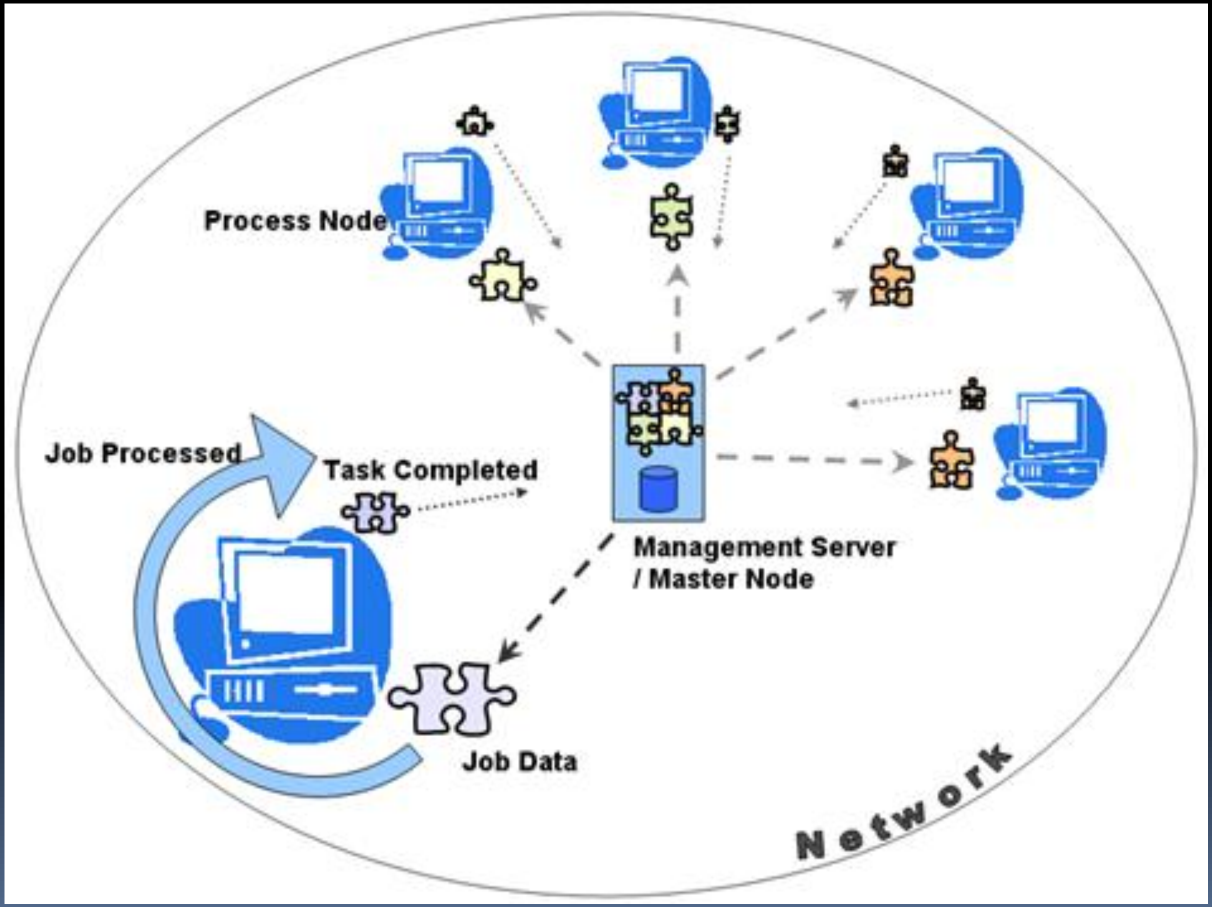
By Jacob White & Dan Harwell

DISTRIBUTED COMPUTING

Overview of Distributed Computing

- Field of Computer Science that studies distributed systems
  - A distributed system is a group of autonomous computers that communicate through a network
- Use of distributed systems to solve computational problems
  - Each problem is divided into many tasks, each of which is solved by one computer

Visual Representation



History of Distributed Computing

- ARPANET – predecessor of the internet
  - Late 1960s
  - ARPANET email was invented in the early 1970s
    - Probably the earliest example of a large-scale distributed application
- The first widespread distributed systems were LANs (local-area networks)
  - Invented in the 1970s
- Became its own branch of computer science in the late 1970s - early 1980s

Advantages of Distributed Computing

- Two main reasons to use distributed computing & distributed systems:
  - The vary nature of the application may require the use of several computers communicating with one another
    - Ex. Data is produced in a different location than it is needed
  - While a single computer may be capable of handling the work, a distributed system could be beneficial for practical reasons
    - Ex. May be more co$t effective to obtain the desired performance from a cluster of low-end computers rather than from a single high-end computer

Advantages of Distributed Computing

- More reliable than a single high-end machine
  - No single point of failure
- Easier to expand and manage than a monolithic uniprocessor system

Applications for Distributed Computing

- Telecommunication Networks
  - Telephone networks
  - Computer networks (internet)
- Network Applications
  - World Wide Web & Peer-to-Peer Networks
  - Distributed Databases
  - Banking & Airline reservation systems
- Real-Time Process Control
  - Aircraft Control Systems
  - Industrial Control Systems
- Parallel Computation
  - Scientific Computing (cluster, grid, volunteer computing projects)
  - Distributed Rendering in Computer Graphics

Architectures of Distributed Computing

- Client-Server
- 3-Tier Architecture (most web applications)
- n-Tier Architecture
- Tightly Coupled (clustered)
- Peer-to-Peer
- Space Based

Our Rough Road to Success

- Using ancient computers from the Computer Science closet
  - Multiple failing/corrupted hard drives
  - Multiple slow/faulty disk drives
  - Unreliable computers
    - Just because you can install Ubuntu doesn’t mean it will boot up the next time you power up the PC!

Our Rough Road to Success

- QADPZ (Quite Advanced Distributed Parallel Zystem)
  - Open source implementation of a system for distributed computing
  - A client-master-slave architecture, using XML format message based communication
  - Allows the management/use of the computational power of idle computers in a network
  - Users of the system can send computing tasks to these computers to be executed
    - in the form of a dynamic library, an executable program, or any program which can be interpreted
  - Supports Linux, Unix, Win32, and Mac OS X.

Our Rough Road to Success

- QADPZ (Quite Advanced Distributed Parallel Zystem)
  - Incompatibility with newest versions of Ubuntu
  - Last Stable Release – version 0.7
    - October 10, 2002 (over 8 years ago!)
  - Ubuntu 9.10 Final Release
    - October 29, 2009

Our Rough Road to Success

- Condor
  - specialized workload management system for computationally-intensive jobs
  - Provides:
    - a job queuing mechanism
    - scheduling policy
    - priority scheme
    - resource monitoring
    - resource management
  - Works with new versions of Ubuntu!
  - Last stable release – version 7.4.2
    - April 6, 2010

More about Condor

- Can use non-dedicated machines to run jobs
  - Harnesses power of idle machines without effecting machines being used at the time
- Simplifies the job submission process
  - Machine advertisements
- Maximizes efficiency
  - Intelligent job allocation
- Use of Universes (runtime environments)
  - Standard
  - Vanilla

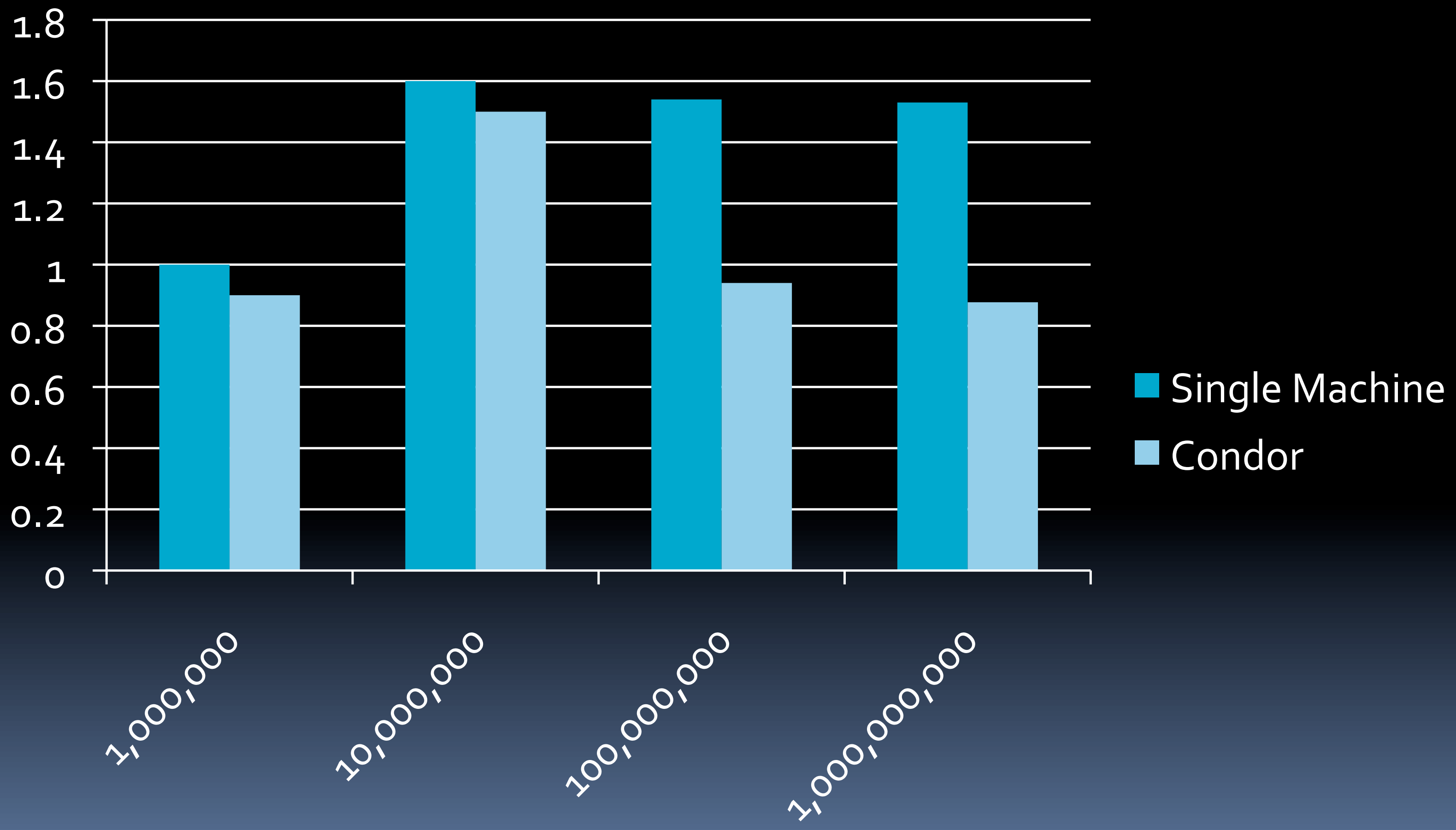
More about Condor

- Developed at the University of Wisconsin-Madison
- Open Source
- Runs on:
  - AIX
  - Solaris
  - HPUX
  - Linux
  - Unix
  - Mac OS X
  - FreeBSD
  - Contemporary Windows OS’s

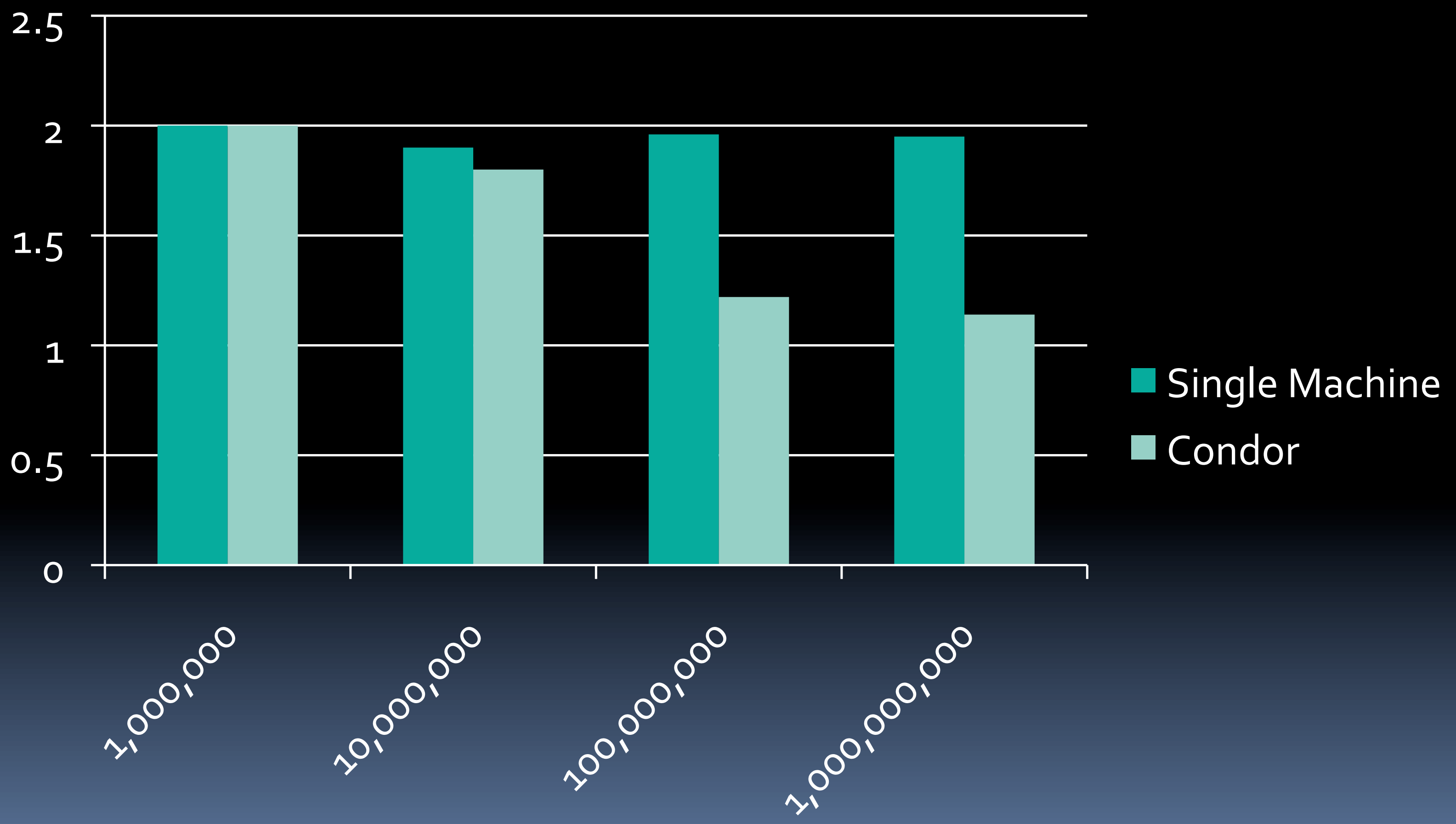
Demonstration

- *It’s demo time!*

AddMultiples - integers



AddMultiples - doubles



Fibonacci

- 1,000,000,000 integers
  - Single machine – 12.05 seconds
  - Condor – 7.76 seconds

Special Thanks!

- Dr. Kirk for guidance throughout the semester
- UU Computer Science Department for the computers
- Dr. Wilms for the KVM, Cart, etc.

Resources

- http://en.wikipedia.org/wiki/Distributed_computing
- http://qadpz.sourceforge.net/
- http://www.naccq.ac.nz/bacit/0203/2004Caukill_OffPeakGrid_files/2004CaukillFigure1.jpg
- http://www.cs.wisc.edu/condor/
- http://en.wikipedia.org/wiki/Condor_High-Throughput_Computing_System