)
  - It is part of EuroGrid project

The LHC was built in collaboration with over 10,000 scientists and engineers from over 100 countries, as well as hundreds of universities and laboratories. It lies in a tunnel 27 kilometres (17 mi) in circumference, as deep as 175 metres (574 ft) beneath the Franco-Swiss border near Geneva, Switzerland.

- **Fraunhofer Grid Alliance**
  - Goal: providing a computational grid for easy access to grid resources via a Web portal
  - Based on Globus Toolkit
  - It is used in academic and industrial environment
  - www.fhrg.fhg.de
Grid Computing | Initiative

- **Jgrid**
  - Framework for Grids consisting of hardware/software components seen as services
  - It is based on Jini technology – infrastructure & programming model for creating dynamic distributed systems
  - jGrid applications can be developed via P-Grade (graphical development environment)
  - [http://jgrid.jini.org](http://jgrid.jini.org)

- **Alchemi**
  - Grid based on .NET Framework
  - Assure interoperability with other Grid systems via Gridbus Grid Service Broker
Grid Computing | Example

- Examples of applications
  - Photorealistic 3D view
    - POV-Ray rendering (Persistence of Vision Raytracer)
  - Virtual Vascular Surgery
    - CrossGrid
      - http://www.crossgrid.org
  - Solving optimization problems
    - TRACER project (use Globus, Condor, Legion, Sun Grid Engine)
      - http://neo.lcc.uma.es/
Grid Computing | Example

- Example: Earthquake Engineering Simulation

- NEESgrid: national infrastructure to couple earthquake engineers with experimental facilities, databases, computers, & each other

- On-demand access to experiments, data streams, computing, archives, collaboration

NEESgrid: Argonne, Michigan, NCSA, UIUC, USC

[http://www.nesc.ac.uk/talks/talks/Grids_and_Globus.pdf]
Grid Computing | Example

- Example: Home Computers evaluate AIDS Drugs

- **Community** =
  - 1000s of home computer users
  - Philanthropic computing vendor (Entropia)
  - Research group (Scripps)

- **Common goal** =
  - advance AIDS research

---

[http://www.nesc.ac.uk/talks/talks/Grids_and_Globus.pdf]
Grid Computing

- Classifications:
  - In relation to the type of managed resources:
    - Compute Grid – used to share computing resources (e.g. CPU) - Examples: intensive graphic processing
    - Data Grid – focused on storage, management and sharing of distributed and heterogeneous resources
    - Application Grid – focused on application management and transparently providing remote access to software and libraries; Example: grids in the bioinformatics field or earth science
    - Service Grid – resulting from Grid and SOA convergence, offers support for sharing services in an efficient manner
  - In relation to the resource sharing domain:
    - Cluster Grid
    - Enterprise Grid
    - Utility Grid Services
    - Partner/Community Grids
Classification

- Cluster Grid

Tipuri

- **Cluster Grid**
  - It is a type of parallel and distributed system and consists of a collection of autonomous computers interconnected used (and seen) as a unique resource at department / group

  *Departmental grid (Sun)/ infra grid (IBM)*
Tipuri

- **Cluster Grid**
  - Enables full use of computer resources (*mainframes*, PCs, laptops, *smartphones*, ...)
  - Cluster = set of computers– from a LAN – which form a unique computing resource
  - Obs. Clusters offer no implicit sharing of resources (improves computing capacity and storage level), and may be considered the first step towards Grid Computing
Implementation

- Beowulf (published in 2003)
  - Offer support for establishment of clusters
  - Computers can be added dynamically
  - Communications via MPI (Message Passing Interface)
  - Offer a programming model which is independent by structure, network technologies or components used
  - It contains: master nodes (coordinator) and slave/worker (processors)

[A. S. Tanenbaum, M. Steen, DISTRIBUTED SYSTEMS]
Implementation

- Typical flow for parallel executions:
Classification

- Cluster Grid
  - HPC – High Performance Computing:
    - Numerical Calculus
    - Computational Graphics 2D/3D (rendering – e.g., ray tracing, shading, ...)
    - Simulations (Biocomputing, military, ...)
    - Distributed resource search
    - Real-time critical applications
    - Distributed storage of large amount of data (*warehouses*)
    - Entertainment – e.g.: online games
Classification

- Enterprise Grid

Classification

- **Enterprise Grid**
  - Facilitates resource sharing between multiple departments within an organization (even a virtual)
    - Politics for resource management
  - It is called *intra grid* or *campus grid*
  - Example: Novartis - Pharmaceutical company
    - Held in 2003 an infrastructure consisting of thousands of desktop
    - Pilot Grid Project: 2003, Basel (Elvetia), 50 PCs “Grid enabled” connected to the existing nodes (Goal: determining the protein structure)
      - In each node there was an agent that checks system load
      - => Result: a week of running in Enterprise Grid led to results that could be obtained in 3.18 years
    - 2700 PCs (Basel, Vienna, Cambridge)
Classification

- Utility Grid

Classification

- Utility Grid
  - The Grid environment is developed and managed by a service provider
  - The usage of computing power or storage services is in *pay-per-user manner*
  - Functionality: the user does not have the Grid, he has no control over operations
    - Data and various computing operations are transmitted and then the result is expected
    - => security and *privacy problems*
    - => *reliability problems*
    - => unnecessary IT infrastructure investments
    - => Utility Computing offers scalability and flexibility on request
  - Examples:
    - Sun Grid Compute Utility from 2006
      - *Pay-per-use*: 1$/CPU per hour
      - Latter it offered support for applications
    - HP Labs offers Utility Computing for DreamWorks
Classification

- Partner/Community Grid

Classification

- **Partner/Community Grid**
  - Provides support for building VO layering on shared IT infrastructure
  - The architecture can be viewed as a collection of independent resources (e.g. Cluster Grids) that are interconnected in a global Grid middleware
  - *Partner grids* – are established between companies and universities that have a common goal
    - It defines sharing politics for resources
  - Community Grids – rely on the donation of resources (often from private individuals)
    - Example: SETI@HOME

- **Vision: Open Global Grid**
  - Represents a collection of heterogeneous Grids geographically distributed over a wide area – continent or planet
    - Global Use Policy
    - General protocols for resource sharing
    - => no additional configuration is required for access
Grid Computing | Evolution

- Generation 1 – Globus project (Goble & Foster)
  - Applications requiring high computing power
  - Includes protocols (LDAP, FTP) and heterogeneous development tools
  - Support for access and files transfer
  - Use Internet technologies, but ignore the Web
  - Employed mainly in academic environment
  - Sharing resources is achieved via GridFTP
  - Implementations: ...Legion, Condor, Unicore, ...
Grid Computing | Evolution

- Generation 2 – OGSA (*Open Grid Services Architecture*)
  - There is convergence of Service-oriented computing (SOC) and Grid Computing
    
    “Service-oriented Computing (SOC) is a new computing paradigm that utilizes services as the basic construct to support the development of rapid, low-cost and easy composition of distributed applications even in heterogeneous environments. The visionary promise of Service-Oriented Computing is a world of cooperating services where application components are assembled with a little effort into a network of services that can be loosely coupled to create flexible dynamic business processes and agile applications that may span organisations and computing platforms.” (Papazoglou et al. 2006)

- We notice the interoperability and sharing vision of SOC at application level versus Grid computing vision mainly at hardware level
  - Generation 1: Grid Computing architecture consists of protocols and services used to describe and share available physical resources
  - By using Web Services Standard (such as: WSDL, SOAP, BPL4WS,...) Grid protocols and Services can be described in a standardized manner
Grid Computing | Evolution

- Generation 2 – OGSA (Open Grid Services Architecture)

- Using the same standards => it was possible the convergence between Grid Computing and SOC => besides hardware and system resources, applications become shareable

OGSA:

“Building on concepts and technologies from both the Grid and Web Services communities, OGSA defines a uniform exposed service semantics (the Grid Service); defines standard mechanisms for creating, naming, and discovering transient Grid service instances; provides location transparency and multiple protocol bindings for service instances; and supports integration with underlying native platform facilities.” (Foster et al. 2002)
Implementation

- **Generation 2 – OGSA (Open Grid Services Architecture)**
  
  Grid services must be:
  
  - **Dynamic and volatile** – set of composed services that can be invoked or removed “on the fly”
  - **Ad-hoc** – there is no central location or central control
  - **Widespread** – orchestrating a large number of services (> 100) should be performed anytime
  - **Available** – potentially long-term (e.g. a simulation can take weeks)
  - **OGSI (Open Grid Service Infrastructure)**
    
    - OGSA Infrastructure - “accommodates” interactions between Grid resources and Web Services
    - Model implemented by Globus Toolkit 3.0
      
      - OGSI was replaced by WSRF (Web Service Resource Framework): WS- Security, WS- Management and other standards for Web Services => Globus 4.0
Grid Computing | Evolution

- Generation 3 – present and future
  - Convergence of Grid Computing and SaaS (Software-as-a-Service) paradigm
  - SaaS
    - Designates software that is owned, delivered and managed by a provider
    - It is used in the pay-per-use principle via a Web browser or APIs
  - Versus traditional software
    - The user pays for the time of use
    - The user does not have the software, he does not invest in the infrastructure or licenses
  - History: Application Service Provisioning (ASP) – appeared in 1988
    - It was a step for IT outsourcing and it comes with the idea of Web applications that can be provided by a central supplier (one-to-many delivery model)
      - The main problem: the inability to provide personalized services
      - Issues regarding scalability, robustness,....
Grid Computing | Evolution

- Generation 3 – present and future
- ASP problems can be solved by using Grid Computing + Web Services
  - Web Services allows services personalization
  - Grid Environment offer flexibility and scalability
  => many-to-many delivery model
Present and future

Overview

- Two directions of evolution:
  - Grid Computing
    - Mature technology
    - It provides computational power in *pay-per-use manner* => new business models for *utility computing*
  - There were many initiatives at hardware level: Sun, IBM, etc.
    - There were many initiatives at software level -> SaaS
      - Microsoft, SAP et. al.

- Next step...
  - A scalable, robust and reliable physical infrastructure,
  - Services that provide developers access to infrastructure by manipulating abstracted interfaces
  - SaaS running on a flexible and scalable infrastructure
Bibliography

• Massimo Cafaro, Givani Aloisio, Grids, Clouds and Virtualization, 2011
• Cloud Computing, Wu Chung – online resource
Abstract

Paradigms

– **Cluster Computing**

– **Grid Computing**
  • Definition
  • Architecture
  • Initiatives and Applications
  • Classification
  • Evolution
  • Present & Future -> **Cloud Computing**
Questions?