INTELLIGENT JOB ROUTER

by

Kartik Vedalaveni

A THESIS

Presented to the Faculty of

The Graduate College at the University Of Nebraska

In Partial Fulfilment of Requirements

For the Degree of Master Of Science

Major: Computer Science

Under the Supervision of Dr. David Swanson

Lincoln, Nebraska

May, 2013

INTELLIGENT JOB ROUTER

Kartik Vedalaveni, M.S

University Of Nebraska, 2013

Adviser: Dr. David Swanson

Most schedulers available come with plethora of features and options for customizations that fulfills myriad goals of a custom built clusters and data centers but there are some important functions required to run clusters at its peak efficiency. One such case is when filesystem crashes over large I/O workload and we see performance of the cluster degrading when there is absence of balance in load pertaining to either I/O, network, CPU or RAM usage. These issues that we encounter in real life at Holland Computing Center are the basis and motivation for tackling these set of problems with a goal to run clusters and data centers at high efficiency. The end result is the development of a Co-Scheduler for the cluster that minimizes if not solves the problem of performance degradation that is caused when some of the resource is throttled to a greater extent for some kind of workloads.

DEDICATION

Dedicated to

ACKNOWLEDGMENTS

Thanks

Contents

Chapter 1

Introduction

Although modern schedulers in clusters provide innumerable features, the problem of cluster performance degradation that occurs due to improper load balancing hasn't been addressed yet. The problem of performance degradation when many jobs are scheduled on single system either based on processor equivalence or based on number of processor slots. Some of these schedulers are smart enough to take into account contention of other resources like RAM but ultimately convert the 2D vector values of CPU and RAM into a single scalar value which equals to hard-coding the value or presenting these resources in some kind of ratio makes us question effectiveness of such scheduling mechanism.

1.1 Co-Scheduler

Chapter 2

Background

HTCondor is a distributed system developed by HTCondor team at the University of Wisconsin-Madison.It provides High-Throughput Computing environment to sites that foster research computing and enables sites to share computing resources when workstations are idle at a given site. HTCondor system includes a batch queuing system for a pool of workstations, HTCondor runs on both UNIX and windows based workstations that are all connected by a network.

2.1 High-Throughput Computing

The workloads that run on condor system doesn't have an objective of how fast the job can be completed but how many times can the job be run in the next few months.To be precise, European Grid Infrastructure defines HTC as a computing paradigm that focuses on the efficient execution of large number of loosely coupled tasks.

2.2 Open Science Grid

Open Science Grid(OSG), provides service and support for resource providers and scientific institutions using a distributed fabric of high throughout computational services. Initially OSG was created to facilitate data analysis from the Large Hadron Collider and about 70% of the resources are used on analysis of data from particle colliders. OSG doesn't own resources but provides software and services to users and enables opportunistic usage and sharing of resources among resource providers.The main goal of OSG is to advance science through open distributed computing. The OSG provides multi-disciplinary partnership to federate local, regional, community and national cyber-infrastructures to meet the needs of research and academic communities at all scales.

Bibliography