



Trinity College Dublin

Coláiste na Tríonóide, Baile Átha Cliath

The University of Dublin

School of Computer Science and Statistics

An Investigation into Building a Multiplayer Online Game Using Named Data Networking

Stefano Lupo

14334933

March 9, 2019

An MAI Thesis submitted in partial fulfillment
of the requirements for the degree of
MAI Computer Engineering

1 Abstract

2 Introduction

[1]

2.1 Background

Existing IP based internet: host abstraction, where it comes from, How it has scaled, What it supports / doesn't support (multicast etc), History of ICN (CCN \rightarrow NDN, Parc etc), Thin waist

3 State of the Art (15-25)

3.1 NDN

3.1.1 Names

Interests, Data, how names are structured. Hierarchical naming to enable scalable routing analogous to IP

3.1.2 Structure of NDN Packets

Diagrams etc

3.1.3 Key Data Structures

FIB, PIT, CS

3.1.4 Caching

3.1.5 Routing

Longest prefix, NLSR

3.1.6 Security

Digests, trust anchors etc

3.1.7 Multipath forwarding (not spanning tree, loop detection)

Every piece of data can be uniquely named, use router memory to know when seen interest before, no restriction on single path forwarding

3.1.8 NFD

3.1.9 NLSR

3.1.10 NDN Tools, Libraries and Frameworks etc

Implementations of CCNx or w/e, useful tools etc

3.2 Real Time Applications using NDN

3.3 Distributed Dataset Synchronhization

Summarize the survey paper

3.3.1 ChronoSync

3.3.2 PSync

3.3.3 CCNx Sync

3.4 Video Game Architectures

3.4.1 Taxonomy of data

3.4.2 Architectures (C/S, P2P, Hybrid)

3.4.3 Dead Reckoning

3.4.4 Interest Zone Filtering

4 Problem Statement

5 Design

Explain the things MOGs must do

5.1 Player Discovery

5.2 Sync Protocol

5.2.1 Differences from existing protocols

outstanding interest catchup, only one writer

5.2.2 Naming Design

5.3 Interaction

This is not handled in Egal Car

5.4 Optimizations to Sync Protocol for Data types

5.4.1 PLayer Status

5.4.2 Projectile

5.5 Game Design

5.5.1 Protos

5.5.2 Very brief game functionality

5.5.3 Linkage between game and backend

diagrams of pubs / subs / game engine

6 Implementation

6.1 Building Game with LibGdx / Ashley

Remote update reconciliation

6.2 Java Backend

6.2.1 Sequence Numbered Cache

6.3 Testing

6.3.1 Docker

6.3.2 NLSR

Building topologies Automating players, simulators, INCREDIBLES

6.3.3 Latency Calculations

7 Evaluation

Key things to examine: scale, overhead (packet size vs app data), latency Evaluation
Matrix Push to breaking point

8 Conclusion

Bibliography

[1] Albert Einstein.

Appendices

App. A Source Code for DNS Management System