DSA 8430
Parallel Computing for Data Analytics

Parallel and Distributed Computing Concepts

Topic Intro Statement

Quickly, the limits of a single computer can be reached with even moderately sized data sets. Multiple computers can be logically organized and utilized as a single massive computing resource.

This topic covers programming paradigms for distributed systems, including both parallel and asymmetric processing.

Sub-Topics Outline

During this topic, we will discuss the following:

- Parallel (Symmetric) Distributed Programming
- Orchestration of asymmetric processing in distributed computing environments

Distributed Computing Architectures

- Computing on a distributed system
- Multiple autonomous nodes, linked through a high-speed network
- Architectures
  - Clusters
  - Grids
  - Clouds

Distributed Computation

Distributed Programming typically follows two models:

- Parallel (symmetric),
  - the problem is divided into a large number of sub-problems that are solved in parallel on multiple nodes, then the results are re-combined
- Network Services (asymmetric),
  - Computational tasks are delegated to an appropriate computing node that exists on the network

Cluster vs Grid vs Cloud

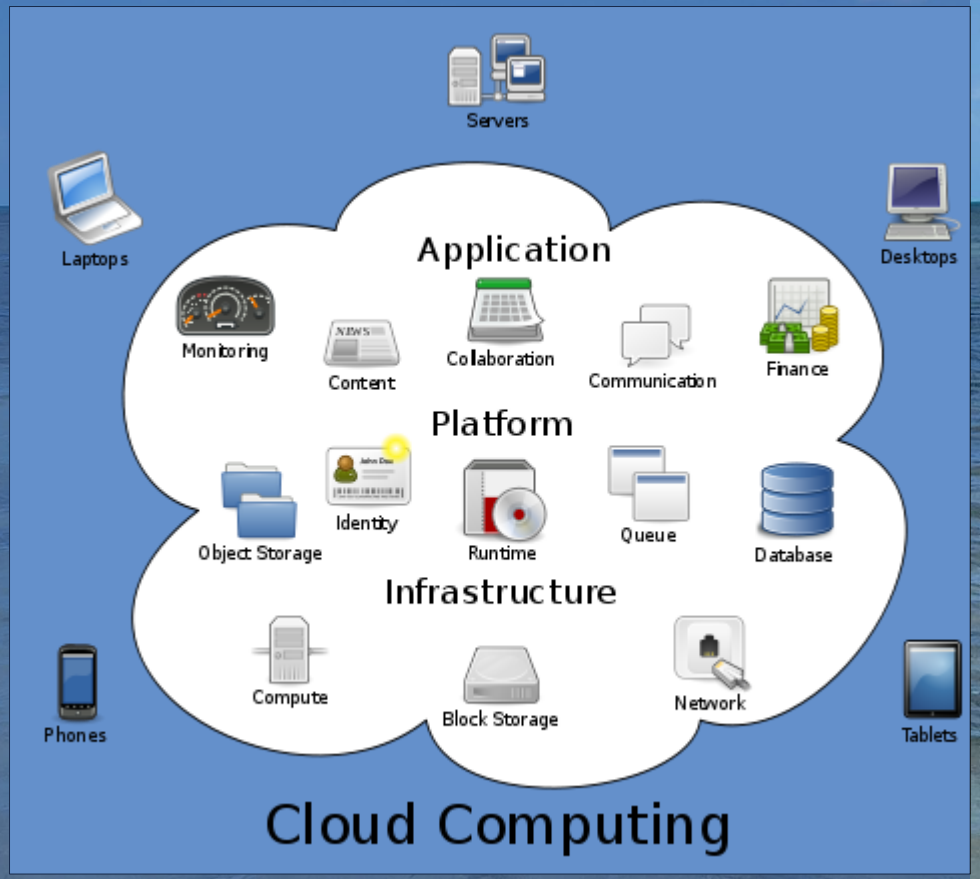
- **Cluster Computing** – a system composed of tightly couple nodes with shared file and network backbones
  - Supports Divide & Conquer, load balancing, and other distributed paradigms
- **Grid Computing** – a system of lightly coupled, often geographically diverse, heterogeneous systems
  - Typically working on a shared problem or providing a low-level compute capability
  - SETI@home, Folding@home, Einstein@home are all BOINC Grids
- **Cloud Computing –** Combining elements of cluster and grid computing to provide computational resources as a service

Cloud Computing

- Delivery of computing as a service
- Most of the technologies related to clouds have been around for decades!
- The proliferation of high-speed wireless networks is facilitating an explosion of cloud-based marketing and applications
- Cloud can represent
  - Abstract network storage
  - Web Apps, data repository
  - Elastic computation infrastructure

Cloud Computing

- Clouds provide services
- Software as a Service (SaaS)
- Data as a Service (DaaS)
- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)



(image: http://upload.wikimedia.org/wikipedia/commons/thumb/b/b5/Cloud_computing.svg/500px-Cloud_computing.svg.png)

Software as a Service

- Software code/framework and data is hosted centrally, access through a thin client, e.g. Web Applications
- In large business: Enterprise Applications
- Popular Application Domains
  - Collaboration
  - ERP, CRM, HRM, SCM
  - Content Management

Software as a Service

- Centralized code means centralized configuration
- Typically, a multi-tier application architecture horizontally scaled
- Commonly referred to as multitenancy
  - Branding / Skins often used to give appearance of multiple instances

Data as a Service

- Centralized data repository
  - Instead of local copies of databases
  - Save replication and centralized copy is always up-to-date
- Applications / or SaaS can be built over the DaaS
  - Data driven web sites, e.g. IMDB, Google Maps
  - Mashups: Combining multiple DaaS to create a hybrid product

Data as a Service

- Key components / advantages of DaaS
  - Agility: Simple data access methods, transform locally as needed for specialized uses
  - Cost-effectiveness: Data experts can work separate from data presentation, helps to isolate data architects from data end-users
  - Data quality: Tightly controlled data access services; never outdated or missing data (as a replication may suffer)

Infrastructure as a Service

- Computing infrastructure is provided on demand
  - Sub-set is Platform as a Service, e.g., a LAMP base or other software stack
- Storage
- Computing resources
- Software platforms / frameworks
  - Typically to allow someone to then provide SaaS or conduct cloud-based HPC

Orchestration

- *Programming in the large*
- Web Services Flow Language (WSFL)
- Business Process Execution Language (BPEL)
- How to guide data through complex workflows in a networked environment?
- Service Oriented Architectures (SOA) typically require orchestration middleware

Orchestration

- High Throughput Computing (HTC)
  - Requires large computing resources for extended periods of time
  - Data set is not a singular problem
  - Instead, the “data set” is an end-to-end computational flow
- Often HPC techniques used for one or more portions of the computation flow

Orchestration

- HTC
  - Maximizes number of jobs that can be processed, measured in months or years
  - vs. HPC measure how quickly a computational step can be accomplished
- HTC systems typically built over computing grids
- Descendant of main-frame batch computing

Orchestration

- HTC requires automatic orchestration
- Orchestration has to control processing in orderly fashion
  - Step dependencies
  - Scheduling on the grid
  - Jobs may have a mix of both parallel and serial steps
  - Step verification

Course Platforms

Google Cloud Platform

GCP

Amazon Web Services

AWS

NSF Pacific Research Platform

Nautilus