Proposal

Andrew Rosen

September 21, 2014

Contents

1 Hypotheses **3**

1.1 DHTs are better for distributed computing under many circumstances . . 3

Robustness and Fault-Tolerance 3

Load Balancing 3

Scalability 3

Heterogeneity 3

Ease of Adding, Removing, Maintaining, and money related factors 3

1.1.1 Different or subproblem: Certain DHTs are better at one application than another due to differences 3

Design Differences Impacts 3

Geometries 3

Implementation Differences Impacts 3

Recursive or iterative seek 3

2 Justification and Why I Think It's Cool **4**

2.0.2 DHTs well understood 4

2.0.3 DHTs are Highly used for their intended purposed 4

Bittorrent, WoW 4

2.0.4 DHTs are being effectively leveraged for other things besides file sharing already 4

PaaS 4

Load Balancing in the cloud 4

Computing is a natural extension 4

3 Possible Experiments and Applications **5**

3.0.5 Map Reduce 5

ChordReduce 5

Comparison of MapReduce paradigm on different DHTs 5

3.0.6 High End Computing 5

Metadata Management 5

Robustness 5

3.0.7 Graph Processing on a DHT 5

Embedding 5

Distribute the work for solving a graph on a DHT 5

Comparison to well established or state of the art methods . . . 5
3.0.8 Machine Learning Problems on A DHT 5
Bayesian Learning 5
3.0.9 DHTs as a volunteer Platform 5

4 DHT Background 6

Chapter 1

Hypotheses

1.1 DHTs are better for distributed computing under many circumstances

Distributed Hash Tables (DHTs) are traditionally used as the backbone of P2P file-sharing applications.

Robustness and Fault-Tolerance

Distributed Hash Tables are designed with a couple of assumptions in mind. One of the most prominent is that DHTs are deployed on a non-static network.

Recent research in using DHTs for High End Computing [**?**] shows what can happen if we remove this assumption by placing the network that is almost completely static.

Load Balancing

Scalability

Heterogeneity

Ease of Adding, Removing, Maintaining, and money related factors

1.1.1 Different or subproblem: Certain DHTs are better at one application than another due to differences

Design Differences Impacts

Geometries

Implementation Differences Impacts

Recursive or iterative seek

Chapter 2

Justification and Why I Think It's Cool

2.0.2 DHTs well understood

2.0.3 DHTs are Highly used for their intended purposed

Bittorrent, WoW

2.0.4 DHTs are being effectively leveraged for other things besides file sharing already

PaaS

Load Balancing in the cloud

Computing is a natural extension

Chapter 3

Possible Experiments and Applications

3.0.5 Map Reduce

ChordReduce

Comparison of MapReduce paradigm on different DHTs

3.0.6 High End Computing

Metadata Management

Robustness

3.0.7 Graph Processing on a DHT

Embedding

Distribute the work for solving a graph on a DHT

Comparison to well established or state of the art methods

3.0.8 Machine Learning Problems on A DHT

Bayesian Learning

3.0.9 DHTs as a volunteer Platform

Chapter 4

DHT Background

Bibliography