A DISTRIBUTED SOURCE LOCATOR MODEL FOR NAME RESOLUTION IN NAMED DATA NETWORK

MAYS SALMAN OBAID

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
UNIVERSITI UTARA MALAYSIA
2016

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Abstrak

Kebelakangan ini terdapat peningkatan dari segi jumlah peranti yang dihubungkan dengan Internet dan jumlah ini dijangka meningkat pada masa hadapan. ICN merupakan satu konsep baru untuk Internet pada masa hadapan. Banyak projek yang terangkum dalam ICN telah diselidiki dan satu daripada projek ini ialah NDN. Kajian ini bermatlamat untuk mereka bentuk pelokasi sumber agihan untuk Sistem Resolusi Nama (NRS) untuk mengelak daripada berlakunya titik kegagalan apabila hanya satu sistem berpusat yang beroperasi. Model baru ini dilaksanakan dalam seni bina NDN untuk memastikan penemuan sebarang objek dalam rangkaian tanpa perlu mencari data secara hop by hop. Kajian ini mengupayakan Kaedah Penyelidikan Reka Bentuk (DRM) dan memperkenalkan tahap utama berdasarkan sifat kajian. Model konsep untuk kajian ini dibina berasaskan kajian lampau NRS dalam projek ICN yang lain dan juga bersandarkan model Chord dalam jadual cincangan teragih (DHT). Kuantiti data yang sangat besar serta kepanjangan nama yang tidak tetap perlu diambil kira dalam penghasilan NRS yang berkesan untuk NDN. Selain itu, sistem sebegini memudahkan lagi agihan data yang sejajar. NDN yang juga projek baru di bawah konsep ICN masih kurang diselidiki dan mempunyai pelbagai masalah yang perlu diselesaikan. Tambahan pula, setakat ini komponen nyata untuk NDN masih belum ada dan kebanyakan operasi dikendalikan dalam bentuk simulasi. Memandangkan kajian ini tertumpu kepada agihan pelokasi sumber untuk NRS, sumbangan kajian lebih terarah kepada usaha yang lebih terjamin untuk mencari objek data dalam seni bina NDN dan menambah baik masalah boleh skala yang wujud dalam rangkaian. Perkara ini boleh menyokong penghalaan data dan pemindahan antara nod serta mengurangkan lalu lintas pertukaran secara keseluruhan. Ini membolehkan penyelesaian satu masalah terbuka yang besar dalam seni bina NDN dan seterusnya merancah letak atur asas konsep Internet yang baru dalam rangkaian ICN. Pengguna juga boleh memindahkan data dengan lebih pasti dan lebih berkesan. Sumbangan utama kajian ini, termasuklah reka bentuk Pelokasi Sumber Teragih (DSL) untuk Resolusi Nama. Kajian ini turut menyumbang dari segi agihan jadual cincangan untuk carian data yang lebih baik dan lebih pantas. Agihan ini juga bermanfaat kepada pengguna kerana pengguna boleh menentukan aras data serta meningkatkan lagi keselamatan rangkaian data. Hal ini boleh memaksimumkan penggunaan sumber rangkaian.

Kata kunci: ICN, NDN, Sistem Resolusi Nama, Jadual Cincangan Teragih, Strategi Ajuan

Abstract

Recently, the number of devices that are connected to the Internet had been significantly increased with much more expected increment in the future. ICN is a new concept for future Internet that has been developed, many projects within the ICN concept are being researched and NDN in one of them. The purpose of this research is to design distribution source locator for Name Resolution System to avoid the point of failure that may occur if there is only a central system and implemented this new model in NDN architecture to guarantee findings of any object in the network instead of looking for data hop by hop. This research employs the Design Research Methodology (DRM) and introduces its main stages according to the nature of this research. The conceptual model had been designed based on the previous study of NRS in other ICN projects, and according to Chord model in the distributed hash table (DHT). The huge amount of data and unfixed name length in NDN architecture are the main points that must be taken into consideration in order to produce an efficient NRS for NDN. Furthermore, such system simplifies the distributing of the data that correspond to it. NDN is a new project under ICN concept and it is still under research with many issues that is needed to be solved, also there is no real component to work on NDN and all work had been done based on simulation environment. Since the present research focuses on distributing the source locator for NRS, the major contribution of this study is to provide a guaranteed way to find the data object in NDN architecture and to improve the scalability issues in the network. This will support the data routing and transfer between the node and reduce the overall exchanged traffic. This permits the development of solving one of the major open issues in NDN architecture and thus aids in supporting the deployment of the new Internet concept base on the ICN networks. It will thus help users to transfer data reliably and more efficiently. The major contributions of this study include the design of a new Distributed Source Locator (DSL) for Name Resolution. Other contributions are the way of distributing the hash tables for better and faster data lookup, on the other hand, this distribution gives the users the privilege to specify the data levels which results in an increment in the data security of the network. All these would contribute toward the maximized utilization of network resources.

Keywords: ICN, NDN, Name Resolution System, Distributed Hash Table, Forwarding Strategies

Acknowledgements

In the name of Allah, Most Gracious, Most Merciful. Praise and peace be upon His beloved our Prophet Muhammad (SAW), his family and his companions. By the will of God, we escaped darkness into enlightenment, with this spirit that I set out to undertake the current study, and the quest for self-actualization provided the additional push that kept me going and finally sees this thesis come to its expected conclusion, Alhamdulillah.

My deepest gratitude is to my supervisor Professor Dr. Suhaidi Hassan. I have been amazingly fortunate to have a great supervisor who gave me the freedom to explore on my own, and at the same time the guidance to recover when my steps faltered. Prof., Suhaidi taught me how to question thoughts and express ideas. His patience and support helped me overcome many situations and finish this dissertation. I hope that one day I would become as good advisor to my students as Professor Dr. Suhaidi Hassan has been to me.

I am also indebted to the members of InterNetWorks Research Lab Group with whom I have interacted during the course of my studies. Particularly, I would like to acknowledge Dr. Adib and all of the Phd students.

My heartiest gratitude goes to my family, my first teacher my father Dr. Salman which is my idol and my inspiration in my whole life, to the greatest woman in my world my mother Afaf who always has faith in me and prays for my success. Most importantly, none of this would have been possible without his love, patience, kindness and support my beloved husband (Ahmed) and also huge thank for my little angel Fatima for being so understanding and receptive.

Special thank goes to my mother in law Amina for all the support and pray that made me stand again each time I feel down. I am also grateful to my uncle Ali Alwardi and my aunt Widad for being my second father and mother, my brother Ahmed and

my sisters Abeer and Ghada who are always ready to push me forward, my father in law, my brothers and sisters in law. Also , I would like to thank all of my friemds, co-workers and collegues who helped me to acheive this difficult task, their support and care helped me to stay focused on my studies, special thanks goes to Samara, Ibtihal, Dr.Raed, Sharaf and Hawazin.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

EB	Exa Byte
P2P	Peer to Peer
IP	Internet Protocol
ICN	Information Centric Network
CCN	Content Centric Network
NDN	Named Data Network
NDO	Named Data Object
NRS	Name Resolution System
MDHT	Multi-level Distributed Hash Table
NR	Name Resolution
OSPF	Open Short Path First
ISIS	Intermediate System-to-Intermediate System
BGP	Border Gateway Protocol
DSL	Distributed Source Locator
ODHT	Overlay Distributed Hash Table
DRM	Design Research Methodology
IRTF	Internet Research Task Force
ICNRG	Information Centric Network Research Group
DNS	Domain Name Sysyem
TCP/IP	Transmission Control Protocol/ Internet Protocol

CDN	Content Delivery Network
DONN	Data Oriented Network Architecture
PURSUIT	Publish Subscribe Internet Technology
NetInf	Network Information
COMET	COntent Mediator
ANR	National Research Agency
NBR	Name Based Routing
URL	Uniform Resource Locator
PKI	Public Key Infrastructure
OI	Object Identifier
AS	Autonomous System
RH	Resolution Handler
PARC	Palo Alto Research Center
CR	Content Router
FIB	Forwarding Information Base
PIT	Pending Interest Table
CS	Content Store
LPM	Longest Prefix Match
EU	European Union
RN	Rendezvous Nodes
RENE	REndezvous NEtwork
DHT	Distributed Hash Table
TM	Topology Management

FN	Forwarding Node
SAIL	Scalable and Adaptive Internet SoLutions
CMP	Content Mediation Plan
CURLINC	Content Ubiquitous Resolution and Delivery Infrastructure for Next Generation Services
CRS	Content Resolution System
PC	Path Configurator
VDI	Versatile Digital Item
BN	Border Node
IN	Internet Node
GUID	Globally Unique IDentifier
GNRS	Global Name Resolution System
LLC	Late Locator Construction
RC	Research Classification
DS-I	Descriptive Study-I
PS	Prescriptive Study

CHAPTER ONE
INTRODUCTION

In the last 50 years since the packet network creation, computer systems and their component had become cost effective and available everywhere. The numerous communication methods that the Internet offer and low cost of data storage, allow the access for a huge of new content " in 2008 alone 500EB (Exa Byte) of data were created" [3]. Users keep looking for content in term to the value of the Internet, but in the other hand, communication still looking for the place of the content in term of communication, the incompatibility of the two models led to a number of issues. Availability is one of these issues, fast and reliable content access requires awkward mechanisms, especially for some application in some mechanisms like P2P networks. Another issues is security like content trust, which is easily unavailable and depending on untrustworthy connection information and location. Location dependence from the other hand is also have several problems in aspect of mapping the content to a host location, which put complications in configuration and implementation of network services [4].

As a result of these problems, and after years of experimental researches and increment in the attentiveness of unsolved problems in contemporaneous Internet architectures like IP, the Information Centric Network (ICN) concept has been created and followed by many approaches and the idea of the Content-Centric Network (CCN) had been created, then the Named Data Networking (NDN) (which is related to the CCN) was appeared and can be considered as one of the future of the Internet architectures [5]. As per any new project, several issues had been identified with the NDN architecture. One of them is data finding, or how to name the data and organize it to ensure fast data lookup and delivery. One idea to name the content in a scalable and easy way to retrieve can be done by depending on hierarchical naming "name tree". One more open issue is the scaling of NDN. In term of data transmission, NDN depend on the name based routing tactic. Request forwarding in name based routing tactic is directly carried out based on the identifier alone and some arrangement of information is setup along the way these set of arrangement data help the content to travel back to the requester.

The names in NDN are longer than IP addresses, but hierarchical architecture aids the effectiveness of lookup and global accessibility of the data. However the problem with name-based routing approaches is that it does not guarantee the discovery of the content [6]. To guarantee the finding of any Name Data Object (NDO) in the network, a Name Resolution System (NRS) is needed, which it has the ability of tracking data in any node, but with one drawback, as the NRS itself can be considered as a network point of failure. Due to this, new techniques need to be introduced to enhance this network's failing possibility. [6].

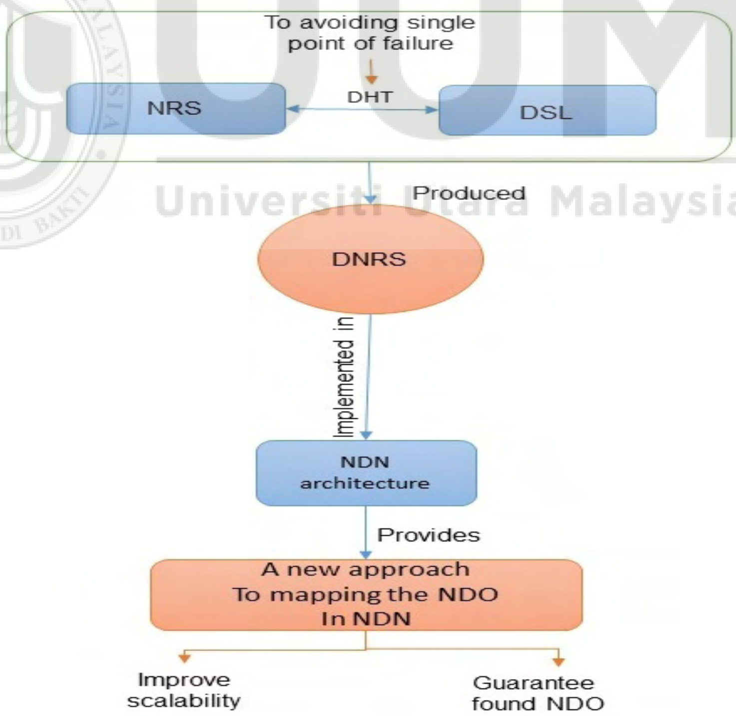


Figure 1.1: Research Overview

1.1 Research Background

This research aims to propose a conceptual model that prevent the NRS from being as a point of failure and implement this model to NDN architecture in order to increase its performance in data routing. This will be done by using Multilevel distributed hash table (DHT) mechanism to distributed the name resolution system since it is compatible with the naming scheme that the NDN use in identifying the object

1.1.1 Named Data Networking (NDN)

The earlier projects on the content centric network put the roots for NDN which initially presented overtly by Van Jacobson in 2006. The NDN project investigating evolution to a data-centric network architecture from nowadays host centric network architecture (IP). The belief is that this conceptually simple shift will have a far-reaching effect for people in how to design, develop, deploy, and use networks and applications [4].

A communications architecture built on named data NDN has no connotation of the lowest level of the host. A packet "addresses" the names content not location. NDN moves to replace where with what by change all the network's stack components from IP packets to the named content chunks. The abstract of host-to-host conversations in networking was chosen to handle the problems of the 1960s. The named data is a better solution for the problems of today's communication rather than named hosts[4].

1.1.2 Naming Resolution System (NRS)

To have efficient data publishing, Name Resolution (NR) approach provides a means for client to search for NDO by name. This approach needs additional entity called Name Resolution System (NRS) to provide the translation. NRS should have the ability of using to any available data copy in the network. This minimize the traffic in the network by choosing the most appropriate set of copy(s) available.

The NRS should also have the ability to retrieval the data which it is near the subscriber with link to some network metric in such cases of being the two end-point exists in one network-domain the data route should be also exists in that domain. This will lead the inter domain traffic to be reduce. The other functionality of NRS is that a portions of it can be worked and used with no direct full work is required. A system that satisfies such requirements, like reduces internal provider traffic, offers essential motivations for providers to deploy such an NRS in their networks. Providing the providers a managerial autonomy of their NRS subsystem further simplifies deployment [2].

1.1.3 Distributed Source Locator

A significant characteristic of any computing system is in the way of how the system mapping the names into the objects and efficient information distribution to make use of any available data in order reduce network load and latency and increase information availability. This represent in the responsibility of the computing system's name resolution mechanism. Because the naming mechanism supports references to objects, it directly influences both the ease with which users refer to objects, and the degree of object sharing allowed in the system [7].

There are several distributed and decentralized system types, one of them is a distributed hash table which it supply a service of lookup for the similar to a hash table. The hash tables is a data structures that use a hash function to efficiently map identifiers to associated values, these identifiers known as a key. A hash function allows to perform a trivial calculation, which will directly refer to the data rather than needing to look through an entire database[8].

In a distributed hash table, any active node have the ability to retrieve the data quickly based on keys, instead of only one central server, accountability for this mapping is distributed between the active nodes, and in this way in disruption in one connected node will not affect the whole system in continuous to this design, distributed hash tables can provide very big numbers of nodes, while handling continuous arrivals, leavings. The data stored in the hash table may not be the actual data requested; it may simply be a link to the data. For instance, when using a distributed hash table as part of a search engine, the result returned from the system may be a link to the server hosting the content searched for rather than the content itself[8].

The MDHT system a group of host computers in a sub-net participate in a DHT overlay as a single node[9]. It is independent of the underlying transport and forwarding/routing layer. For example, data can be transferred by traditional topology-based routing (e.g., OSPF, ISIS, BGP) to ensure an effective data transmission via the topologically best path [2].

1.2 Research Motivation

NDN architect, still considered to be an open case with a lot of pending issues. In fact, this provides the researchers the needed area to work on how to enhance the shape of NDN in a way to make it the shape of the future Internet. An added value of this approach is the availability of its own simulator that can be used to facilitate the work on its related issues. One of the possible open enhancement in NDN is to add an NRS to its architecture. This is can be done by utilizing the currently available NRSs. Having an NRS will support for better information dissemination and make information access independent from storage locations by taking the benefit from copies that are available in the network. However, and as mentioned before in the introduction of this chapter, NRS itself can be a network point of failure which may lead to lose many available data in the network. To overcome this, a distributed source locator for the NRS is being proposed.

1.3 Problem Statement

The paradigm of (ICN) addressing number of current Internet's issues and the content distribution architectures. Generally, the new requirement for efficient and scalable content distribution has supported a diffusion of overlay networks such as Peer-to-Peer networks and Content Distribution Networks. The idea of the ICN is to provide access model where end-point basically try to get named content, without the needed for transport layer session that is engaged to a specific host (or even a network interface). Moreover, the other worries about the scalability of the Internet routing infrastructure led to research for alternative approaches in routing and addressing which are not constrained by the current host-locator- based routing [10].

In NDN architecture the routing is completely based on prefixes of content names this approach called name based routing, so it does not need a resolution system. This routing approach causes scalability and mobility issues [11]. Furthermore it does not guarantee the discovery of content [6]. Network components such as the NRS could perform additional services and facilitation of communication between a group of senders and receivers where each set is identified by a single name [12]. The NR steps enables a subscriber to search for NDO by using names. The first step is to map the name and the source locators where the NDO is stored, the second step is to forward the request message from the subscriber to the source. However, this two-steps technique need an additional entity which is called the Name Resolution System (NRS) for providing translation.

One of the drawbacks of this mechanism is that, the NRS itself can be a Point-of-Failure (softwre and hardware failure)which can results into many of the NDO published and registered on that NRS to be unavailable [13]. The other drawback is that NRS requires a large storage to store NDO mapping. However, this approach guarantees the finding of the requested NDO since NRS already provides a pointer to the NDO source[6].

1.4 Research Questions

The main question in this research is how to avoid the NRS from being a point of failure in order to implement it in the NDN architecture?

- How to avoid a point of failure in NRS by using DSL?
- How to implement NRS in NDN architecture in order to have a full database of the network's content locations?
- How to evaluate the research solution and its enhancement to the NDN networks in finding and forwarding data?

1.5 Research Objectives

The main objective of this research is to design a Distributed Source Locator (DSL) scheme for name resolution in NDN to address the Point-of-Failure problem in a Name Resolution System.

- To design a conceptual Distributed Source Locator (DSL) scheme for Name Resolution to avoid being a Point-of-Failure.
- To implement the DSL for NRS in NDN architecture by using ODHT mechanism.
- To evaluate the proposed model by comparing its performance with the current solution in a simulation environment in term of reducing the packet number and amount of traffic on all node.

1.6 Research Scope

This research dealing with Distributed Source Locator mechanism called DHT to distribute Name Resolution System in order to make it suitable with name scheme of NDN which is one of the ICN approaches to enhance the performance of the data routing of this approach. This is mainly can be done by adding an NRS for the NDN. This work will start by finding a mechanism to distribute the NR in a way to avoid any issues that may affect the the system before implement it in NDN.

1.7 Significance of the Research

This research has proposed a Distributed Source Locator (DSL) model for NRS in NDN. This is needed to have a significant process of mapping or matching a content name from the publisher/provider/source up to the subscriber/customer/distention by founding the proper route. This process will provide many advantages in terms of scalability, guaranty of founding any available nodes in the network and congestion, also from the other hand, the availability of DSL will make the system failure-proof as it will continue to work whenever any node got out of service.

1.8 Organization of the Thesis

This thesis is organized in five chapters, where the following is a summary of key chapter highlights:

Chapter One presents an overview about the thesis as a whole starting with simple introduction to define what the is NDN or the future Internet and to review the idea behind this research by explain one of the problem that this new project facing in problem statement section. Furthermore, what is the suggestion to solve this problem has been presented through the following sections.

Chapter Two critically evaluates the related work and what is the solution taking in facing such problem in other projects developed under ICN concept.

Chapter Three presents the Design Research Methodology (DRM) as the research framework to conduct this study and combines several methods adopted to propose and implement DSL for NRS in NDN architecture

Chapter Four concerns about modeling the DSL for NRS to preventing it from being a point of failure and represent the evaluation of the proposed model in terms of memory usage and data availability. After that this chapter moving to achieve the second objective in this research in how implemented the proposed model to make it compatible with the NDN architecture in terms of naming, naming storage management and security at the end of this chapter the evaluation result for the impact of this research had been presented.

Chapter Five states the conclusion as well as the contributions of the research work presented in this research, then suggests future directions for further studies for develop the proposed model and enhancing the forward mechanism to make it more satisfy for all research requirement

CHAPTER TWO
LITERATURE REVIEW

2.1 Introduction

The new idea of Information Centric Networking (ICN) shows new model in communication that highlights on what data is being transmitted on the network rather than what network components are transmitted the data. From the ICN point of view, contents are the network's major component instead of hosts [1]. The ICN has drawn attention the Internet research community to the contents as the main network part rather than the hosts in the ordinary network typologies. The Internet future based on ICN architecture also draw high attention to the Internet research community and this led to great increment in the interest of the ICN approaches. This attention was confirmed by the foundation of the ICN Research Group (ICNRG http://irtf.org/icnrg) within the Internet Research Task Force (IRTF), meanwhile, ICN projects are still on the research phase to create the Future Internet architecture [2]. Section 2.2 of this research will explain the motivation for the ICN. From the other hand, the ICN approaches will be shown in Section 2.3. Section 2.6 compares these approaches in terms of naming and data routing. While the source locator will be clarified in section 2.7.

2.2 Motivation for ICN

Based on the previous forecasts, IP traffic worldwide would increase four times from 2009 to 2014 to reach about 64 Exabytes per month in 2014 compared with 15 Exabytes per month only on 2009 [14]. Moreover, the international mobile traffic is predicate to double every year in 2014, the increment is 39 times from 2009 till 2014. This traffic mainly include video (TV, VOD, Internet video and P2P) which took more than 91 percent from the whole traffic in 2014 [10]. This huge growth in the Internet and the appearance of new applications based on the current needs, rise new requirements from the network architecture respective such as security, supporting of scalable content distribution, trust and mobility. The Internet was not supposed to handle and address such needs, so in order to help it make its way, various patches and work around solutions appeared, such as Mobile IP. Most of these work around solutions make the Internet architecture more complicated and proved to be only temporary solutions [1].

At the moment, the available solutions like TCP/IP will be unable to handle the current situation anymore and subject to specific issues. For example, in order to locate a content, it need to be mapped to a host name, then the DNS server to change this name will be translated to its related location i.e. IP address. The two-steps mapping incurs access overhead. From the other hand, IP addresses are lined to their location and it does not handle content or node mobility. Also, security is another issues, since the host of the main component, they are being targeted by many security threats. In addition, IP routers are stateless and does not have caching functionality, this makes the same request to be made several times in the same route and resulting unneeded bandwidth waste. Due to the previous, the current researches are aiming to shift the Internet architecture form being host centric to information centric [6].

The first appearance of the Internet showed an ad-hoc manner. Several work around solutions were found for the Internet in order to manage new requirement as they appear. For example, the DNS provides the main name resolution functionality, the fact that it was designed and implemented long time after the Internet deployment. As the DNS main purpose is to resolve the address of a host in the Internet, it does not have the capability for supporting content movement, awareness of location and replication. To beat these problems, Content Distribution Networks (CDNs) were implemented. also, Peer-to-Peer (P2P) file sharing systems (like Torrent and Gnutella) were developed for mluti-source content retrieval, rapid content dissemination and inherent replication.

These techniques were obviously have good contribution in improving the content access over the Internet. However, they considered as a work around solution and not a permanent one with the optimal performance that working in the underlying network topology [15].

2.3 Information Centric Network (ICN) approaches

Information-centric networking is the hot research topic in recent years, with various research initiatives like DONA [16], CCN/NDN [17], PSIRP/PURSUIT [18], 4WARD " http://www.4ward-project.eu ", CONVERGENCE " http://www.ict-convergence.eu ", NetInf [19], COMET " http://www.comet-project.org ", MobilityFirst [20] and ANR connect http://www.agence-nationale-recherche.fr/?Project=ANR-10-VERS-0001. Content is targeting this emerging research area with the aim of shifting from the current Internet architecture which is built and designed for a host-to-host communications model [13].

ICN manages the process of routing and forwarding NDO packets through two different steps namely the name resolution which is a 2-way mechanism and the Name Based Routing (NBR) which is just a 1-way mechanism [13]. The name resolution and data routing in ICN design can be integrated together to form a coupled approach where the requested content by a subscriber is routed to content providers and then this provider sends the content back to the subscriber by using the reverse path over which the request was forwarded. When the two functions are not integrated together, it form the decoupled approach where there are no restrictions on the path that will be used to forward data to and from a subscriber to the content publisher

The ICN design models for the future Internet architecture available so far, have proposed different naming systems, as well as different data routing processes. Some ICN architectures perform name resolution and data routing separately, in such a way that they perform name resolution first and then perform data routing. Other ICN proposals tries to combine these transactions together. The following section provides in depth description of the naming, name resolution and data routing process in ICN approaches [13].

2.4 Naming

Naming is one of the ICN features in which the naming mechanism of each data object (like web pages, videos, photos, documents, live streaming and interactive media) is an important aspect. Three features under naming in ICN have been proposed which are name-data integrity that establish a verifiable binding between the object and its name; object authenticity which ensures authentication of received objects to a receiver in such a way that the received object represent the actual object published on the network, and the object's provenance which enables to know who published the object on the network. The proposed naming mechanisms of ICN architectures are location independent.

These proposed naming schemes in ICN are the hierarchical namespace, which has a similar structure like the Uniform Resource Locator (URL), and it have the advantages that it enables routing information aggregation, and the routing system is improved in terms of scalability and flat (self-certifying namespace), means that the name-data integrity is to be verified with no need of a Public Key Infrastructure (PKI). Flat naming offers uniqueness and persistence. In general, naming mechanisms in different ICN designs may range from flat to hierarchical and sometimes other approaches employ the human readable naming scheme which enables a user to type the names manually and retrieve contents according to his or her needs [13].

In ICN, the information itself is names instead of having a specific source-destination address in order to communicate. In addition, no data can be received in the requester side unless it is explicitly requested, unlike the Internet where the senders have full control of the data being exchanged. From the other hand, in ICN, if a request is submitted, the network have the responsibility to locate the best source that can give the needed information. Routing of information requests thus seeks to find the best source for the information, based on a location-independent name [1].

2.5 Name Resolution and Data Routing of ICN

ICN has two major roles that must be achieved when there is a request for a specific Named Data Object (NDO). The first one is to find a node (e.g. contents server) that stores a copy of the NDO, and forward the request to that node. The second is to find a path from that node back to the requester over which the NDO can be delivered. One way to do this is through name resolution, which means to get one or more lower-layer locators for the name of NDO. These locators can be used to retrieve the object. An alternative way is to directly route the request to that node based on the NDO's name. This is often referred as name-based routing. In name-based routing scheme name resolution step is often omitted [21].

The name resolution and data routing in ICN design can be integrated together to form a coupled approach where the requested content by a subscriber is routed to content providers and then this provider sends the content back to the subscriber by using the reverse path over which the request was forwarded. When the two functions are not integrated together, it form the decoupled approach where there are no restrictions on the path that will be used to forward data to and from a subscriber to the content publisher. The ICN design models for the future Internet architecture available so far have proposed different naming systems as well as different data routing processes. Some ICN architectures perform name resolution and data routing separately in such a way that they perform name resolution first and then perform data routing. Other ICN proposals tried to combine these transactions together.

The naming resolution process includes two steps: the first step involves the content name resolving to one or several locators, while the second step involves the routing of the request towards one of these locators by the mean of the shortest path routing. From the other side, in name based routing, the forwarding of the request is being directly performed based on the identifier (name) alone, and some sort of information is set-up along the route that the content is passing through in its way back to the requester [22]. Between the current available research projects, NetInf and PURSUIT follow the name resolution approach, but from the other hand, NDN follow the name based routing approach. The following section provides in depth description of the naming, name resolution and data routing process in ICN approaches [13].

2.5.1 Data Oriented Network Architecture (DONA)

While many projects, starting with TRIAD proposed extending the Internet with content routing capabilities, the Data Oriented Network Architecture (DONA) from UC Berkeley is one of the first complete ICN architectures.

2.5.1.1 Naming in DONA

The naming structure in DONA is flat in the form P:L. Part P represent the Object Identifier (OI) which is unique in the global domain and is defined as the concept that containing the cryptography hash of the public key of the publisher. From the other side, the name of L is unique in the local domain (in the scope of P) that uniquely identifying an object with the local scope. Syntax L is left to the principle's (P) that may select only to provide a reading form that is readable by human or even cryptographic hash of the object's content. The P:L naming is making sure that each OI in DONA is unique in the global domain. [23, 1, 6].

2.5.1.2 Naming resolution and data routing in DONA

In order to achieve the name resolution, DONA is using an NRS that have a Resolution Handler (RH). Any Autonomous System (AS) should have one RH at least. All RHs in the topology are connecting to each other to produce a Name Resolution System (NRS) with hierarchy shape as shown in the Figure 2.1.

The publishing of the OI inside the network is achieved by permitting the sources to register the data into the NRS (arrows 1-3). When the source want to register a specific name to its local RH, the RH will saving a pointer in a table to map this OI with the source that register it and forwarding this OI to its relative parent and peering ASs, this is will cause that each intermediate RH will record a mapping between the address and the OI of the RH that had been delivered the registration. As all registrations are moved up towards Tier-1 ASs which are peers with each other, then Tier-1 will have all of the registered OIs. Authoritative sources can also register wild cards OIs (P1) to RHs in their local domain, showing that they have the capability to provide services for all the P1 principals requests.

To resolve an OI, a request is sent by a subscriber in a form a FIND(OI) message towards the local RH (arrows 4-7). the local RH will look up its mapping table to check if the requested OI is found by a pointer mapping it, then the FIND(OI) message will follow the route of the pointers that created by the registration procedure till reaching the publisher. From the other hand, if the requested OI is not found during mapping the local RH, the FIND(OI) message will be forwarded to the parent RHs till finding the requested OI by mapping it by the pointer. Since Tier-1 have the full picture of all the registrations in the system, then at the end, a pointer will be found if the request OI is exists in the system. It's possible to couple or decouple the data routing that returned to the subscribe (arrows 8-11). In the coupled, the traversed RHs during the resolution steps and the answer with data is send to the subscriber using the reverse path. In the decoupled situation, the answer with the data is routed irrespective to the routing path of the FIND(OI) message by using the ordinary IP routing and forwarding [23, 1, 16].

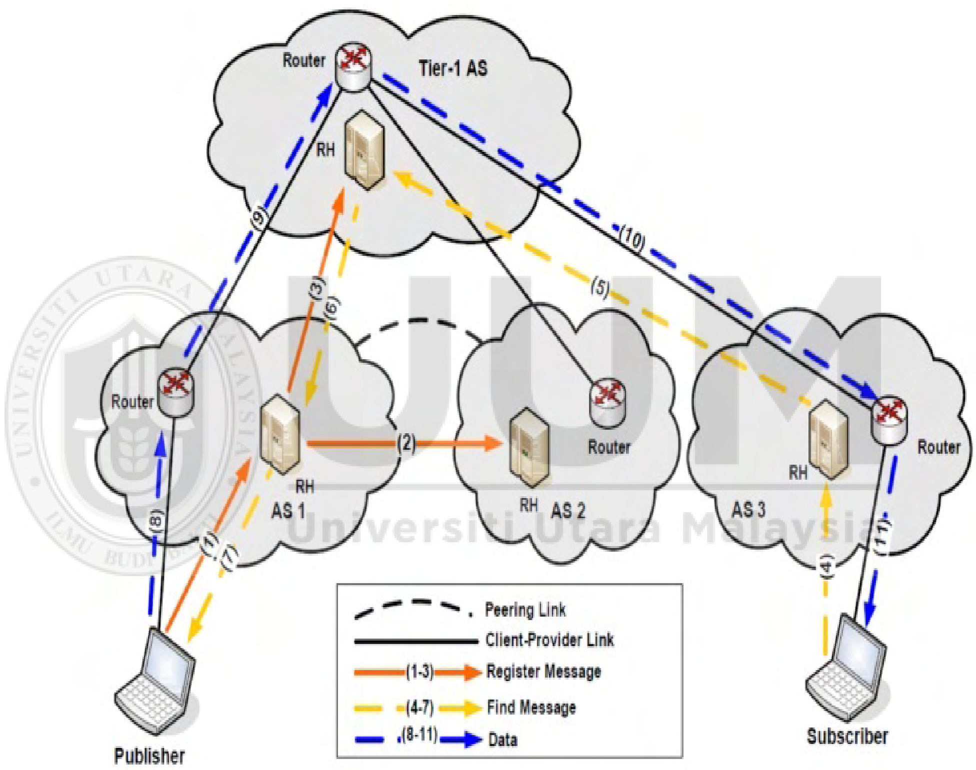


Figure 2.1: DONA approach [1]

2.5.2 Named Data Networking (NDN)

Another pioneering of Internet architecture approach is NDN. It is from PARC which follows the ICN concept. Its essential concepts were described in a Google tech talk, long time prior to the first paper which draw attention on the CCN architecture was published. The US future Internet architecture program found the NDN project which is also developing the CCN architecture. NDN architecture is reforming the Internet protocol stack by making the named data exchange is the thin waist of the Internet architecture, using many networking technologies below the waist for connectivity, including, but not limited to, IP. In NDN a strategy layered between the named data layer and the underlying network technologies to optimize resource usage, e.g., to select a link in a multi-homed node, while a security layer applies security functionalist directly on named data [1, 23].

2.5.2.1 Naming in NDN

Name in NDN is hierarchical, It’s not mandatory of OIs in NDN to be in the form of human readable in all hierarchical layers so the (www.os3.nl/courses/cia/<dns.pdf content hash>) can also be an OI hierarchy. Structure of the naming formation in NDN is very ticklish for the architecture’s scalability, while it permits aggregation of OI prefixes in OI resolution and data routing. An NDN name can be an identifier of an endpoint, a service in an endpoint, a file, or even a command to turn on some lights [13, 23, 1].

Thus allowing name resolution and data routing information to be aggregated across similar names, something considered to be critical for the scalability of the architecture.

2.5.2.2 Naming Resolution and Data Routing in NDN

In NDN, subscribers send Interest packet to request some information objects. This Interest packet will arrive in the form of Data packet, carrying the name of the requested information object. As shown in Figure 2.2, packets forwarded by Content Router (CRs) in form of hop by hop. CR maintains three data structures: the Forwarding Information Base (FIB), the Pending Interest Table (PIT) and the Content Store (CS). The FIB is responsible on maps information names to the output interface(s) that must be used to forward Interest packet to appropriate data sources. The PIT then tracks the incoming interface from which pending interest packet have arrived. Finally, the CS serves as a local cache for information objects that have passed through the CR.

When an Interest arrives, the CR extracts the information of the name and looks for the information object whose name is matching the requested prefix in its CS . If it finds something, then it will send it back immediately through the incoming interface in a data packet and the Interest is discarded. Else, the router using the longest prefix match on its FIB in order to decide where to forward this interest packet. If an entry is found in the FIB, the router will register the interest's incoming interface in the PIT and pushes the interest to the CR indicated by the FIB. As showing in Figure 2.2, the subscriber sends an interest for the name */aueb.gr/ai/new.htm* (arrows 1-3).

If the PIT already contains an entry for the same name, meaning that this exact information object had already been requested, the router adds the incoming interface to this PIT entry and discards the interest, effectively forming a multicast tree for the information object. When an information object that matches the requested name is found at a publisher node or a CS, the Interest message is discarded and the information is returned in a data packet. This packet is forwarded back to subscriber(s) in a hop-by-hop manner, based on the state maintained in the PITs. Specifically, when a CR receives a data packet, it first stores the corresponding information object in its CS and then it performs a Longest-Prefix Match (LPM) [24] in its PIT to locate an entry matching the data packet; if a PIT entry lists multiple interfaces, the data packet is duplicated, thus achieving multicast delivery. Finally, the CR forwards the data packet to these interfaces and deletes the entry from the PIT.In case there are no matching entries in the PIT, the router discards the data packet as a duplicate.

In NDN name resolution and data routing are coupled, since data packet follow the pointers left in the PITs by Interest packet, therefore routing is by definition symmetric. In order to populate the FIBs, NDN can use distributed routing protocols like OSPF, in which CRs advertise name prefixes rather than IP address ranges, e.g., a router could advertise /aueb.gr to inform the network that it can provide information objects whose prefix is /aueb.gr. A CR may have multiple interfaces in its FIB for a prefix, for example, if it is multi-homed or if it is aware of multiple CDN servers hosting the information. In this case its strategy layer may choose to send the Interest either to all these interfaces (if multiple Data packet are returned, all but the first are automatically discarded) or only to the interface that has exhibited the best performance so far [17, 1].

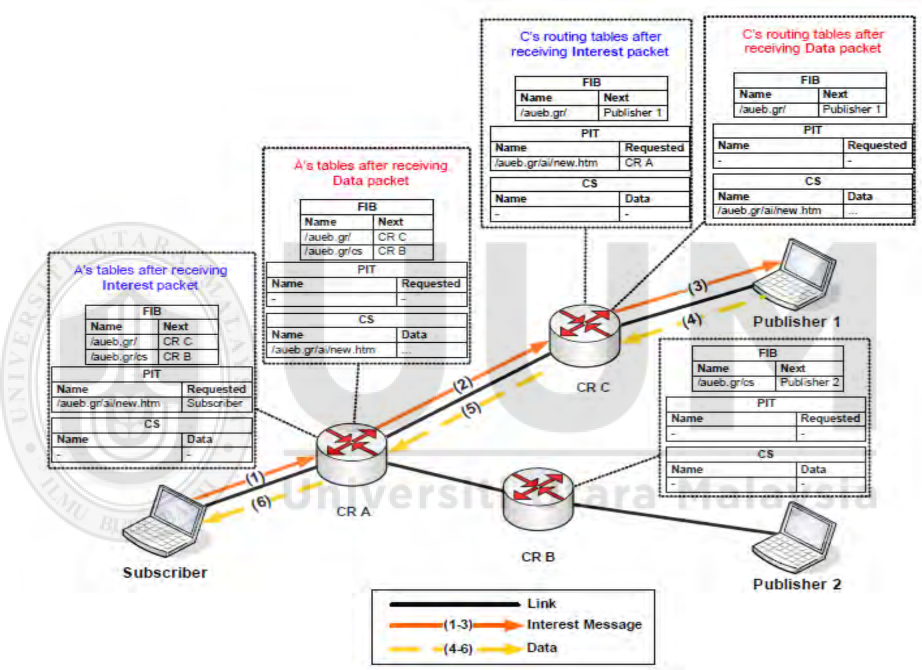


Figure 2.2: NDN approach [1]

2.5.3 Publish Subscribe Internet Technology (PURSUIT)

A new project and its continuations were found by EU Framework 7 Programme. The PURSUIT produced an architecture that completely replaces the IP protocol stack with a publish-subscribe protocol stack.

2.5.3.1 Naming in PURSUIT

In PURSUIT information objects represented by a unique pair of IDs, which are the scope ID and the rendezvous ID. The scope ID groups the related information objects while the other ID (rendezvous ID) represents the effectual identity for a specific part of information . Information objects may belong to multiple scopes (possibly with different rendezvous IDs), but they must always belong to at least one scope. Scopes serve as a means of a) defining sets of information objects within a given context and b) enforcing “boundaries” based on some dissemination strategy for the scope. For example, a publisher may place a photograph under a “friends” scope and a “family” scope, with each scope having different access rights.While PURSUIT names are flat as in DONA, scopes in PURSUIT can be organized in scope graphs of variable forms, including hierarchies, therefore a complete name consists of a sequence of scope IDs and a single rendezvous ID, thus generalizing the DONA naming scheme.

2.5.3.2 Naming Resolution and Data Routing in PURSUIT

PURSUIT architecture consists rendezvous function, topology management function and forwarding function, each function separate from the other functions. When the function of rendezvous matches a subscription to a publication, it directs the function of topology management to create a route from the publisher to the subscriber. This route is finally used by the forwarding function to perform the actual transfer of data. Name resolution is handled by the rendezvous function, which is done by a collection of Rendezvous Nodes (RNs), the Rendezvous Network (RENE), implemented as a hierarchical DHT.

If the publisher want to publish some information, he needs to export a publish message for its local RN to advertise an information object .the NR will then route it to the other RN in corresponding scope ID. In the other hand the subscriber need to send subscribe message for this information object to its local RN, the subscribe message will routing by the DHT to the exact RN. Topology Manager (TM) node will direct by NR to create a route to connect the publisher with the subscriber to deliver the data. The TM sends that route by a START PUBLISH message to the publisher to use it to send the information object by a set of Forwarding Nodes (FNs). The TM nodes in PURSUIT jointly implement the topology management function by executing a distributed routing protocol to discover the network topology.

The actual delivery paths are calculated upon request by the rendezvous function as a series of links between FNs and encoded into source routes using a technique based on Bloom filters. Specifically, each network node assigns a tag with long bit string produced by a set of hash functions to each of its outgoing links, and advertises these tags via the routing protocol. A path through the network is then encoded by ORing the tags of its constituent links and the resulting Bloom filter is included in each data packet. When a data packet arrives at a FN, the FN simply ANDing the tags of its outgoing links with the Bloom filter in the packet; if any tag matches, then the packet is forwarded over the corresponding link. In this manner, the only state maintained at the FNs is the link tags [1].

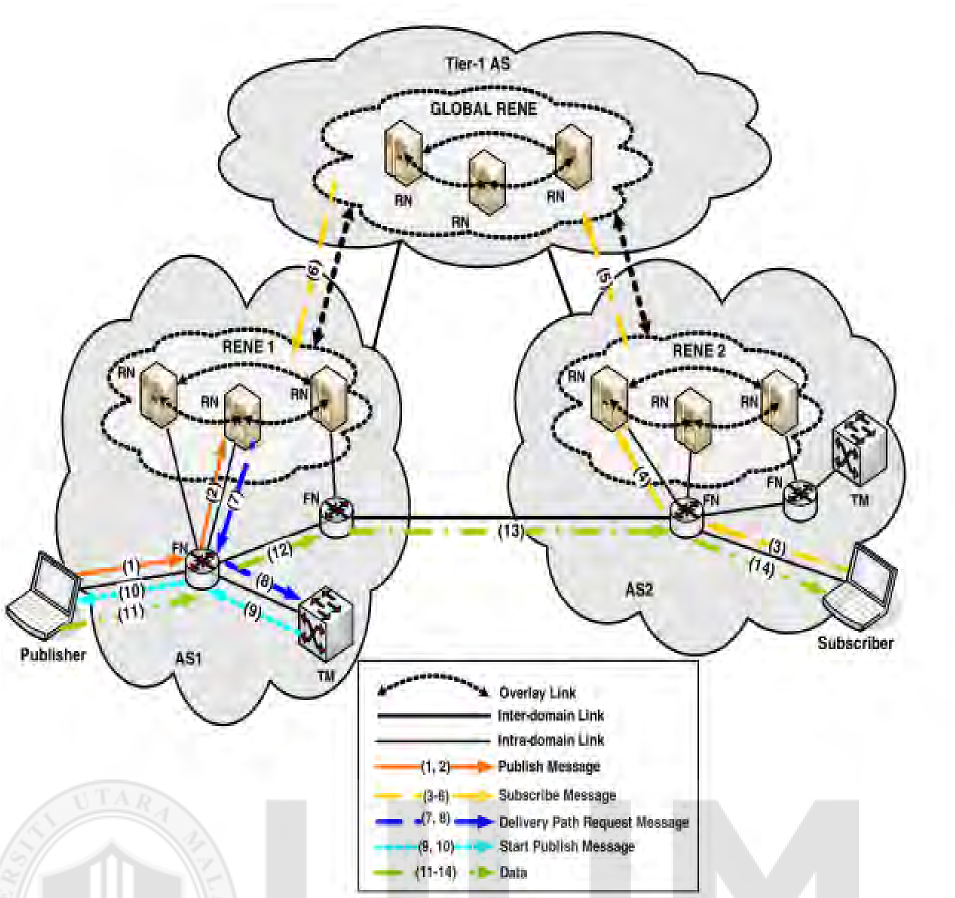


Figure 2.3: PURSUIT approach [1]

2.5.4 SAIL

The design and architecture for the future Internet (4WARD) project and its continuation Scalable and Adaptive Internet Solutions (SAIL), both funded by the EU Framework 7 Programme, are investigating designs for the Future Internet and ways to facilitate a smooth transition from the current Internet. While both projects have a very wide scope, this survey will focus on one of their research areas, the Network of Information (NetInf), which designs an ICN architecture that supports the exchange of named information objects. Beyond the aspects covered below, the SAIL architecture includes many other services, such as searching for information objects via keywords. The SAIL architecture is very generic one: it merges elements available in the NDN and PURSUIT approaches and can even work in a hybrid mode. Furthermore, it can be implemented over different routing and forwarding technologies, by introducing convergence layers to translate SAIL messages to actual network packets [1].

2.5.4.1 Naming in SAIL

Flat structure is what Naming formation in SAIL in the form of A/L. Part A represents the authority and is globally unique of the OI. While part L represents the object and is locally unique within the authority. This naming mechanism can guarantee the OI is globally unique. Each part can be a cryptographic hash or a regular string "(e.g. ni://Authrity_1/<Hash of DNS.pdf data>, ni://<Hash of Authority_1 public key>/DNS.pdf, ni://< Hash of Authority_1 public key>/<Hash of DNS.pdf>)". Naming in SAIL can have the same scenario in NDN by make the structure of OI in hierarchical form [23].

2.5.4.2 Name Resolution and Data Routing in SAIL

Name resolution and data routing in SAIL can be coupled or decoupled, and even hybrid operation is possible. In case of the decoupled, NRS is used to map object names to locators that can be used to reach the corresponding information object, such as IP addresses. The NRS is some form of DHT, either a multilevel DHT or a hierarchical SkipNet. In the multilevel DHT solution, each authority maintains its own local NRS to handle the resolution of the L part, while a global NRS handles the resolution of the A part.

A publisher makes an information object available by forwarding a PUBLISH command message with its locator to the local NRS, which stores the L to locator mapping. The local NRS aggregates all the L parts for the same authority A into a Bloom filter, and sends a PUBLISH message to the global NRS . The global NRS stores the mapping between the authority A plus the Bloom filter and the local NRS, replacing any previous such mapping. When a subscriber is Interested in an information object, it can send a GET message to its local NRS which consults the global NRS in order to return a locator for the object. Finally, the subscriber sends a GET message to the publisher, using the returned locator, and the publisher responds with the information object in a Data packet in the coupled case, a routing protocol is used to advertise object names and populate the routing tables of Content Routers (CRs), as in NDN. A subscriber sends a GET message to its local CR, which propagates it hop-by-hop towards the publisher or a cache.

When the information object is found, it is returned via a Data packet, reversing the path taken by the GET message. However, in contrast to NDN where pointers left in CRs are used for the return path, in SAIL the GET messages accumulate routing directions along their path, which are simply reversed at the publisher or cache in order to reach the subscriber. In the hybrid mode of operation, the NRS returns routing hints, that is, partial locators that can direct a GET message in one or more directions where more information about the requested information object may be found. A GET message can thus start with some routing hints from the NRS to reach the vicinity of the requested information object, and then exploit name-based routing information stored in the CRs to reach its destination. Alternatively, a GET message can start with the name-based routing information stored in the CRs and resort to the NRS for further routing hints when a CR does not have sufficient information to forward it. As a result, routing in SAIL can be a mix of hop-by-hop and partial paths [1].

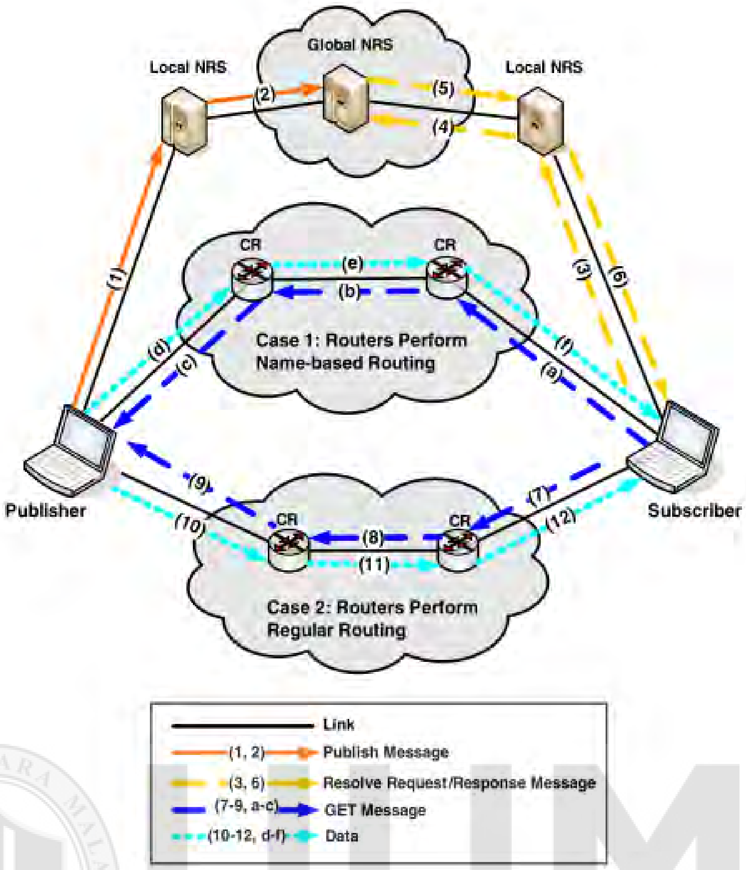


Figure 2.4: SAIL[1]

2.5.5 COMET

The COntent Mediator architecture for content-aware nET-works (COMET) project, funded by the EU Framework 7 Programme, is designing mechanisms for optimizing information source selection and distribution by mapping information to appropriate hosts or servers based on transmission requirements, user preferences, and network state . The core component of the COMET architecture is a Content Mediation Plane (CMP) which mediates between the network providers and the information servers, being aware of both information and infrastructure. The COMET project has produced two very different architectures for the CMP: a coupled design called Content-Ubiquitous Resolution and Delivery Infrastructure for Next Generation Services (CURLING), which is an ICN architecture with coupled name resolution and routing, and a decoupled design that enhances information delivery without fundamentally changing the underlying Internet. Unlike other ICN approaches which strive for location independence, COMET allows both subscribers and publishers to explicitly include location preferences for information, following established business practices. For example, a subscriber may ask for bookstores in a specific country, and a publisher may only make videos available to a specific country .

2.5.5.1 Naming in COMET

A precise naming scheme has not been defined for COMET. However, in COMET the information names are provided by a Content Resolution System (CRS) when the information is registered by the publishers, thus allowing names for related information to be explicitly aggregatable, e.g., episodes of a TV series can have sequential names. This allows the naming system to scale by exploiting existing relationships between information objects.

2.5.5.2 Name Resolution and Data Routing in COMET

COMET use the coupled approach as presented in Figure 2.5. When the publisher that wants to make some information available sends a REGISTER message to its local CRS node which issues a name for the information and stores the actual location of the information, e.g., the IP address of the publisher (arrow-1). This information is propagated upstream in the AS hierarchy using PUBLISH messages, so that each parent CRS ends up with a pointer to its child CRS that sent the PUBLISH message (arrow 2). The publisher may limit the propagation of this information to a specific area, e.g., an IP prefix, so PUBLISH messages may not reach the Tier-1 provider.

A subscriber that is Interested in some information issues a CONSUME message to its local CRS, which is similarly propagated upwards in the CRS hierarchy until it reaches a CRS that has information about that name (arrows 3-4). The subscriber may either limit the propagation of this information to a specific area or exclude specific areas from this propagation. When a match is available, the CONSUME message will go after the pointers in the CRSs to reach the actual publisher (arrows 5-6). As the CONSUME message travels from the subscriber to the publisher, each CRS on the way installs forwarding state at the Content-aware Routers (CaRs) of each intermediate AS, pointing back towards the subscriber (arrows 3a-5a). The publisher can thus send the corresponding data to the subscriber by using these pointers (arrows 7-10).

While the coupled approach in COMET shares many ideas with DONA on name resolution and with NDN on data routing, there are some important differences. With respect to name resolution, in COMET the PUBLISH messages are not propagated to peering AS's, but only to parents, in order to reduce the state maintained at CRSs. This has two implications: first, when a CONSUME message reaches a tier-1 provider without finding a match, it must be propagated to all other Tier-1 providers to guarantee that a match will be found, if one exists, since all tier-1 providers are peers with each other; second, both name resolution and data routing (which are coupled) do not exploit peering links, therefore additional signaling is needed to switch to peering paths if available. For example, in Figure 2.5 data routing can switch to the peering link between AS1 and AS2 .

With respect to data routing, while in NDN both name resolution and data routing use the same CRs, in COMET name resolution uses the CRSs while data routing uses the CaRs, thus allowing the CRSs in each AS more flexibility in choosing the most appropriate paths between the available CaRs of that AS. The decoupled approach in COMET is also presented in the Figure 2.5 . In this case, the CRS system is similar to DNS, in that the CRSs split the object namespace among themselves in a fixed hierarchical manner. This means that when a publisher wants to make some information available, it simply sends a REGISTER message to its local CRS (arrow 1), which is not propagated further because it must belong to the namespace assigned to that CRS. When a subscriber issues a CONSUME message for some information (arrow 2), this is resolved by the root CRS to a pointer towards the publisher's CRS (arrows 3-4). The subscriber's CRS contacts the publisher's CRS to get the location of the publisher (arrows 5-6).

Then the subscriber's Path Configurator (PC) contacts the publisher's PC (shown co-located with the CRS nodes for simplicity) requesting a source route from the subscriber to the publisher (arrows 7-8). This source route is returned to the subscriber (arrow 9) which uses it to request information (arrows 10-12); its reverse is used by the publisher to return the information (arrows 13-15). COMET's decoupled approach has some of the limitations of DNS, for example, names are location-dependent due to the fixed assignment of namespace areas to network areas. As a result, it cannot be considered a true ICN architecture, hence in the remainder of this paper the term COMET architecture will refer only to the coupled approach [1].

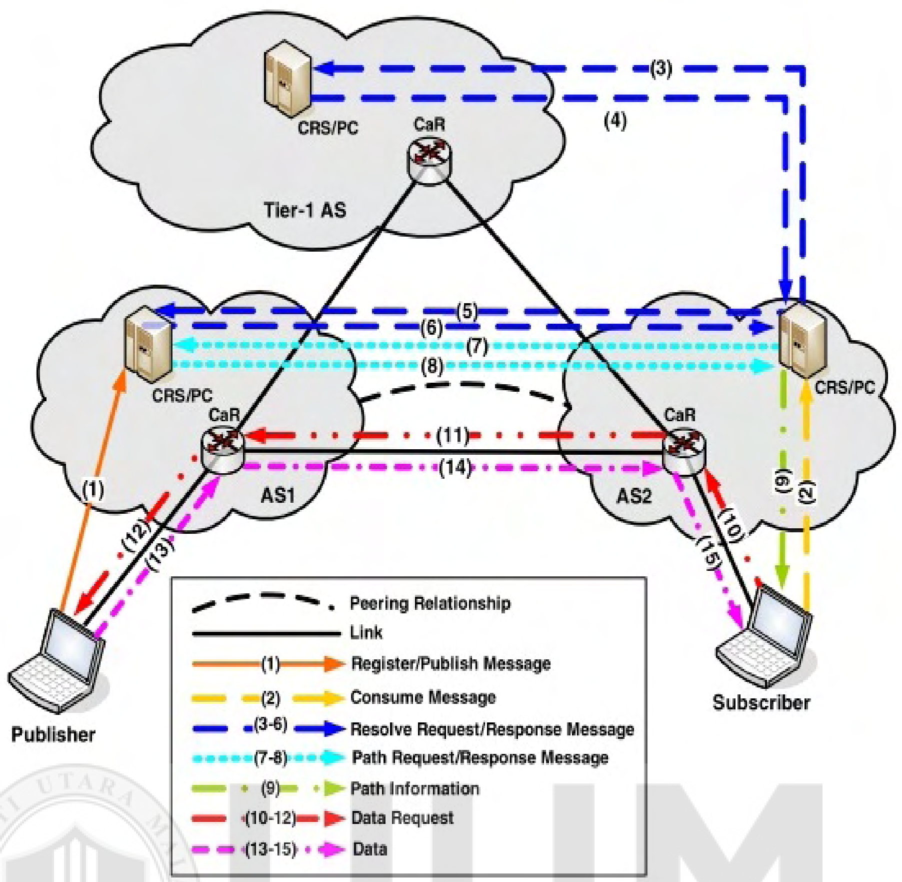


Figure 2.5: COMET approach[1]

2.5.6 CONVERGENCE

The CONVERGENCE project, funded by the EU Framework 7 Programme, envisions an ICN-based Future Internet that facilitates user access to information, spanning from digital data and services to people and real world objects. Each such object in CONVERGENCE is represented by a Versatile Digital Item (VDI), a common container for all kinds of digital information . A Content Network (CONET) allows publishers to make available VDIs and subscribers to express Interest in those VDIs. A distinguishing characteristic of the CONVERGENCE architecture is that it attempts to ease transition from IP by reusing existing functionality. For example, since CONVERGENCE messages are expected to be large due to naming and security meta-data, rules are defined for splitting them to carrier packets, e.g., IP data- grams. Furthermore, an IP header option has been defined to carry the essential information from CON-

VERGENCE message headers, allowing CONVERGENCE-aware IP routers to treat IP datagrams containing CONVERGENCE messages differently[1, 25].

2.4.6.1 Naming in CONVERGENCE

In CONVERGENCE object names consist of a namespace ID and a name part. The name part format its determined by the namespace ID.While the default format of CONVERGENCE names is similar to that in DONA, and the hierarchical names may also be used as in NDN, or even URLs. The exact properties of the names depend therefore on the specific namespace used. Since CONVERGENCE is most similar to NDN, we assume in the following the use of hierarchical names. [1].

2.4.6.2 Name Resolution and Data Routing in CONVERGENCE

The CONVERGENCE architecture, shown in Figure 2.6, has many similarities with NDN; indeed, its prototype has been implemented as a modification of the NDN prototype. Subscribers issue interest packet requesting an information object, which are forwarded hop-by-hop by Border Nodes (BNs) to publishers or internal Nodes (INs) that provide caching. Publishers respond with data packet which follow the reverse path (arrows 7-10).

In order to reduce the state requirements at the BNs, CONET diverges from NDN in three aspects. First, BNs do not maintain name- based routing information for every advertised name prefix, but only for a small portion of them, hence their routing table operates like a route cache. If an Interest message cannot be forwarded because there is no routing information for the corresponding name, the BN consults an external NRS, e.g., DNS, in order to find out how to forward the Interest.

Second, as Interest packet are propagated they accumulate the network addresses of the BNs they pass, allowing the publisher to route the Data packet by reversing this path information, without requiring the maintenance of pointers at BNs. Third, BNs do not have to be directly connected; instead, the path between two BNs can involve multiple hops, e.g., via IP routers as shown in the figure below, hence their designation as border nodes. Therefore, unlike CRs in NDN, BNs map names to network addresses, e.g., IP addresses, rather than to interfaces. In CONVERGENCE name resolution and data routing are coupled, since the path taken by a Data packet is the reverse of the path followed by the corresponding Interest packet, even though each step of this path may not be a single hop but an entire IP path, hence the path segments between BNs which an Interest packet and its corresponding Data packet follow are not necessarily symmetric. The NRS is used if an appropriate route is not found at some BN. The details of the NRS used have not been defined by the CONVERGENCE project. The name-based routing tables at BNs may also be partially populated without resorting to the NRS, by running a routing protocol for name prefixes, e.g., OSPF, as in NDN[1].

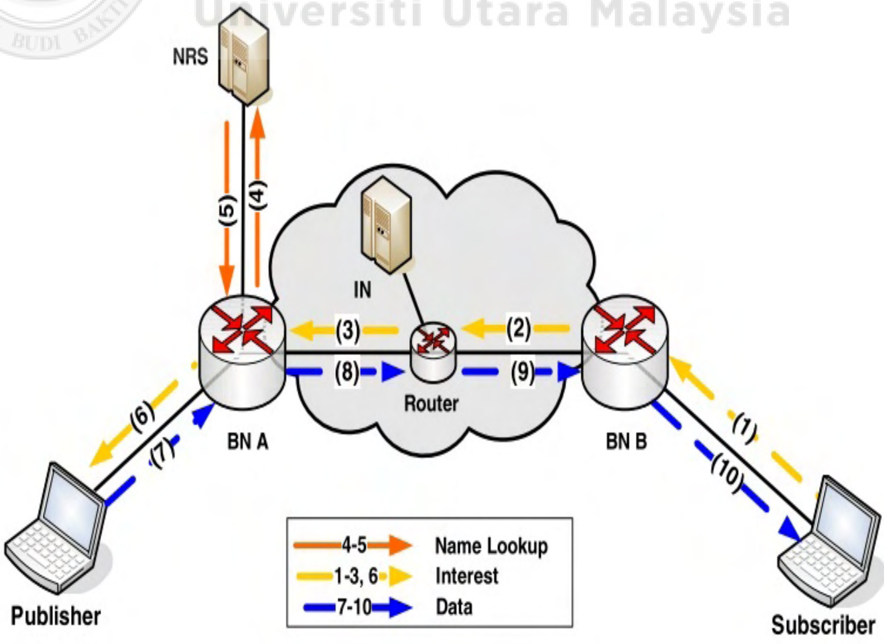


Figure 2.6: CONVERGENCE [1]

2.5.7 MobilityFirst

The MobilityFirst project (see Figure 2.7), funded by the US Future Internet Architecture program, proposes a clean- slate Future Internet architecture with an emphasis on treating mobile devices as first-class citizens. As a result, MobilityFirst provides detailed mechanisms to handle both mobility and wireless links, as well as multicast, multi-homing, in- network caching and security. The basis of the MobilityFirst architecture is the separation of names for all entities attached to the network (including information objects, devices and services) from their network addresses: each entity has a globally unique name, which can be translated into one or more network addresses at various points in the network, thus allowing messages to be dynamically redirected in order to follow a mobile device or content.

2.5.7.1 Naming in MobilityFirst

Each network entity in MobilityFirst is assigned a Globally Unique Identifier (GUID) via a global naming service that translates human-readable names to GUIDs. Every device in MobilityFirst must obtain GUIDs for itself, its information objects, and its services. GUIDs are flat 160-bit strings with no semantic structure and they may be randomly selected, since their length ensures that the probability of a collision is small. Alternatively, GUIDs can be self-certifying hashes of information objects, thus allowing information integrity verification, or hashes of public keys, thus binding devices to principals. Each network attached entity has a unique GUID, and if an entity (e.g., video file) is available in multiple network locations, then all of its copies will have the same GUID. By naming all network entities, MobilityFirst can support both name-based information delivery (via information GUIDs) and host-to-host communication (via device GUIDs).

2.5.7.2 Name Resolution and Data Routing.

In MobilityFirst all communication starts with GUIDs, which are translated to network addresses in one or more steps, via a Global Name Resolution Service (GNRS) as shown in Figure 2.7. A publisher that wishes to make some information available asks the naming service for a GUID and then registers the GUID with its network address in the GNRS (arrow 1). A GUID is mapped via hashing to a set of GNRS server addresses, which are contacted using regular routing. When a subscriber wants to receive some information, it sends a GET message that includes the GUID of the requested object, along with its own GUID for the response, to its local Content Router (CR) (arrow 2).

The CR can only route based on actual network addresses, for example IP addresses, and hence it asks the GNRS for a mapping between the destination GUID and one or more network addresses (arrow 3). The GNRS replies (arrow 4) with a set of network addresses. The CR selects one of these network addresses, adds it to the GET message, which it then forwards using the regular routing tables in the CRs (arrows 5-6 and 9). The GET message includes both the destination GUID and the destination network address, and any CR along the path can consult the GNRS to receive an updated list of network addresses for the destination GUID (arrows 7-8) if, for example, due to mobility the GET message cannot be delivered to the publisher. The publisher sends its response to the subscriber's GUID, using the same procedure (arrows 10-13).

The resulting name resolution and data routing process is a hybrid between IP routing and name-based routing. The actual routing is performed based on network addresses, with the GNRS only used to map GUIDs to network addresses. For less dynamic services, MobilityFirst can translate each GUID to a network address once, as with DNS, and operate based on network addresses only, ignoring the GUID. For more dynamic services, the GUID may be translated multiple times; the first router (optionally, others too) asks the GNRS for the network addresses bound to a given GUID and makes forwarding decisions based on the reply from the GNRS. Forwarding can thus be "fast path", when the GNRS is bypassed, or "slow path", when routers (re-)consult the GNRS in order to obtain an updated list of network addresses. This late binding or re-binding is especially useful for mobile destinations. Note that each message is delivered separately, i.e., the GET message and the information object sent in response to it are individually routed based on their destination GUIDs, therefore name resolution and data routing are decoupled in MobilityFirst.[1].

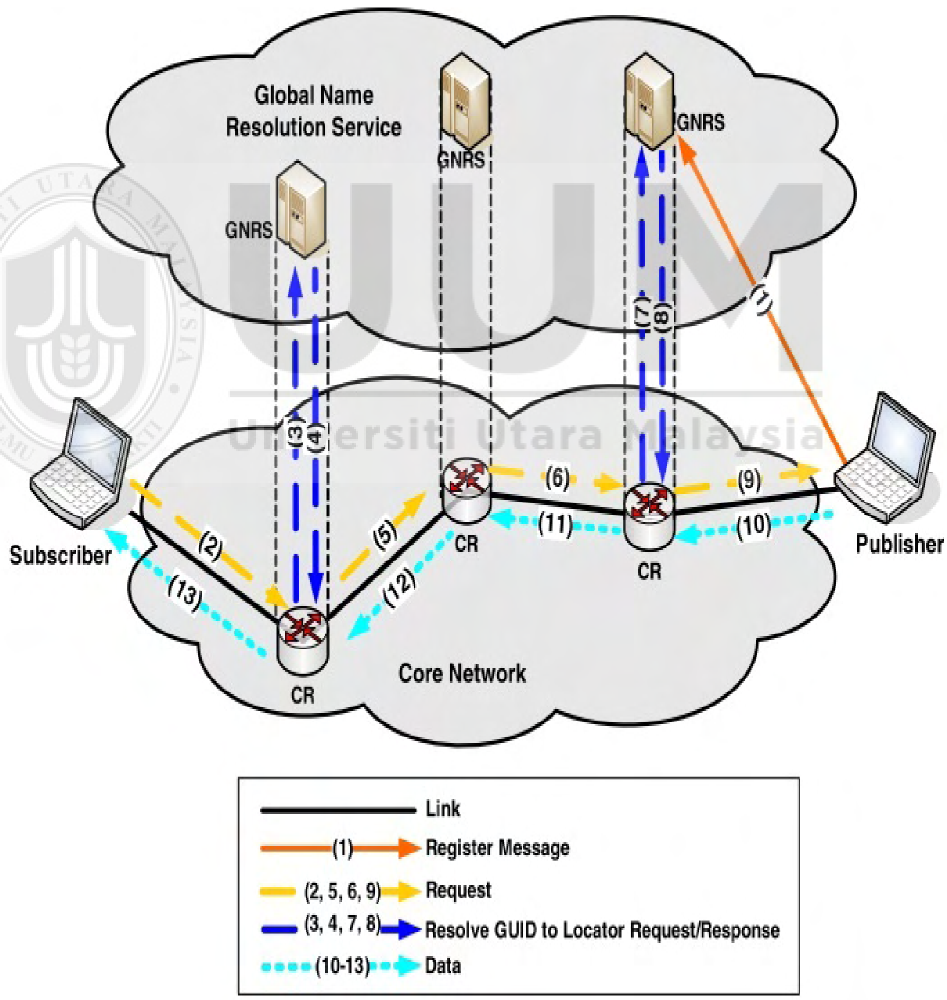


Figure 2.7: MobilityFirst approach [1]

2.6 Comparison of ICN Approaches

Table 2.1 below can summarize the most salient characteristics of each ICN architecture

Approach	Naming	Name Resolution and Data Routing
DONA	Flat, consisting of principal and label part.	Resolution handlers organized according to AS hierarchy. Coupled: during resolution. Decoupled: the network address is returned by the resolution handlers
NDN	Hierarchical, may contain publisher- specific prefix.	Routing protocol used to flood name prefix information. Coupled: during the propagation of the request.
PURSUIT	Flat, consisting of scope and rendezvous part. Scopes may be organized hierarchically	DHT-based rendezvous network compares publications to subscriptions and then matching them. In order to limit resolution, scopes can be used. Decoupled: separating the topology management and forwarding rendezvous.
SAIL	Flat-ish names, consisting of authority and local part. Possible name aggregation for the same authority. Unspecified	Coupled: During NR. Decoupled: NR based on DHT returns content locator. Hybrid: to assist coupled operations, routing hints are returned by the NR.
COMET	Unspecified. Names produced by name resolution system which creates aggregatable names for related content	Coupled: DONA-like resolution but the content router is the place where the the forwarding state is being installed during the resolution. To limit resolution, filtering are being used.
CONVERGENCE	Either flat or hierarchical, most work based on hierarchical	Coupled: During NR , routing state is being accumulated by Interest packet.
MobilityFirst	Flat, consisting of a single component	To map names to network addresses, hash based global NRS is used , Decoupled: requests and data are independently resolved and routed

Table 2.1: Comparison of Approaches

2.7 Source Locator (SL)

ICN makes the information as the main component that its architecture is focusing, and moving the communication concept from being focusing of the node-centirc paradigm which is focusing on the interconnection to an Information-Centric one that focusing of the information itself. Good information distribution need to ensure to use the available data sources to reduce the network latency and load and raise the information availability [26]. In order to support good information distribution and raise the availability, the NRS need to make sure to use the copy that is available, it also need to make sure to keep the data retrieval and the resolution local to enhance the network performance. Another parameters like server load should be also considered to be used for locator selection. if a copy of a local object is available to be retrieved, resolution for this object ID need to be done locally (i.e. with the same sub-network) (called resolution locality) and the resolver need to return the locator of the local object (called content locality) of the two endpoints of communication (i.e. receiver and sender) are in the domain of the same network, the path of resolution need also to be included within that network (called resolution path locality) [26].

The existing research's efforts dealing with the above can be categorized into two. One reduces the number of name entries in the world by aggregating entries, as in CCN and "deepest match". The other provides name resolution service in a distributed manner, especially, adopting an improved form of Distributed Hash Table (DHT) [8]. Both of these two kinds of research efforts reveal the correct directions of implementing a scalable name resolution [27].

In DHT, any involving node can be quickly get data according on keys. Instead of a single server, the mapping responsibility is shared among the nodes, and this is done by a way that changes in which nodes are connected to the system cause minimal disruption. Because of this design, distributed hash tables can extremely support huge nodes numbers, while handling the continuous departures, arrivals and failures. The data stored in the hash table may not be the actual data requested; it may simply be a link to the data. For instance, when using a distributed hash table as part of a search engine, the result returned from the system may be a link to the server hosting the content searched for rather than the content itself [8].

The ICN architectures that are available on ground can be divided into systems with hierarchical namespace for objects, i.e., the object namespace has a hierarchical structure (e.g., CCN), and systems with a flat object namespace (e.g., NetInf, DONA, PSIRP).

For flat names, the approach of systems based on Distributed Hash Tables (DHT) is promising. DHTs are highly scalable, decentralized and mostly self-organized, this limit the need for administrative entities. Many compact routing protocols are available, they are usually used for P2P overlay network, which can route the messages in O(LogN) routing steps, with compact routing tables of O(LogN) state, were N refer to the number of nodes. O(LogN) resolution steps may, however, result in unacceptably large latency. Recently proposed, many promising approaches guarantees a constant number of hops per lookup. however an issue related to the convergance time still exist, approaches can decrease the number of required hops in exchange [28].

Globally, however, an NR system that based on DHT is becoming more problematic. There is a problem in binding control and placement because of the flat namespace and the substantial unfriendly nature of the Autonomous systems (ASs) and other administrative domains. Scalability and the increment in churn is also need to be taken into consideration. In general, flat namespace have many advantages over the hierarchical one, a possible solution to the problem can be done by the integrating of new promising ideas from the field of flat name routing identifiers in the NR system, like the Late Locator Construction (LLC). It is one of the main obstacles of the NR design to find a compromise between the scalability (aggregation based on hierarchical names) and the flat namespace name persistence feature [28].

NR systems like the ordinary Distributed Hash Table (DHT) and Domain Name System (DNS) do not fulfill the new needed. So a researcher proposed a distributed NRS with embedded name-based routing capability called Multi-level Distributed Hash Table (MDHT). The MDHT system have the provision of a nested, hierarchical DHT architecture for scalable distributed name resolution and effective anycast flat-IDs routing. If the IDs have some additional structure like owner/publisher information. The MDHT system can make used of this structure to do name aggregation on the global level for simple global deployment and additional improve to the scalability. The hierarchical architecture can show the current nature of the existing network topology, topologically emerging the NRS for effective data dissemination in the Internet. The MDHT system is usable for any system that needs a distributed dictionary in general or specifically an NRS, just like many ICN approaches [2].

There are two main primitives provided by MDHT interface , the first one is PUT(ID, meta-data) which is needed to register bindings in the directory of the network, each ID is public and bound to a set of locators or meta-data . The second one is GET(ID) that can return a list of locator(s) where the data of the object can be got back or the object itself. To make a new object known and accessible for other users, it has to be registered (via PUT) in the MDHT system (Figure 2.8). The two types of bindings can be registered in Dictionary Records (D-RECs), indirection bindings, which map IDs to other IDs and location bindings, which map object IDs to network locations. First, the user's device ID TK is registered in the MDHT system. On the user's AN, TK is mapped to its local address k, which can be private. Thereafter, the data object can be registered in the user's AN to make it accessible for local users connected to the same AN.

A new D-REC is created by registering a new object to store an indirection binding which maps the object ID X into the user device ID Tk. The registration request then, propagated up the MDHT tree so that a new binding for object X is recorded in the upper DHT areas along the path from leaf (AN) to root (AS DHT). The hash(ID) is used as DHT key On all levels except the top level REX to store the binding. The used of the hash function depends on the implemented DHT system [2].

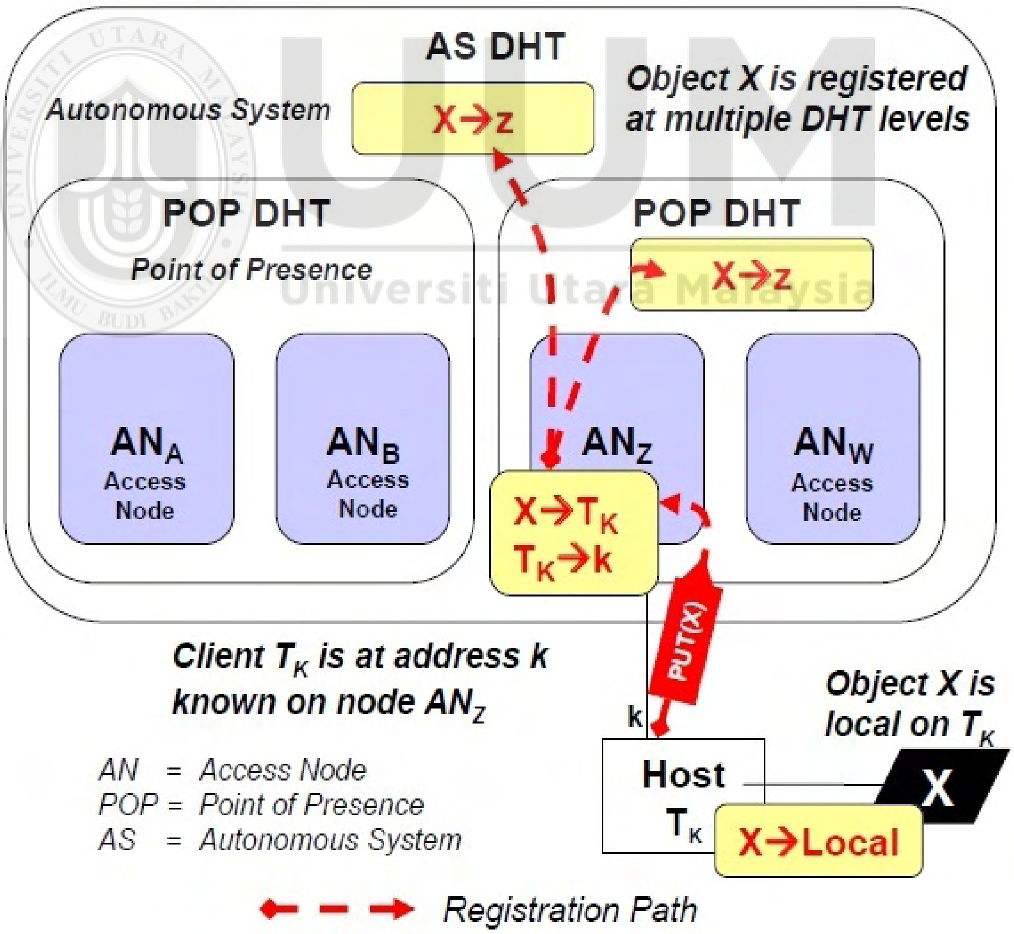


Figure 2.8: Register of object in MDHT[2]

2.8 Point of failure types

As presented in [29] and [30]Failures fall into two obvious categories: hardware and software. Hardware failures were a dominant concern until the late 80's, but since then internal hardware reliability has improved enormously. Decreased heat production and power consumption of smaller circuits, reduction of off-chip connections and wiring, and high-quality manufacturing techniques have all played a positive role in improving hardware reliability. Today, problems are most often associated with connections and mechanical devices, i.e., network failures and drive failures. Software failures are a significant issue in distributed systems. Even with rigorous testing, software bugs account for a substantial fraction of unplanned downtime (estimated at 25-35%).

- Fail-stop: A halting failure with some kind of notification to other components. A network file server telling its clients it is about to go down is a fail-stop.
- Omission failures: Failure to send/receive messages primarily due to lack of buffering space, which causes a message to be discarded with no notification to either the sender or receiver. This can happen when routers become overloaded.
- Network failures: A network link breaks.
- Network partition failure: A network fragments into two or more disjoint sub-networks within which messages can be sent, but between which messages are lost. This can occur due to a network failure.
- Timing failures: A temporal property of the system is violated. For example, clocks on different computers which are used to coordinate processes are not synchronized; when a message is delayed longer than a threshold period, etc.
- Byzantine failures: This captures several types of faulty behaviors including data corruption or loss, failures caused by malicious programs.

2.9 Summary

This chapter introduced an overview on the ICN and the main reason for this new technology. Furthermore, it specified two concepts in ICN which are (Naming) and (Name Resolution and Data Routing).

There are many approaches in ICN and each one has its own way in naming the data objects (NDO) and how to retrieve the data object (DO) between nodes. The general studying of these approaches will help to have an overview on how naming resolution system works, what issues it has, and how these approaches are overcoming these issues .

A brief explanation about what is the source locator and what is distributed source locator was given with highlighting some of its types (e.g. DHT and MDHT).

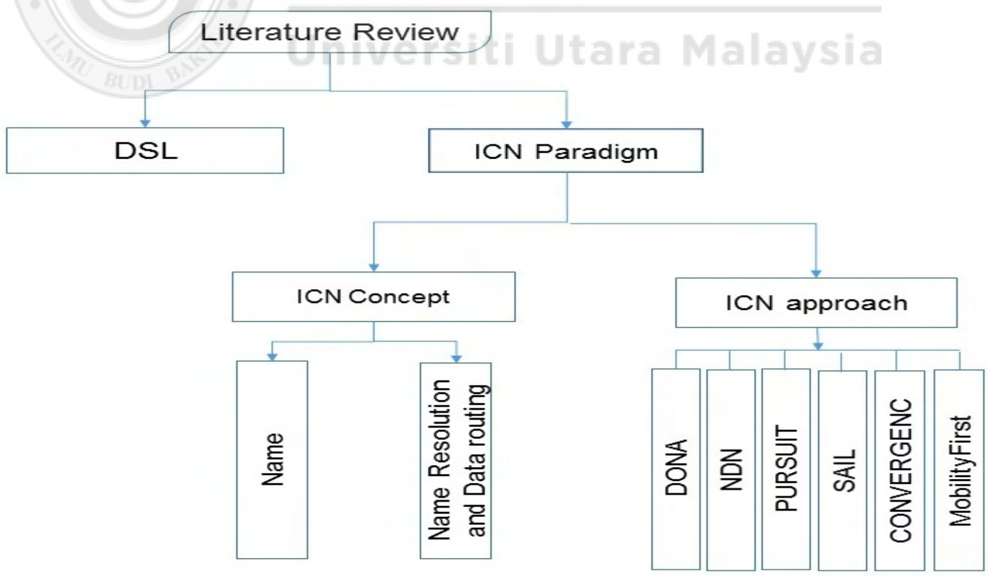


Figure 2.9: literature Review

CHAPTER THREE
RESEARCH METHODOLOGY

This thesis aims to propose a conceptual Distributed Source Locator (DSL) for Naming Resolution System (NRS) in Named Data Network NDN approach by depending on/and developing previous studies from the other ICN approaches. This NRS is promising technique that guaranties the founding of any NDO available in the network in a fast way compared with the name-based routing technique currently adopted by NDN approach. As earlier mentioned, this will be done by studying the NRSs mechanisms of the other ICN approaches, and identifying any issues accompanying their implementations with these approaches. In order to achieve these objectives, it requires a rigorous methodology to follow, and this is the focus point of this chapter. For this purpose, this research employs the Design Research Methodology (DRM) and introduces its main stages according to the phenomena of this research. The chapter starts with the overall research approach as shown in Section 3.1. Section 3.2 introduces the first stage of DRM known as Research Clarification (RC). It discusses the aims of RC stage, methods to support this stage, and main deliverables. Section 3.3 describes the second stage called Descriptive Study-I (DS-I). It discusses steps needed to obtain sufficient understanding of the current situation, designs a reference model, and proposes a conceptual model. Section 3.4 highlights the methods adopted in designing the proposed distributed source locator mechanisms for NRS in NDN , named Prescriptive Study (PS). Methods for performance evaluation and its metrics are discussed in Section 3.5 toward the end of the chapter, where a chapter summary rounds of this chapter.

3.1 Research Approach

The primary aim of this research is to propose distributed source locator mechanisms for NRS in NDN by considering the naming schema that the NDN project currently uses. This demands a careful mapping between naming-based routing process in finding NDO to develop a new one leading to an effective and efficient solution [2]. These requirements are fitted with the design research definition as proposed by Blessing in [31][32], where design research integrates the development of understanding and the development of mechanism. In the way to achieve the goal of this research it demands to have a full view on how the NRS works and what is the problems it faces. This will lead us to find the proper distributed source locator in order to avoid these problems and make it compatible with NDN. Figure 3.1 illustrates the (DRM) framework where the links between DRM stages, the methods used in each stage, and the main deliverables are shown. The light arrows between the stages illustrate the main process flow, while the bold arrows to each stage demonstrate methods used and deliverables of the particular stage.

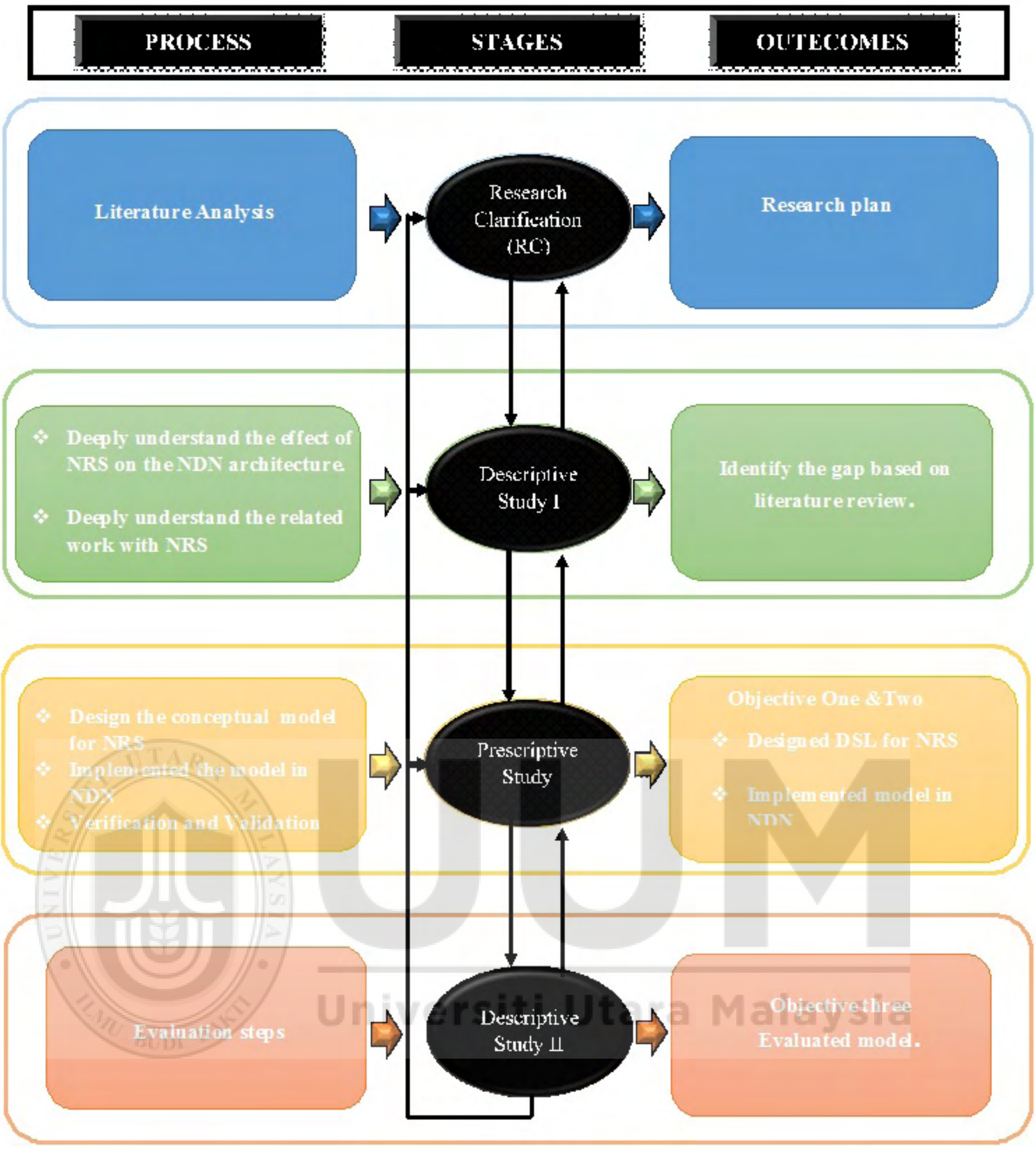


Figure 3.1: Research Approach

3.2 Research Clarification (RC)

RC is the first stage of DRM, in this stage the paradigm of ICN will be steadied, describes and reviews the NRS in the literature, as well as a NDN architecture. This will help to putting the whole research plan . RC has six iterative steps, shown in figure 3.2.

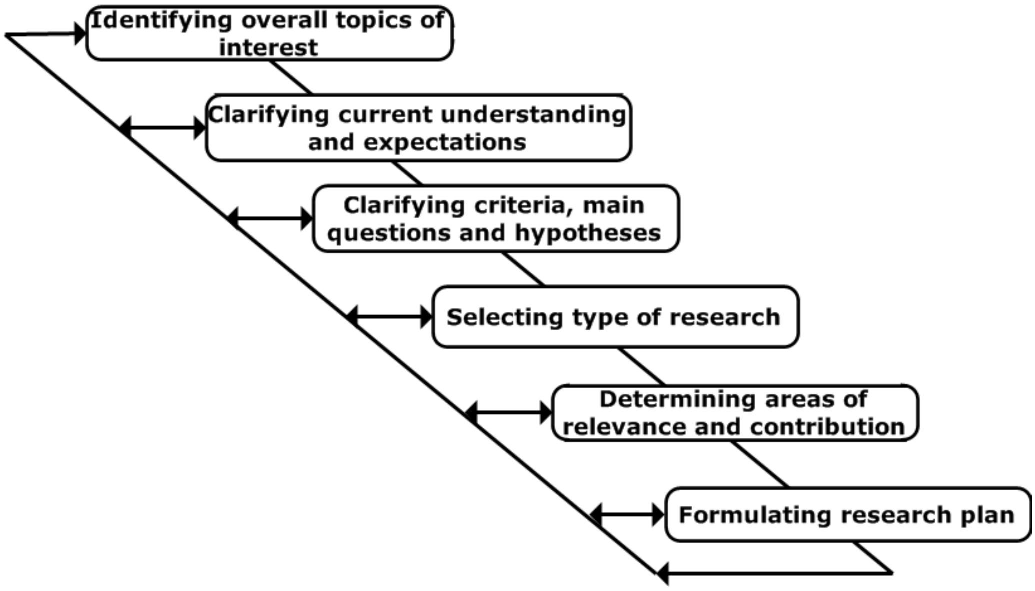


Figure 3.2: Main Steps in the Research Clarification Stage

The outcomes of the RC stage are:

- The research motivation and the research focus.
- The research problem along with research questions in conjugation with research objectives.
- The research scope, type of research and research methods.
- The expected research contribution and deliverable.

3.3 Descriptive Study-I (DS-I)

Upon the completion of the RC stage which presents the overall research plan, the research moves to the second stage, known as DS-I as showing in Figure 3.3. The DS-I is used to gain deep understanding of the current situation and it involves a critical review of the literature in the research area as well as empirical studies. During the course of this stage, details on what is the NRS will be reviewed and what are the most issues that face it based on actions have been taken by previous approaches in ICN.

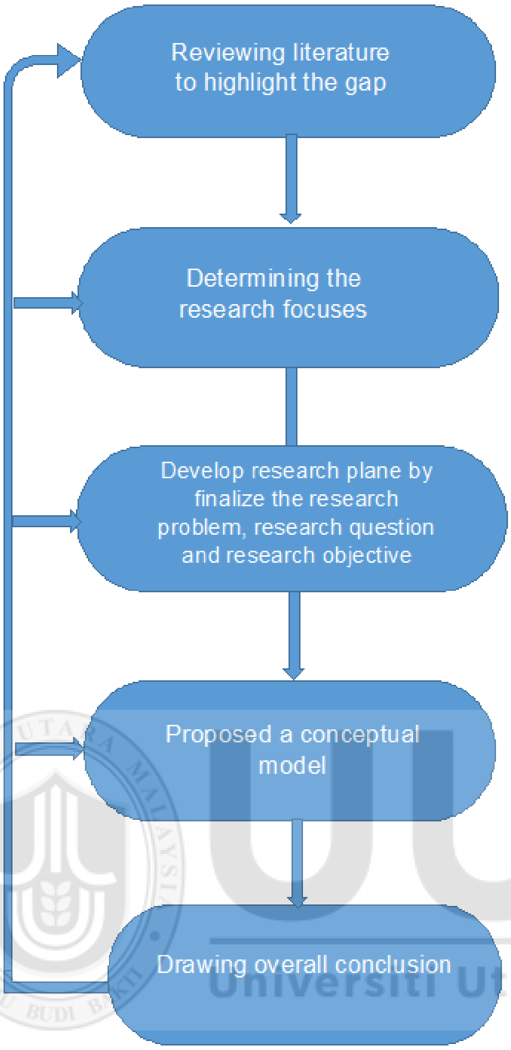


Figure 3.3: Main Steps in the Descriptive Study-I

.

3.3.1 Conceptual Model

To improve the NDN performance in routing data with the same namespace, there is a need to add a name resolution system to guarantee finding any NDO in the network. The

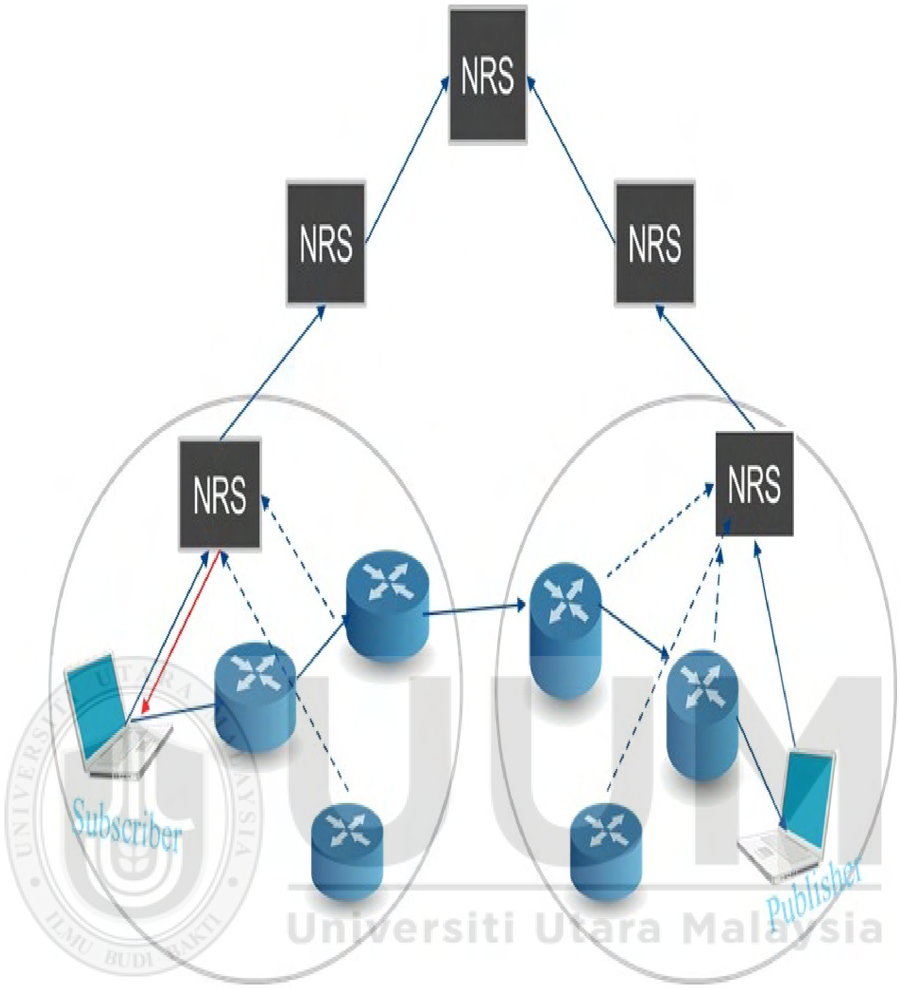


Figure 3.4: Conceptual Model

problem with current NRS lead us to look up for an efficient distributed source locator for NRS. A proposal conceptual model is designed to clarify the expected improved situation by proposing Interest mechanism.

3.4 Prescriptive Study (PS)

PS is the main stage in DRM, as it includes the design of the proposed mechanisms. For the purpose of this research, Figure 3.5 presents a block diagram describes the main process of PS according to this research marvels. The specifications of the proposed conceptual model are described in the first block. Second block constitute conceptual/analytical modeling of this research, it is built to describe the expected preferred and improved situation using the proposed DSL for NRS in NDN. The third and fourth blocks represent the design and implementation of the proposed mechanism in the simulator environment, The NDN simulator based on NS3 will be used [33].

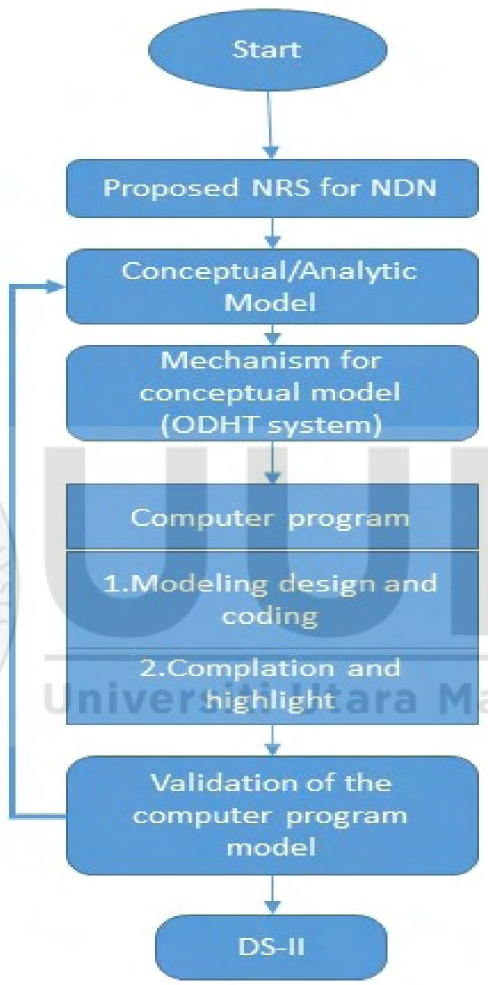


Figure 3.5: Prescriptive Study Steps

Validation of the computer program model is illustrated in the fifth block if everything is passes correctly, then proceed with DS-II stage.

3.4.1 Verification and Validation (V&V)

Model verification is the confirmation that the model can be changed from one form into another, as required, with the needed accuracy. Model verification works with constructing the DSL for NRS in the right way to make it immune from previous NRS issues and adding it to the NDN architecture. The accuracy of changing a problem formulation into a designed model specs or the accuracy of changing a model representation from a flowchart into an executable computer program is evaluated in model verification[34]. In the other hand Validation needs to be performed to ensure that the mechanism meets its intended requirements in terms of the methods employed and the results obtained, which is all part of the process to build the correct model.,As stated in Chapter One, this research works with constructing the right model of DSL for NRS and implemented this model in NDN networks, as such the research has proposed a DSL for NRS this will done by using simulation to verify and validate the results. Then, the proposed model was used to implemented in NDN which was also verified and validated by. Comparing the actual simulation results of the current NDN architecture with the designed model result and comparing the analytic model result with simulation model result. On the other hand, the research had focused on, An activity of accuracy assessment can be identified as verification or validation based architecture, by compared the model behavior with respect to the previous system behavior in the same simulation environment[35]

3.5 Descriptive Study-II (DS-II)

DS-II focuses on the model and evaluation of the designed mechanisms. This will achieve the third objective of this research. Performance evaluation is crucial step in the evaluation of any research[31]. In this research we will evaluate the effective of our conceptual DSL model for NR in NDN architecture by using simulator environment, by compare its functionality and performance in forwarding strategy with the current architecture performance for the studies in the same extant.

3.5.1 Evaluation Methodology

Particularly, ICN seriously isn't yet clear what is eligible as a ”realistic workload”. Therefore, more works are needed to define property's workloads for ICN evaluation studies. According to that, the experimental practices itself in addition to the evaluation methodology are now being actively investigated ICN architectures. There are many factors that can effect on the experimental results such as: network condition (e.g., Available link capacity); topology selected; link delay, node mobility, background traffic load, loss-rate characteristics, and other aspects (e.g., The variety of devices used).

3.5.2 Evaluation Techniques

The evaluation technique can by divided in to three different technique that evaluated the communication system which is analytical modeling, measurement and simulation.

3.5.2.1 Analytical modeling

It is an explanation of the behavior of the real system and association between variables of the network system developed in a mathematical structure.These variables may be computable amounts, such as weight, size, length, bit rate, and information flow. But the mathematical method is not appropriate for analyzing a complex system as it requires an enormous arrangement for generalization and supposition. there are two major drawbacks of evaluating performance of the network system using mathematical modeling. First, it creates a lot of hypotheses and simplifications that can disturb the model. Second, it does not take the interaction into consideration, which has a significant feature that may cause the functionality of the network system.

3.5.2.2 Measurement

Measurement is the second model used for evaluating the performance of the network system that requires an actual or prototype of the protocol. However, a prototype of the protocol may be designed and tested on real network or test-bed, but this designing and testing on real network or even in test-bed requires some programs and devices. The output of this prototype on actual programming tolls demonstrates the functionality of the protocol in the actual network. Therefore, it may be the standard approach for the performance evaluation of the protocol if there are ample network resources for the provision of fundamental constraints of the performance evaluation.

3.5.2.3 Simulation

The third model is called simulation, which is the process of planning and developing a model for real or proposed protocol to have a well-defined representation of model behavior for a provided set of conditions. Furthermore, it is a method for analyzing numerous models that provides an easy way for the performance prediction or comparison of several systems. Moreover, the researchers can select the simulation over measurement even if a measurement system is available, because it allows different models for comparison with workloads and a number of network structures

3.5.3 Named Data Network Simulation (ndnSIM)

The ndnSIM which is plugged into the NS-3 simulator (which can also be used for the implementation of many CCN applications, for, e.g the implementation of CCN routing, caching and forwarding schemes), is an open source simulator , the code of both NS-3 and ndnSIM is freely available for the research community and students to be used as the foundation for the implementation of the ICN protocols. The ndnSIM implemented as a new network layer protocol model. The ndnSIM can run on top of any available link layer protocol model, as well as on top of network layer (IPv4, IPv6) and transport-layer (TCP, UDP) protocols. The ndnSIM flexibility enable it to simulate different scenarios (homogeneous and heterogeneous deployment scenarios). The ndnSIM using C++ programming language. This modular structure allows any component to be easily modified or replaced with no or minimal impact on other components. In addition, the simulator provides an extensive collection of interfaces and helpers to perform detailed tracing behavior of every component and NDN traffic flow [33].

3.5.4 Topology Selection

Davies et al.,[36] mentions "there is no single topology that can be used to easily evaluate all aspects of the ICN paradigm". According to that, in this research, this scenario will be used which consisted of two subscriber, one publisher and six NDN router nodes. NDN router will divided in two Autonomous System (AS) area, each one has 3 nodes and one NRS.

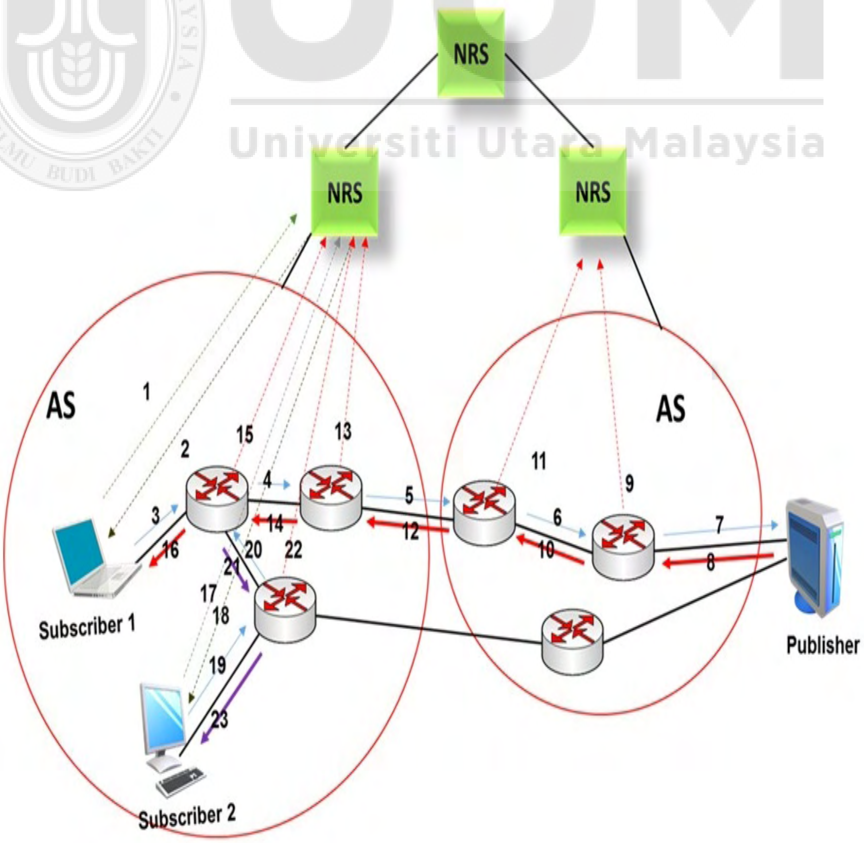


Figure 3.6: System Topology

3.6 Evaluation Metrics

The goal of this project is evaluating the performance of distributing content in NDN context. The metrics like, memory usage , Amount of traffic(number of packet) and link failure metrics that could be used to measure the performance of the proposed objectives of this research. It should be selected to reflect the quality of services provided by the studied (specific) system. The essential functionality provided by DSL for NRS in NDN is distributing content to the end users.

a) Amount of traffic: it is the size of packet that each node receive it.

b) Memory usage: it is point to the amount of memory that the content storing process consumes.

c) Delay time with link failure : is to explain how each forwarding strategy deal with any failure occur in the way of transfer the data

3.7 Summary

This chapter discusses in detail the research approach to ensuring that the research objectives can be achieved. This research concentrates on the proposed conceptual DSL for NRS and the implementation of the proposed model for NDN. Four main activities of the research are outlined in this chapter in line with DRM. The first activity is the Research Clarification (RC) stage, which discusses methods to support the initial stage of this research. The aims of RC are to identify and refine a research problem, objectives, and research questions that are both academically and practically worthwhile and realistic. The second activity is called Descriptive Study-I (DS-I), which discusses steps to obtain sufficient understanding of the current situation, designs a reference model, and proposes a conceptual model. The third activity highlights the methods adopted in designing and implement the DSL for NRS in NDN architecture, named Prescriptive Study (PS). The last activity named DS-II focuses on the evaluation of the designed model.

CHAPTER FOUR
SIMULATION EXPERIMENT, RESULTS AND DISCUSSIONS

4.1 Introduction

After establishing the research methodology mentioned in Chapter Three as a guideline to achieve the objectives of this research, this chapter proposes the idea of a new structure to distributed the NRS node based on distributed Hash table (DHT) algorithm, we call it Overly Distributed Hash Table (ODHT) . The proposed mechanism focuses on distributing the name resolution into a number of nodes connecting together using point to point and number of levels to efficiently locate the node that stores a particular data item, to avoid point of failure and improve the lookup mechanism , in each node.

Two things must be taken into consideration of this design, the first one is the naming scheme that is being used in the NDN architecture, as the aggregation naming currently used by NDN, put many challenges like the length of the packet, the bit error rate and other considerations, which make the normal DHT not effective if used in our design. Secondly, the huge number of data that is being uploaded to the Internet may cause a scalability problems when it comes to the memory size needed to store the names of all of the stored data.

Section 4.2 is an introduction to the ODHT idea and its representation of the basic concepts for our design and goes deeply in representing the Overly Distributed Hash Table (ODHT) model, and how to improve its functionality based on some equations. This conceptual model has been implemented in NDN architecture, the step of the implementation and the work has been shown in Section 4.3. In order to show the effectiveness of the new idea in the NDN architecture, it has been evaluated based on the way illustrated in Section 4.4. Finally, Section 4.5 concluded the overall concept of this chapter in a brief summary.

4.2 Distributed Source Locator (DSL) for NRS in NDN

In NDN, NRS has to manage the global data mapping based on hierarchical IDs efficiently and with low latency. It must return a list of locator(s) for the requested objects. The main goal for our distributed source locator for name resolution is to get a decentralized system which will make sure that the failure in one node will not affect the whole system. According to the huge number of data and the aggregation design for name in NDN, the normal DHT will not be a good solution for distributing the prefix message in name data network since the prefix name in NDN is not fixed. Due to this, our goal is to design a conceptual model that is decentralized, scalability with low memory usage, and have a flexible naming. Thus, to achieve these points, we suggest Overly Distributed Hash Table architecture design to improve our based on Chord structures which is set its identifiers in a circle.

4.2.1 Theoretical Analysis

In Computer science field, Chord is a peer-to-peer distributed hash table algorithm and protocol. The key-value pairs are being stored in the distributed hash table by assigning keys to various nodes. A node stores the keys values of all the keys which it is relying under its responsibility. Chord identify how keys are assigned to nodes, and how a node can gain the value of a provided key by initially locate the node responsible for that key.

The Chord software takes the form of a library to be linked with the client and server applications that use it. The applications within the client and server are dealing with Chord in two main ways. First, Chord delivers a key lookup algorithm that fetch the node's address responsible for the key. Second, at each node, the Chord software keep monitoring the application of changes in the keys' set under the responsibility of a node. This make it possible for the application software to move the matching values towards their new location when the node joins. The application that is being used by Chord have the responsibility of giving any needed authentication, replication, caching and user-friendly naming of data. These feature are being simplified by the Chord's at key space. The design of peer-to-peer systems and applications based on the current Internet can be simplified by using Chord by providing the below features[37]:

Load balance: Chords is considered to be a distributed hash function, distributing keys equally between the nodes; this gives a level of ordinary load balance.

Decentralization: Chord is fully distributed, all nodes have the same importance. This enhances the system's robustness and makes Chord suitable for loosely-organized peer-to-peer applications

Scalability: Chord lookup cost have direct relationship with log of the nodes number.

Availability: Tables are automatically adjusting themselves when new nodes are joined the network or leave the network due any type of failure, to ensure that the major failure in the underlying network is barred, so the node responsible for a key is always found. This is achievable even if the system is continuously changing its state.

Flexible naming: Chords put no restrictions on the keys structure that it looks up: The key-space of the Chord is flat. This provides high flexibility for the application on how to map their own names to Chord Keys.

Chord keys are stored at their related Chord nodes after being mapped. Every node has a predecessor and a successor. the predecessor is next node of a specific node in the identifier circle in the a counter-clockwise order, while the successor is the next node in the clockwise order. If a node with ID 0, then its successor will have 1 while the predecessor of this node will be node 2^m-1

Each Chord node is responsible for the keys of its successor nodes, it also stores the list of the successor lists of its own successor. From the other hand, it also stores the IDs of its neighbors nodes which is called fingers. These IDs are stored in a special table called Finger Table (FT). Therefore, in Chord, a lookup for a key terminates at the key's predecessor.

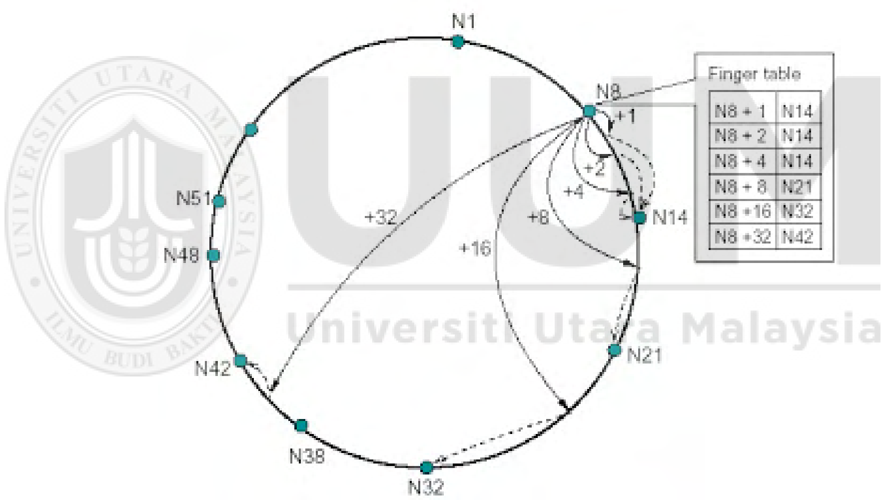


Figure 4.1: Chord structure

Chord designed is very suitable model that satisfies our requirement in having a decentralizes system to distribute the source locator and avoiding the single point of failure. However, when huge amount of data is being exchanged in a Chord-like model, huge numbers of nodes are needed to be connected together in a flat structure, and each one of these nodes must handle a big number of nodes' IDs. Due to all of this, it is expected that high memory usage will take place in each node. A proper modification in the Chord structure is needed to overcome this problem and make it capable of handling such huge amount of data in the NDN architecture.

4.2.2 Overly Distributed Hash Table (ODHT) model

In this section, the model of ODHT has been decelerated to improve the need of having an efficient distributed source locator . This model is built based on the DHTs model [37] and MDHT: A Hierarchical Name Resolution Service for Information-centric Network [2] aims to understand the impact of the overly design on memory usage and efficient lookup for specific data in this design . Our design start by separate the network to a number of levels. Each level have a number of zones (e.g. each ISP is a zone). The user is free to upload his data to any level, for example, if the data is intended to be distributed between the nodes of the same zone, it should initially uploaded to the first level (level number from is up side down and starts from the nearest part to the user).

Each ISP have a number of Content Routers (CR) to cache the data. The routers of the same zone are connecting together in a small DHT ring topology to distribute the data of its node. The content router is using the hash function to store the data packet, this packet consists of the the name of the data object, the ID of the node that stores the data, and the scope (which is the number of the levels that the data are needed to be distributed). Each node will pass an image of its content to the other nodes in the same ring, so if one node fails, the other nodes will immediately covers its absence. At this point, an update on the FT of all the nodes is also taken place by removing the failed node.

In ODHT, if the key that is being looked up is in the same ISP, then the number of hops that is needed to be looked up is equal to O log N. Otherwise, when the intended key is not in the same ISP, then the number of hops that being looked up at specific level Li will be according to the below equation.

$$h = Olog(N)L_1 + Olog(N)L_2 + ... + Olog(N)L_i \quad (1)$$

(Li the level where the key is found) The average number of overlay hops within an ODHT resolution domain at level L depends on the underlying DHT technology, which can be independently chosen by each provider.

The distance between two nodes in the metric space is defined as their distance along the ring in clockwise direction. Each node has two different types of connections (according to chord design) to other nodes: short-range connections to a fixed number of closest nodes (in clockwise direction) and long-range connections to some distant nodes. These nodes are called short-range and long-range peers of the node, respectively. The distance of the node and its farthest short-range peer is denoted by dS. Routing is assumed to be greedy: a node forwards a lookup request to its peer being the closest to the target node in the metric space of the DHT (without overshooting it). This greedy routing process can be described by the following pseudo-code algorithm[38]:

Algorithm 4.1 Greedy overlay routing

```
while node =target do
    proxy ⟵ Get ClosestPeer(node,target);
    if Distance(proxy,target) < Distance(node,target) then
        node ⟵ proxy;
    else
        error
    end
end
```

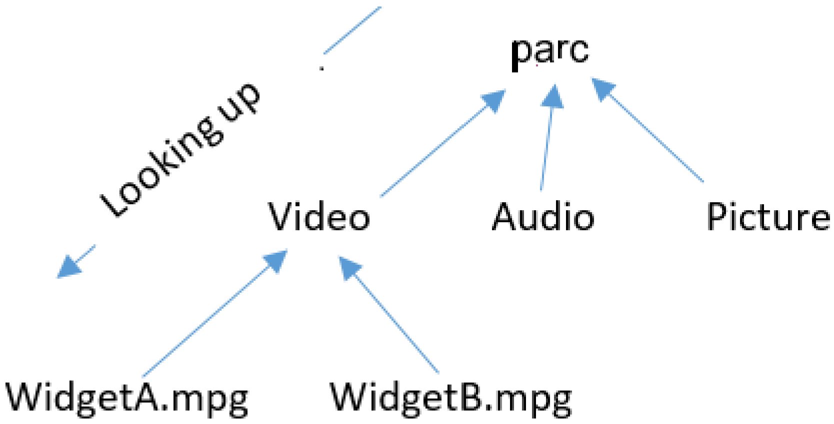
Each peer is assigned a (unique) id corresponding to a location on a virtual ring. The host list of each peer stores peers with IDs close to a certain IDs when a peer wishes to propagate information (such as a job announcement). Every peer maintains an incomplete host list of other peers in the network. The host list saved in a table called finger tables in Chord [37] and contains entries (e.g. addresses) on some other peers together with time stamps of the last successful contact to that peer. Additionally, every node ni is associated with an id value id(ni) □ [0.....id Max]. For any two nodes nx and ny, the distance in the circular id space can be defined as in the following equation[39]:

$$Dist(nx,ny)=\begin{cases} id(nx)-id(ny), & id(nx)<id(ny) \\ id(ny)+(idMax-id(nx)), & id(nx)>id(ny) \\ 0, & Otherwise \end{cases} \quad (2)$$

4.2.3 Evaluation of ODHT Model

4.2.3.1 Availability

By arrange the data in a hierarchical shape, the looking for specific data will be easier than arranging them in the flat shape. For example if we looking for /parc/videos/WidgetA.mpg instead of looking for hole the prefix on time with the ODHT, we will start looking for /parc/ then in parc under the level where parc found, we will looking for video, and under videos level, we will find WidgetA.mpg. The looking for the data will be based on o(n) algorithm



4.2.3.2 Memory usage

To avoid the linear search, Chord deployed a faster search way by asking each node to store a finger table that have up to m entries. The i^{th} entry of node n will contain successor((n+2^{i-1}mod 2^m). The first entry of finger table is in fact the node's immediate successor (and therefore an additional successor field is not needed). When a node need to look up a key k, it will pass the query to the nearest successor or predecessor (based on the finger table) of k in its finger table (the "largest" one on the circle whose ID is smaller than k), until a node finds out the key is stored in its immediate successor.

With such a finger table, the number of nodes that must be contacted to find a successor in an N-node network is O(log N), by divided the number of node to a small rings of node this will help each node to reduce the finger table. According to chord-like DHT, if we considered that a DHT have an N node, each node will have a unique ID ranging from 0 to N-1 and each node will build a finger table containing $2^i$where (i) is the binary representation for the number of bit needed to define a unique name for each node .

For example if we have 16 node in a cluster, in order to represent these nodes in the finger table, the table need to consist of 4 Bits X 16 rows starting from 0 till 15 and as below:

Node Number	1st bit	2nd Bit	3rd Bit	4th Bit
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Table 4.1: Bit arrangement example

Reducing the number of bits in representing the a successor node will help in reducing the bit error which may lead to the loss of the node by taking the wrong name, so the node error rate will depend on the bit error rate if the last one reduced, the first one will reduce . The probability of error bit can be x/i where x is the number of the error bit. The probability of error bit will increased when the number of bits is increased, and the number of nodes that may be effected by this error is equal to O (N), since each node can connecting to O log N node, this error will make the number of node unreachable to be increased even if they are exist in the ring. Chord have a fast distributed computation of a hash function mapping keys to nodes responsible for them. It uses consistent hashing , which has several desirable properties. With high probability, the hash function balances load (all nodes receive roughly the same number of keys). Also with high probability, when an Nth node joins (or leaves) the network, only an O(1=N) fraction of the keys are moved to a different location, this is clearly the minimum necessary to maintain a balanced load.

4.3 Implementation the ODHT in NDN architecture

The main objective of this research is to apply the feature of the DNS in the NDN architecture design and allow its use to solve some challenges in the NDN network. Since one of the most effective components in the current Internet architecture is maybe the DNS , although it is used to resolves the application-layer domain names to the network-layer host addresses, it is also used in many other solutions for the name-data mapping, including mail server management, DNSSEC key management, and many other types of mappings. All these facilities given the great success to the DNS, which is mainly the highest distributed and very well scaled system as online database, including the need for careful management for the namespace, all of this motivate the idea behind the creation on NDNS. NDNS which is the NRS for NDN consists of three major components: namespace, name servers, and resolvers. The namespace which is used and controlled by the NDNS system is a subset of the general NDN. Managing data by using a hierarchical namespace and namespace zone location provides a simple and a good distributed management, as well as ensures virtually limitless database storage.

4.3.1 Implementation and Results

In our model, the data up-loader is free to choose in which level the data should be distributed according to the needed security level and with the less possible congestion (Figure 4.2), also it will effectively enhance the lookup mechanism since it start looking for prefix from the end to the beginning using LPM (longest prefix match) algorithm. Naming is the most important part of any application in NDN data-centric. According to the fact that NDN directly operates on names application, so in order to guarantees the correct application functionality and get the all benefits of the NDN architecture, the developers which is work on application must design the data naming model with more attention.

To avoid problem with namespace, the designed model in this research provides the user with the capability of choosing the name for the data only, the rest parts of the name will completed aggregatable in each level to take the name of the cached node (Figure 4.3).

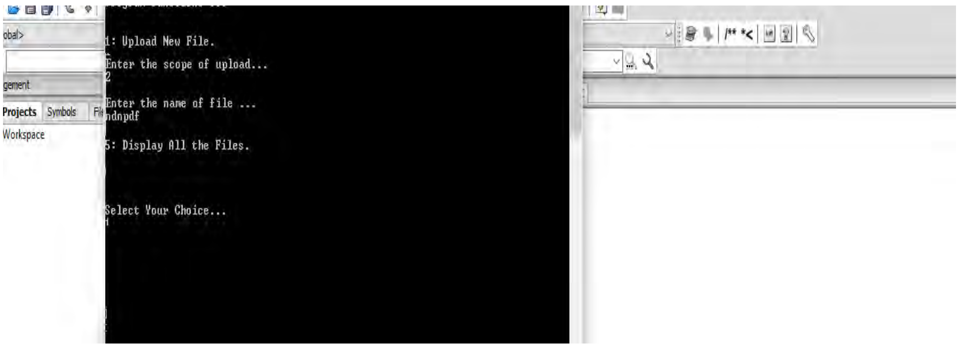


Figure 4.2: Data level

Figure 4.3: Aggregation name

The design of NDNS provides two types of the resolvers: caching resolver and stub resolver . Caching resolver are applications working from dedicated servers inside the network of the ISP and they are responsible for executing iterative queries in response to the incoming recursive queries from the stub resolvers. On the other hand, stub resolvers are simple applications that usually embedded inside software that perform NDNS queries upon software requests. When the user looking for specific data, he will send his interest packet to the ISP which is connected to. The CR. in that ISP will look for the location of the data in its hash table, if the router finds it, then it will send back the location to the requester, otherwise it will send it to the CR in the upper layer which is the ISP connected to and so on until getting the needed data (the location of the data packet). once the data is received, it will be sent back to the router or the user by the normal way in current NDN architecture, with the note that at this stage, all nodes in NDN will use the best-route strategy instead of flooding strategy. By applying this conceptual model for name data network, the whole nodes will use the best-route strategy instead of the flooding strategy for data looking since it is already know where to find the data packet, and it is not like the current DNS since the data location in NDN is changed based on nearest router request the data so the ODHT need to update its table each time that new events happen. (Figure 4.4) below shows the comparison between the forwarding processes currently used by an NDN Node and our new conceptual design which implements the NRS in NDN architecture.

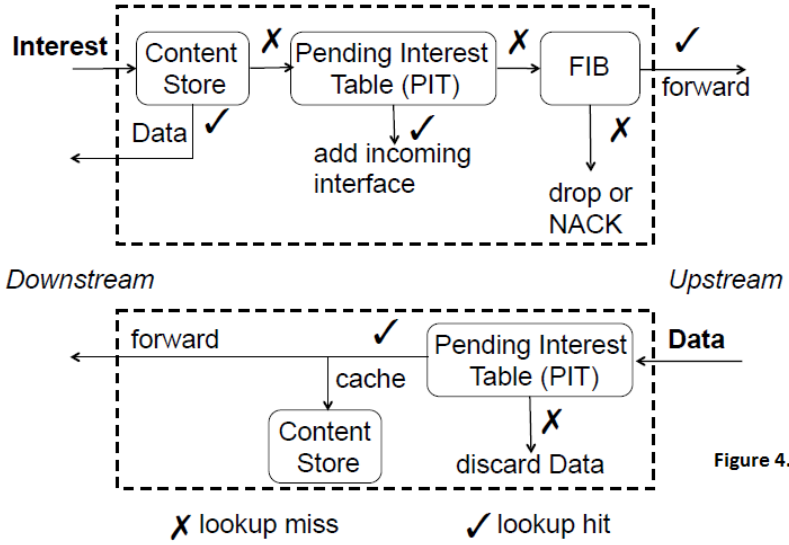


Figure 4.4 - a -

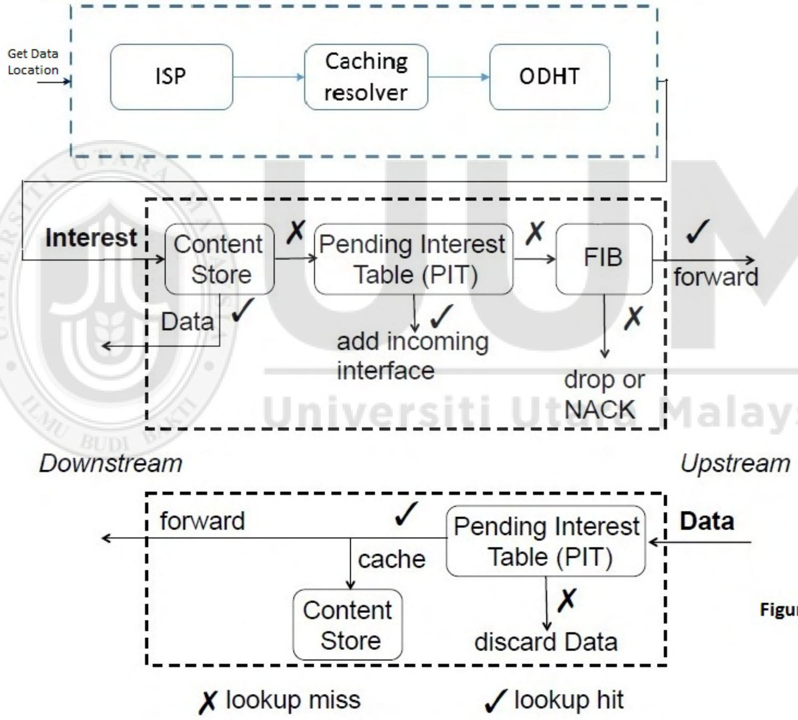


Figure 4.4 - b -

Figure 4.4: (a)Forwarding Process at an NDN Node (b)Conceptual model for NRS in NDN

4.4 Performance of NDN forwarding strategy.

To evaluate the work based on the metrics mentioned in chapter three (traffic amount, delay time and memory usage), ndnSIM simulator was chosen this purpose. it has been used to compare the performance of the best-route strategy, which is chosen to be used in the new design, and the broadcast strategy, which is currently used in the NDN architecture. Simple scenario has been chosen to prove the work and to determine the challenges. The scenario consisted of nine routers , two consumers and one producer and as mentioned in Figure 4.5 below.

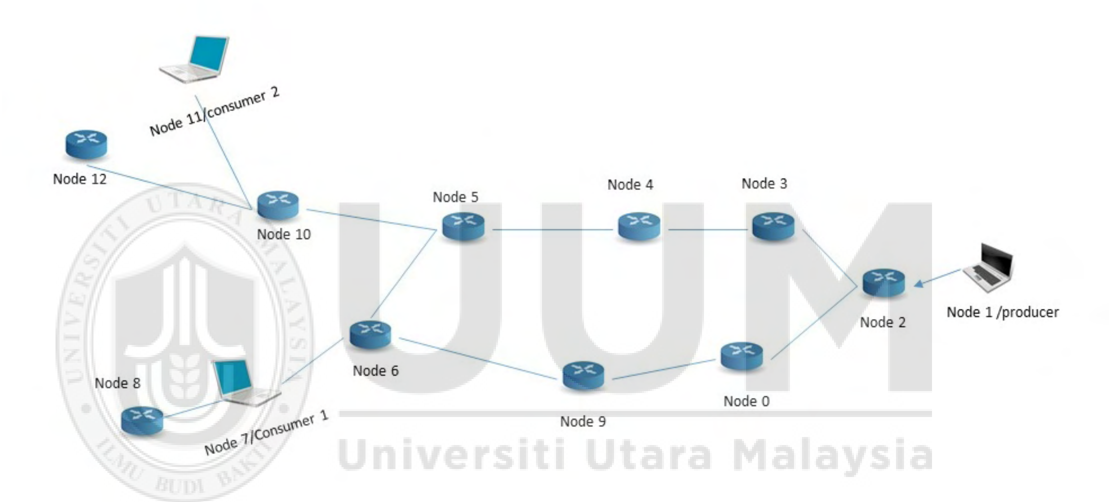


Figure 4.5: Simulation Topology

4.5 Results and discussions

Using the name resolution system guarantees that the NDO will be found and this is one of the goals in this research, thus will introduce another type of routing mechanism that will enhance the QoS of the sysem by avoiding any congestion that may occur on any node by reducing the amount of traffic and decrease the number of the exchanged packets.

Reducing the amount of traffic and the number of packet on each node will reduce the scalability problem in NDN [10] which is a common problem in the most ICN approaches. On the other hand, the delay time on link failure is one of the must important metrics to check the QoS since the last idea from the NDN is to be as good as possible for future Internet.

4.5.1 Amount of traffic

The collected results from ndnSIM are shown in the following tables and charts. Figure 4.6 illustrate the difference between the total number of packets in each strategy and based on the data mentioned in table (4.2) . It is clear that the number of interest packets are twice the number of the data packet in broadcast forwarding strategy, while there is no big difference between them in best-route forwarding strategy. This because the nodes in broadcast strategy keep sending a multicast message to all nodes in the topology in order to find the specific route to forward the data.On the other hand the best-route strategy will send the interest packet only through the interface which it connected to the node in the shortest way to the producer. Reducing the number of interest packets will reduce the amount of traffic on all the nodes resulting less congestion on the whole topology and as shown in (Figure 4.6) and table (4.3).

Metric	Best Route	Broadcast
Total In Interest Packets	575	1012
Total In Data Packets	297	450
Total Out Data Packets	330	500
Total Out Interests Packets	473	950
TimeOut Interests Packets	0	60

Table 4.2: Total number of packet

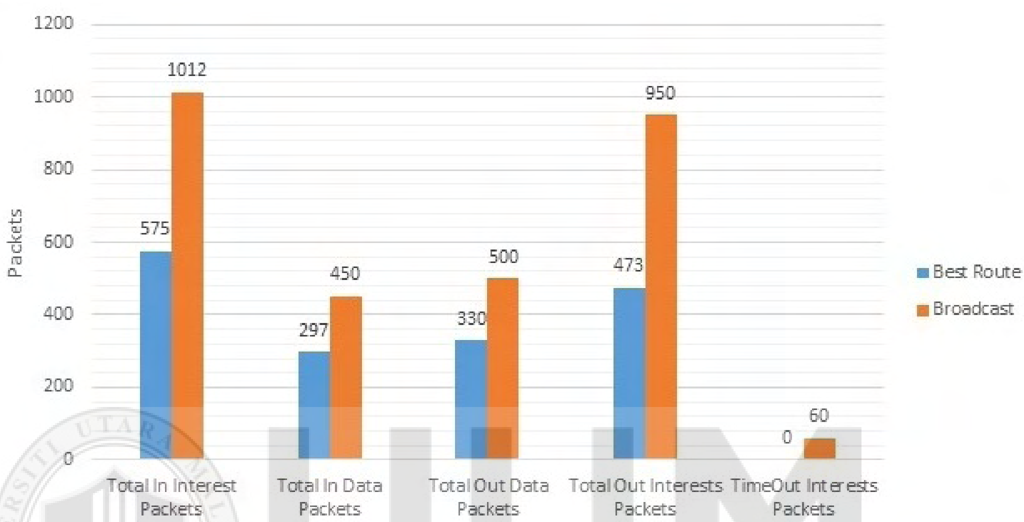


Figure 4.6: Total number of packet

Metric	Best Route	Broadcast
Total In Interest Kilobytes Raw	13.2	24.3
Total In data Kilobytes Raw	307	465.3
Total Out Interests KiloBytes Raw	9.4	21.5
Total Out Data KiloBytes Raw	341.2	517
TimeOut Interests KiloBytes Raw	0	0

Table 4.3: Total amount of traffic

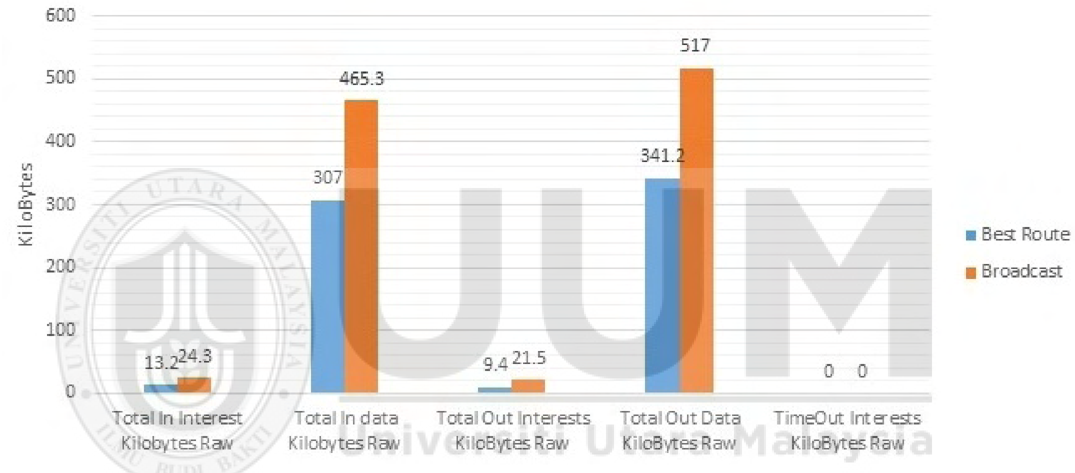


Figure 4.7: Total amount of traffic

The total number of transferred packets and the overall traffic is reduced when using best-route strategy in compared with broadcast strategy, this is due to the fact that many nodes in the topology are not carrying traffic as they are not being on the “best-route” of the data. However, this is not the same scenario when using broadcast strategy, as the data are flooded to the whole nodes inside the network. This comparison is shown in figure 4.7 and 4.8 below.

Metric	Best Route	Broadcast
In Interest Kilobytes Raw For Node 3	0.076	2.03
In data Kilobytes Raw For Node 3	3.1	20.6
Out Interests KiloBytes Raw For Node 3	0.076	1.26
Out Data KiloBytes Raw For Node 3	3.1	20.6
TimeOut Interests KiloBytes Raw For Node 3	0	0

Table 4.4: Traffic on node 3

Metric	Best Route	Broadcast
In Interest Kilobytes Raw For Node 4	0.076	1.26
In data Kilobytes Raw For Node 4	3.1	20.6
Out Interests KiloBytes Raw For Node 4	0.076	1.26
Out Data KiloBytes Raw For Node 4	3.1	20.6
TimeOut Interests KiloBytes Raw For Node 4	0	0

Table 4.5: Traffic on node 4

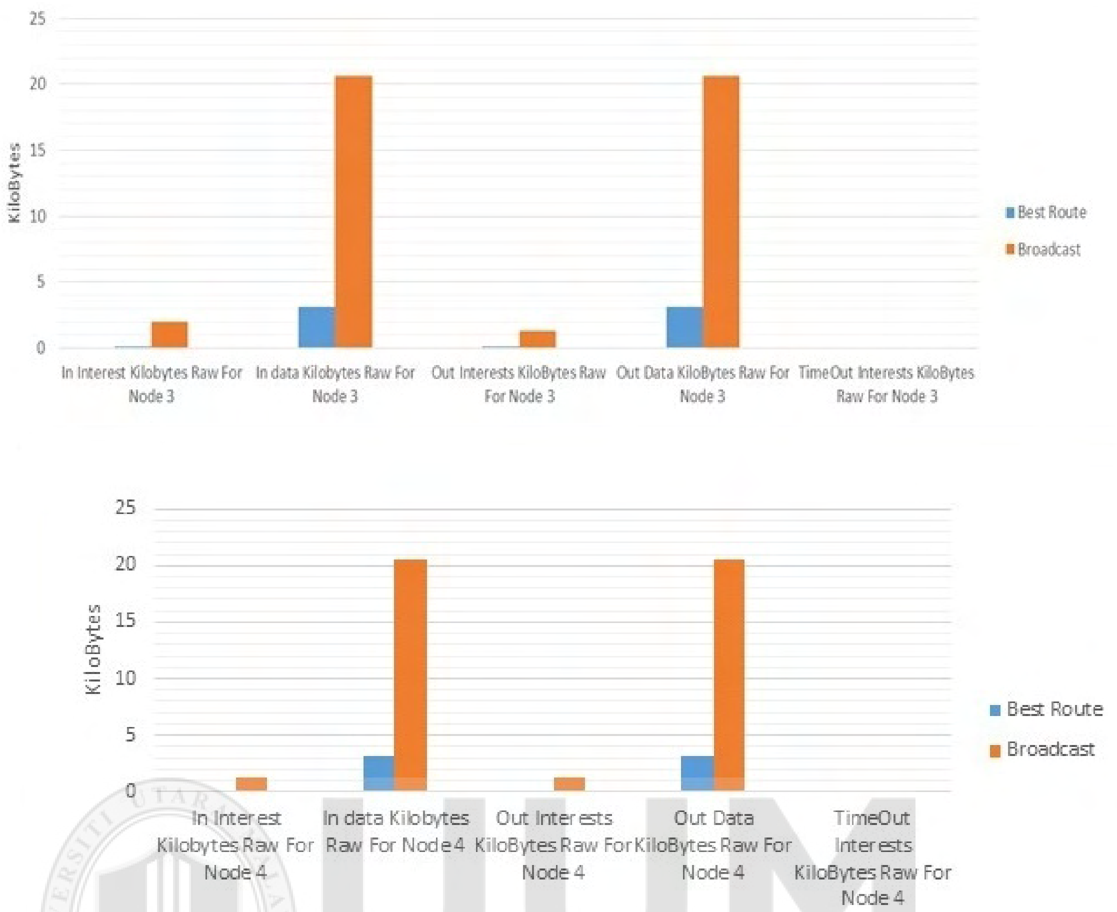


Figure 4.8: Traffic on node 3 and 4

By taking the amount of traffic on the producer side, there is a clear difference between the two strategies, which prove that the best route strategy is much better than the broadcast. This is emphasis one of the ICN basic goals which is to reduce the traffic amount on the sever side, this is shown in Figure 4.8

Metric	Best Route	Broadcast
Total In Interest Kilobytes Raw	13.2	24.3
Total In data Kilobytes Raw	307	465.3
Total Out Interests KiloBytes Raw	9.4	21.5
Total Out Data KiloBytes Raw	341.2	517
TimeOut Interests KiloBytes Raw	0	0

Table 4.6: Traffic on producer node

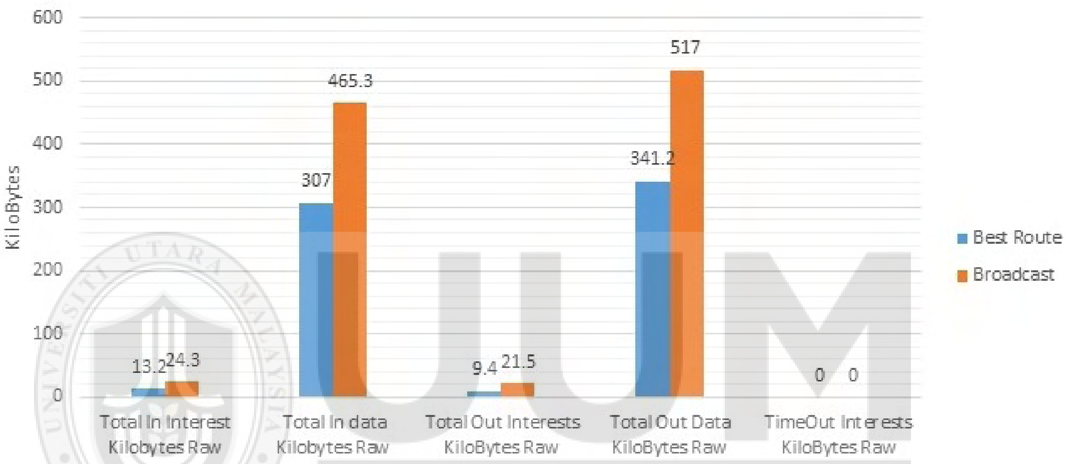


Figure 4.9: Traffic on producer node

4.5.2 Link Failure

The link failure scenario was tested by triggering an intentional failure in one of the topology links at the third second of the scenario. The gained results didn't satisfy the idea of our strategy. This is because even after the link failure, the packets exchanging seems to be completely stopped for a while until getting new route to transfer the packets. Figure(4.10) shows the traffic routes to consumer which is represented in node 7, before and after the link failure . In fact, there is no real stop, what actually occurred is that the client keep sending interest packets until it get NACK (Negative-

Acknowledgment), after getting this message, it will start looking for a new route, once the new route is obtained, it will send only one packet through it, then it will recheck of the old route is back to service or not by waiting for another NACK message and so on. All of this will create huge delay in the network. Figure 4.9 shows the time delay before and after the failure. It is seen that the time delay is the same for both strategies before the link failure, however, once the link failure took place, the time delay was increased almost three times in broadcast strategy if compared with the best-route one.

	Best Route	Broadcast
Before Link Failure	0.321	0.321
After Link Failure	3.485	0.385

Table 4.7: Time delay

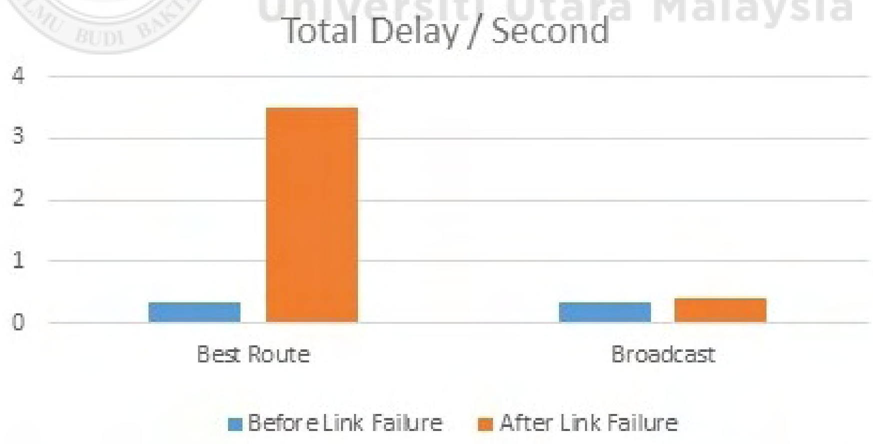


Figure 4.11: Time delay

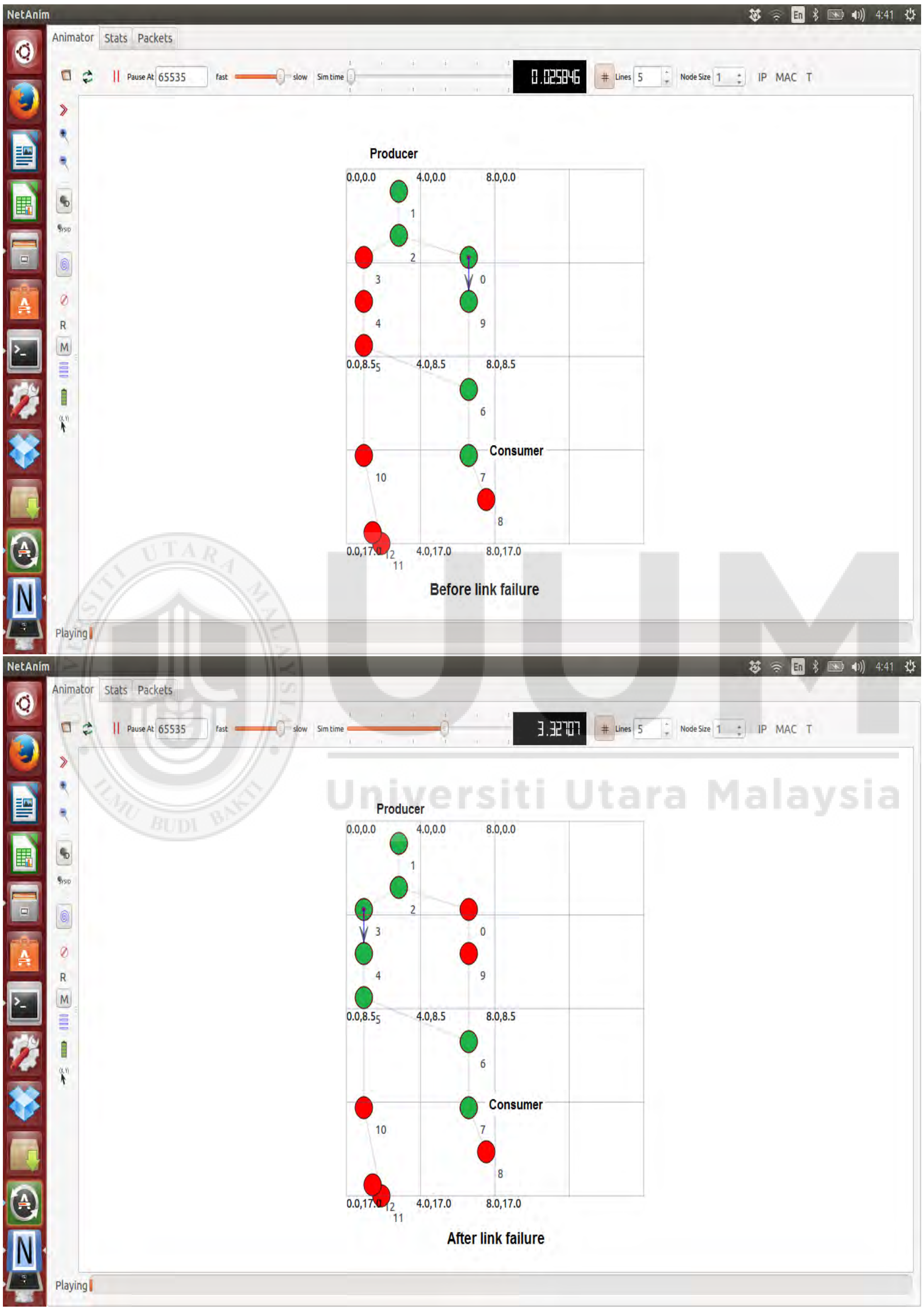


Figure 4.10: Traffic routes before and after link failure

4.6 Summary

This chapter highlighted the need for a decentralizes NRS that avoiding the point of failure which is needed have the efficient location's database for each object in the network. This is fulfill the requirement for the first objective of this thesis. It is also achieved the second objectives by implementing the DSL for NRS in NDN architecture and explain the user possibilities in dealing with the system in more transparent and easy way. The new design shows significant impact on NDN architecture in term of security by allowing the user to choose the level of data distribution, in fact, this will also reduce the needed storage as the data will be stored where they are needed only (inside their related ISP zone only for example). This chapter ending by represent the the results of the evaluation of the performance of the best route strategy which is used with the new design instead of the broadcast strategy which is currently used in NDN for data looking.

CHAPTER FIVE
CONCLUSION AND FUTURE WORKS

This thesis aims in developing a Distributed Source Locator (DSL) for Name Resolution System (NRS) in Named Data Network (NDN) and evaluate the effect of this new technique in the performance of the forward strategy used by NDN via simulation. This chapter provides the conclusion of the research work. It starts with Section 5.1, where the research findings and importance of the NRS using the ODHT are discussed with possible implementations and its benefits in the data forwarding in NDN networks. In Section 5.2 the contributions made of this research are highlighted. The limitation and the challenges of the research is then followed in Section 5.3. Finally, Section 5.4 provides some suggestions for further studies.

5.1 Summary of the Research

The new idea of having DSL for NRS in NDN is build based on the DNS idea within current Internet. This idea introduce an good way for data looking. Traditional IP-based networks are looking for data based on their location, NDN's communication model retrieves data based on their name, the data looking is done hop by hop by sending multicast messages to all of the nodes. In fact this is not a guaranteed way to find a message. By having the NRS, each user can announce for the data that he have in any zone or in the whole world web. Depending on DSL for NRS, the node can look for any data and retrieve its location to choose the right way of data forwarding. All the nodes in the topology are going to use the best-route strategy for data forwarding states to learn the successful outgoing faces for Interest forwarding. With this learning mechanism, the FIB table entries are learned automatically from successful Data retrieval. The Interest forwarding is optimized owing to self-adaptive path discovery for the required data or content. We evaluated the data delivery performance by comparing the best-route mechanism with the broadcast/flooding mechanism of the FIB. In additional to these comparison the best route mechanism had been tested under adverse conditions such as link failure. The simulation results show that our FIB learning mechanism works efficiently to reduce redundant traffic and provides good performance when handling link failures. But unfortunately the delay time for best-route with the link failure was not satisfied to the needed aims of this research. Finally, Today's enormously increasing number of mobile end-user devices, such as smart phones and tablet PCs, mobility becomes a major concern. Thus, an application example was provided to demonstrate the FIB forwarding mechanism's resilience when handling mobile nodes in NDN.

5.2 Research Contributions

The overall contribution of this research is to develop an new way for looking for data to improve the performance in finding and forwarding the data object in networks. The ODHT conceptual model is designed to investigate the impact of DSL for NRS in avoiding the point of failure in that may happen if there is only one central NRS. Furthermore, Implementing the NRS will reduce the traffic amount on all node in specific topology.

5.3 Research Limitation

This Research have many limitation since it revolves around the ICN which is the idea for the future Internet. Many projects under ICN are currently available under research of them and the NDN is one of them. Due to this, there are no real components to work on, and most of the current research are depending simulators environments as the ndnSIM which had been used in this research. However. even the used ndnSIM is new simulator with few advanced tutorials and papers showing the proper way of using it.

5.4 Future Works

The proposed conceptual model in this research improve the performance of looking for data in NDN project. However, there are some limitations as well as pending works that can be pursued for future work. This section outlines some possible extensions as follows:

Enhancing the Chord's way of nodes tracking by making it in a bidirectional form instead of the one direction way currently implemented. By doing this, faster performance is expected from the system relying on ring topology as the looking up will be based on the shortest distance from a node based on two direction instead of only one.

On the other hand as previously mentioned in this thesis, both of the best-route and the broadcast forwarding strategies have advantages and disadvantages. Generally, Best-route strategy is better in handling network traffic, however, it shows high delays in the case of a link failure that may occur in the network. on the other hand, broadcast strategy shows better behavior at the link failure scenario. Due to this, having a hybrid forwarding strategy that can work according to the best-route strategy during normal operation and according to change to a new route in the broadcast strategy during the link failure can be put as a future work for our study.

Achieving the above goals will enhance the overall system performance by increasing the probabilities of finding an object located in any part of the network at all times and avoiding the possible points of failure that may lead to data unavailability. Also, this will enhance the performance speed of looking up for the data objects.

REFERENCES

[1] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 2, pp. 1024–1049, 2014.

[2] C. Dannewitz, M. D. Ambrosio, and V. Vercellone, "Mdht: a hierarchical name resolution service for information-centric networks," in *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*. ACM, 2011, pp. 7–12.

[3] W. Sappanyoovith, W. Muttitanon, S. Tritilanunt, T. Phienthrakul *et al.*, "The performance evaluation of the content centric network on uninet network," 2014.

[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," *Communications of the ACM*, vol. 55, p. 117, 2012.

[5] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang *et al.*, "Named data networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 66–73, 2014.

[6] D. Gunatilaka, "Recent information-centric networking approaches," *http://www.cse.wustl.edu/ jain/cse570-13/ftp/icn/*, pp. 1–16, 2013.

[7] D. E. Comer and L. J. Peterson, "A name resolution model for distributed systems," 1984.

[8] A. Kaluszka, "Distributed hash tables," *System*, April 2010.

[9] J. W. Lee, H. Schulzrinne, W. Kellerer, and Z. Despotovic, "mdht: multicast-augmented dht architecture for high availability and immunity to churn," in *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*. IEEE, 2009, pp. 1–5.

[10] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking (draft)," *Information-Centric Networking*, pp. 1–26, 2011.

[11] R. Wang, "Container resolution system in icn draft-ietf-icnrg-icn-container-name-system," pp. 1–16, 2013.

[12] N. B. Melazzi, A. Detti, M. Arumaithurai, and K. Ramakrishnan, "Internames: a name-to-name principle for the future internet," in *Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), 2014 10th International Conference on*. IEEE, 2014, pp. 146–151.

[13] A. A. Barakabitze, T. Xiaoheng, and G. Tan, "A survey on naming, name resolution and data routing in information centric networking (icn)," *international Journal of Advanced Research in Computer and Communication Engineering*, vol. 2013, 2014.

[14] [Online]. Available: http://newsroom.cisco.com/dlls/2010/prod_060210.html

[15] M. Bari, S. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu, "A survey of naming and routing in information-centric networks," *Communications Magazine, IEEE*, vol. 50, no. 12, pp. 44–53, 2012.

[16] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4. ACM, 2007, pp. 181–192.

[17] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, D. Massey, C. Papadopoulos *et al.*, "Named data networking (ndn) project," *Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC*, 2010.

[18] N. Fotiou, P. Nikander, D. Trossen, and G. C. Polyzos, "Developing information networking further: From psirp to pursuit," in *Broadband Communications, Networks, and Systems*. Springer, 2012, pp. 1–13.

[19] C. Dannewitz, "Netinf: An information-centric design for the future internet," in *Proc. 3rd GI/ITG KuVS Workshop on The Future Internet*, 2009.

[20] I. Seskar, K. Nagaraja, S. Nelson, and D. Raychaudhuri, "Mobilityfirst future internet architecture project," in *Proceedings of the 7th Asian Internet Engineering Conference*. ACM, 2011, pp. 1–3.

[21] J.-C. Lee, W.-S. Lim, and H.-Y. Jung, "Scalable domain-based routing scheme for icn," *IEEE*, pp. 770–774, 2014.

[22] M. Bari, S. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu, "A survey of naming and routing in information-centric networks," *Communications Magazine, IEEE*, vol. 50, no. 12, pp. 44–53, 2012.

[23] A. Karakannas and Z. Zhao, "Information centric networking for delivering big data with persistent identifiers," *University of Amsterdam*, 2014.

[24] M. Kobayashi, T. Murase, and A. Kuriyama, "A longest prefix match search engine for multi-gigabit ip processing," in *Communications, 2000. ICC 2000. 2000 IEEE International Conference on*, vol. 3. IEEE, 2000, pp. 1360–1364.

[25] L. Chiariglione, A. Difino, N. B. Melazzi, S. Salsano, A. Detti, G. Tropea, A. Anadiotis, A. Mousas, I. Venieris, and C. Patrikakis, "Publish/subscribe over information centric networks: a standardized approach in convergence," *Future Network & Mobile Summit*, pp. 1–8, 2012.

[26] C. Dannewitz, M. D□Ambrosio, and V. Vercellone, "Hierarchical dht-based name resolution for information-centric networks," *Computer Communications*, vol. 36, no. 7, pp. 736–749, 2013.

[27] Y. Zhu and A. Nakao, "A practical study on distributed resolution service for icn," in *Computer Software and Applications Conference Workshops (COMPSACW), 2013 IEEE 37th Annual*. IEEE, 2013, pp. 736–741.

[28] C. Dannewitz, "Netinf: An information-centric design for the future internet," in *Proc. 3rd GI/ITG KuVS Workshop on The Future Internet*, 2009.

[29] [Online]. Available: http://1000projects.org/types-of-failures-in-distributed-systems.html

[30] F. Eliassen, "Introduction to distributed systems (ds)," 2006, p 1-17.

[31] L. T. Blessing and A. Chakrabarti, *DRM, a design research methodology*. Springer Science & Business Media, 2009.

[32] A. HABBAL, "Tcp sintok: Transmission control protocol with delay-based loss detection and contention avoidance mechanisms for mobile ad hoc networks."

[33] A. Afanasyev, I. Moiseenko, L. Zhang *et al.*, "ndnsim: Ndn simulator for ns-3," *University of California, Los Angeles, Tech. Rep*, 2012.

[34] O. Balci, "Verification validation and accreditation of simulation models," in *Proceedings of the 29th conference on Winter simulation*. IEEE Computer Society, 1997, pp. 135–141.

[35] ——, "Validation, verification, and testing techniques throughout the life cycle of a simulation study," *Annals of operations research*, vol. 53, no. 1, pp. 121–173, 1994.

[36] E. Davies, G. Tyson, B. Ohlman, P. Mahadevan, D. Gellert, S. Eum, S. Spirou, D. Corujo, K. Pentikousis, A. Molinaro *et al.*, "Icn baseline scenarios and evaluation methodology," 2013.

[37] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 149–160, 2001.

[38] P. Kersch and R. Szabo, "Mathematical modeling of routing in dhts," in *Handbook of Peer-to-Peer Networking*. Springer, 2010, pp. 367–401.

[39] P. Merz and K. Gorunova, "Fault-tolerant resource discovery in peer-to-peer grids," *Journal of Grid Computing*, vol. 5, no. 3, pp. 319–335, 2007.