

République Algérienne Démocratique et Populaire  
Ministère de l'enseignement Supérieur et de la Recherche Scientifique  
Université de Mohamed El Bachir El Ibrahimi de Borj Bou Arreridj  
Faculté des Mathématiques et d'Informatique  
Département d'informatique



## **MEMOIRE**

Présenté en vue de l'obtention du diplôme

### **Master en informatique**

Spécialité : Technologie de l'Information et de Communication

## **THEME**

# Une étude comparative des réseaux basés sur Named Data Networking et IP paradigme

*Présenté par :*

Tehami Amel

Attia Hadjer

*Soutenu publiquement le :* 22/06/2023

*Devant le jury composé de :*

**Président :** Benabid Sonia

**Examineur :** Belalta Ramla

**Encadreur :** Khelifi Hakima

MCB, Université de BBA

**2022/2023**

# Remerciement

*Nous à remercier en premier lieu « Allah » Tout-Puissant et Miséricordieux, qui Nous a donné la force et la patience de faire cet humble travail.*

*Nous adressons nos sincères remerciements à « Mme KHÉLIFI HAKIMA », à qui nous exprimons toute notre gratitude pour son encadrement et ses encouragements tout au long des travaux. Nous la remercions pour sa bonne volonté, sa patience et ses précieux conseils ainsi que l'importance de ses commentaires. Nous sommes très heureux travailler avec lui.*

*Nous remercions également les membres du jury d'avoir accepté d'examiner notre travail et partager leurs avis pertinents avec nous. Nous exprimons également nos sincères remerciements Mes remerciements vont également à tous les enseignants de la Faculté des Mathématiques et informatique à Bordj Bou Arreridj qui contribue directement ou indirectement à notre formation.*

*Enfin, c'est avec beaucoup de plaisir que nous pouvons remercier profondément nos familles pour leur prière et leurs encouragements, nous avons pu surmonter tous les obstacles.*

# Dédicace

*Le voyage n'était pas court et il ne devait pas l'être, le rêve n'était pas proche, ni la route pleine d'installations, mais je l'ai fait.*

*Je dédie ma graduation à celui dont je porte le nom avec fierté, à celui qui a récolté les épines sur mon chemin pour paver le chemin de la connaissance à mon « **cher père** ».*

*Après la grâce de **Allah**, ce que je suis est dû à mon père, l'homme qui n'a même pas obtenu une petite partie de ce que nous avons obtenu, et l'homme qui s'est efforcé toute sa vie d'être meilleur que lui.*

*À la main invisible qui a enlevé les épines de mon chemin et qui a enduré chaque moment de douleur que je passais et m'a soutenu dans ma faiblesse et mon enjouement, «**ma mère bien-aimée** ».*

*A mes sœurs, et à tous ceux qui m'ont aidée et soutenue sur ce chemin, je vous suis reconnaissante à toutes, je ne serais pas arrivée sans votre grâce après **Allah**.*

*Tehamí Amel*

# Dédicace

*A ma très chère mère*

*Quoi je fasse ou que de dise, je ne saurai point te remercier comme il se doit. Ton affection me couvre, ta bienveillance me guide et ta présence à mes côtés a toujours été ma source de force pour affronter les différents obstacles.*

*A mon très cher père*

*Tu as toujours été à mes cotes pour me soutenir et m'encourager.*

*Que ce travail traduit ma gratitude et mon affection.*

*A mes chers frères et sœurs, merci pour votre soutien.*

*Puisse dieu vous donne santé, bonheur, courage et surtout réussite*

*Attia Hadjer*

# Résumé

Le protocole Internet (IP) est une couche réseau universelle qui implémente toutes les fonctionnalités nécessaires à l'interconnectivité mondiale, mais sa nature de communication point à point n'est pas favorable aux tendances récentes selon lesquelles Internet est principalement utilisé comme réseau de distribution, fournissant des contenus partout sur le réseau. Named Data Networking (NDN) est une architecture entièrement nouvelle visant à se concentrer sur le contenu lui-même plutôt que sur le chemin pour obtenir le contenu. Cette recherche est menée pour comparer les performances du réseau NDN au réseau IP selon trois paramètres : le débit, la perte de paquets et le délai. Les résultats montrent en général, NDN entraîne un délai inférieur en raison de l'existence d'un content store. NDN peut également surpasser IP en termes de débit, tandis que dans certains nœuds, il peut également obtenir les mêmes résultats avec IP. La perte de paquets dans un réseau IP est plus performante que dans NDN.

# Abstract

Internet Protocol (IP) has been a universal network layer, which implements all the functionality necessary for global interconnectivity, but its point-to-point communication nature is not beneficial to the recent trends, which Internet is primarily being used as a distribution network, delivering contents all over the network. Named Data Networking (NDN) is an entirely new architecture aimed to focus on the contents itself instead of the path to get the contents. This research is conducted to compare the performance of NDN to IP-based network in three parameters: throughput, packet drop, and delay. The results show in general, NDN results in lower delay because of the existence of content store. NDN can also outperforms IP in term of throughput, while in some nodes it can also gain same results with IP. Packet loss in an IP network is more efficient than in NDN.

# ملخص

كان بروتوكول الإنترنت (IP) عبارة عن طبقة شبكة عالمية تقوم بتنفيذ جميع الوظائف اللازمة للتوصيل البيئي العالمي، ولكن طبيعة الاتصال من نقطة إلى نقطة ليست مفيدة للاتجاهات الحديثة التي يتم استخدام الإنترنت بشكل أساسي كشبكة توزيع، مما يوفر المحتويات في جميع أنحاء الشبكة. شبكة البيانات المسماة (NDN) هي بنية جديدة تمامًا تهدف إلى التركيز على المحتويات نفسها بدلاً من المسار للحصول على المحتويات. تم إجراء هذا البحث لمقارنة أداء NDN بالشبكة القائمة على IP في ثلاث معاملات: الإنتاجية، وإسقاط الحزمة، والتأخير. تظهر النتائج بشكل عام، ينتج عن NDN تأخير أقل بسبب وجود مخزن المحتوى. يمكن أن يتفوق NDN أيضًا على IP من حيث الإنتاجية، بينما في بعض العقد يمكنه أيضًا الحصول على نفس النتائج مع IP. بالمقارنة مع NDN، يعد فقدان الحزم في شبكة IP أكثر فعالية.

# Table des matières

Liste des abréviations.....	xi
Liste des figures .....	xiii
Liste des tableaux.....	xiv
<b>Chapitre 01 : Introduction Générale.....</b>	<b>1</b>
1.1. Contexte .....	1
1.2. Objectifs.....	2
1.3. Méthodologie et résultats.....	2
1.4. Structure du rapport .....	2
<b>Chapitre 02 : Compréhension sur les bases IP.....</b>	<b>4</b>
2.1. Introduction.....	4
2.2. Réseau base sur le protocole IP .....	4
2.2.1. C'est quoi un réseau informatique ? .....	4
2.2.2. Classification des réseaux.....	4
2.2.3. Topologies des réseaux .....	5
2.2.4. Le routage .....	6
2.2.5. Réseau Internet.....	7
2.2.6. Architecture des réseaux .....	7
2.2.7. Adressage IP .....	12



2.3. Conclusion .....	14
<b>Chapitre 03 : Compréhension sur les bases du NDN.....</b>	<b>15</b>
3.1. Introduction.....	15
3.2. Named Data Networking .....	15
3.2.1. Architecture NDN.....	15
3.2.2. Communication dans NDN.....	17
3.2.3. Caractéristique du NDN.....	19
3.2.4. Les bases de la mise en réseau des données nommées .....	19
3.3. Conclusion .....	24
<b>Chapitre 04 : Évaluation et analyse des performances .....</b>	<b>25</b>
4.1. Introduction.....	25
4.2. NDN vs IP.....	25
4.2.1. Nommage et l'Adressage.....	25
4.2.2. Routage et transmission .....	26
4.2.3. Contrôle de la congestion.....	26
4.2.4. La mise en cache en réseau .....	26
4.3. Evaluation des performances .....	26
4.3.1.Présentation des simulateurs NS-3/ndnSIM .....	28
4.4. Scénarios de simulation .....	31
4.5. Résultats de la Simulation.....	32
4.6. Évaluation .....	38

4.7. Conclusion .....	40
<b>Chapitre 05 : Conclusion générale .....</b>	<b>41</b>
5.1. Contributions.....	41
5.2. Critique du travail .....	42
5.3. Travaux futurs et perspectives .....	42
<b>Les références .....</b>	<b>43</b>

# Liste des abréviations

<b>IP</b>	<b>I</b> nternet <b>P</b> rotocol
<b>TCP</b>	<b>T</b> ransmission <b>C</b> ontrol <b>P</b> rotocol
<b>OSI</b>	<b>O</b> pen <b>S</b> ystems <b>I</b> nterconnection
<b>NDN</b>	<b>N</b> amed <b>D</b> ata <b>N</b> etworking
<b>PAN</b>	<b>P</b> ersonal <b>A</b> rea <b>N</b> etwork
<b>WAN</b>	<b>W</b> ide <b>A</b> rea <b>N</b> etwork
<b>MAN</b>	<b>M</b> etropolitan <b>A</b> rea <b>N</b> etwork
<b>LAN</b>	<b>L</b> ocal <b>A</b> rea <b>N</b> etwork
<b>FTP</b>	<b>F</b> ile <b>T</b> ransfer <b>P</b> rotocol
<b>SSH</b>	<b>S</b> ecure <b>S</b> ocket <b>S</b> hell
<b>SFTP</b>	<b>S</b> ecure <b>F</b> ile <b>T</b> ransfer <b>P</b> rotocol
<b>DNS</b>	<b>D</b> omaine <b>N</b> ame <b>S</b> ystem
<b>HTTP</b>	<b>H</b> yper <b>T</b> ext <b>T</b> ransfer <b>P</b> rotocol
<b>SNMP</b>	<b>S</b> imple <b>N</b> etwork <b>M</b> anagement <b>P</b> rotocol
<b>UDP</b>	<b>U</b> ser <b>D</b> atagram <b>P</b> rotocol
<b>ICN</b>	<b>I</b> nformation <b>C</b> entric <b>N</b> etwork
<b>CS</b>	<b>C</b> ontent <b>S</b> ore
<b>PIT</b>	<b>P</b> ending <b>I</b> nterest <b>T</b> able
<b>FIB</b>	<b>F</b> orwording <b>I</b> nformation <b>B</b> ase

**NAT**    **N**etwork **A**ddress **T**ranslation

**OSPF**   **O**pen **S**hortest **P**ath **F**irst

**BGP**    **B**order **G**ateway **P**rotocol

**NLSR**   **N**amed-Data **L**ink **S**tate **R**outing

**LCE**    **L**eave **C**opy **E**verywhere

**LCD**    **L**eave **C**opy **D**own

**ndnSIM** **N**amed **D**ata **N**etworking **S**IMulator

**NS-3**    **N**etwork **S**imulator **3**

**Wi-Fi**   **W**ireless **F**idelity

# Liste des figures

Figure 1 : Les Types des topologies des réseaux [3] .....	6
Figure 2 : Les couches du modèle de référence OSI .....	8
Figure 3 : Les Couches du modèle de référence TCP/IP .....	11
Figure 4 : Le paquet d'intérêt et le paquet de données dans NDN .....	16
Figure 5 : Exemple sur la communication en NDN.....	17
Figure 6 : Processus de transfert dans un nœud NDN [10].....	18
Figure 7 : Topologie d'un cas d'étude .....	31
Figure 8 : Throughput 100 paquets (a) /Throughput 10 paquets (b) en NDN .....	33
Figure 9 : Delay 100 paquets (a) /Delay 10 paquets (b) en NDN.....	34
Figure 10 : Packet loss 100 paquets (a)/ Packet loss 10 paquets (b) en NDN .....	35
Figure 11 : Throughput scenario IP .....	36
Figure 12 : Delay scenario IP.....	37
Figure 13 : Packet loss scenario IP .....	38

## Liste des tableaux

Tableau 1 : Les classes des adresses IP privés .....	13
Tableau 2 : Les classes des adresses IP publiques .....	13
Tableau 3 : Comparaison entre internet actuelle et NDN .....	27

# Chapitre 01 : Introduction Générale

## 1.1. Contexte

Internet a été développé comme un réseau de données par paquets où les utilisateurs et les sources de données (serveur) avec des adresses IP (Internet Protocole) spécifiques interagissaient sur un canal de communication préétabli.

Ce modèle de communication de données client-serveur a évolué vers un mode de partage de données peer-to-peer ces derniers temps. Des applications comme YouTube, Bit Torrent, les réseaux sociaux ont révolutionné l'idée de contenu généré par l'utilisateur. Les utilisateurs modernes ne s'intéressent qu'à des éléments de données spécifiques, quelles que soient leurs sources. Ainsi, l'idée d'utiliser des adresses IP pour identifier les serveurs hébergeant un contenu particulier perd de son importance. De plus, le manque d'adresses IP est un problème qui hante la communauté Internet depuis longtemps.

Le besoin du moment est une plate-forme de réseau centrée sur le contenu où les hôtes de données sont de moindre importance, et le réseau Named Data Networking (NDN) a été proposé à cette fin. NDN permet aux utilisateurs de faire flotter une demande de données sans aucune connaissance de l'entité d'hébergement. NDN peut gérer la mobilité des utilisateurs et les problèmes de sécurité plus efficacement que l'Internet actuel. Bien que le NDN ait été proposé en 2010, jusqu'à présent, il n'existe aucun document d'enquête étudiant son architecture et les divers schémas proposés pour ses différentes caractéristiques, telles que la dénomination, le transfert et le routage adaptatifs, la mise en cache, la sécurité, la mobilité, etc.

## 1.2. Objectifs

Le but de notre travail est de faire une comparaison entre un réseau IP et un réseau NDN, pour trouver la meilleure technologie pour traiter les données sans problème. Dans cette rubrique, nous traitons et analysons les fichiers de données de traceur pour les deux technologies. Nous avons utilisé les modèles ndnSIM et NS-3 pour extraire les informations. Nous comparons les deux modèles pour trouver le meilleur modèle pour le contrôle de la congestion.

### 1.3. Méthodologie et résultats

Dans ce sujet, nous avons utilisé un ensemble de fichiers de trace, nous traitons d'abord ces fichiers et extrayons des données utiles. Nous avons utilisé Wireshark et Microsoft Excel sur ces données et les convertissons en courbes, pour nous aider dans le processus de comparaison IP et NDN et savoir quelle technologie est la meilleure. Après avoir obtenu ces courbes, nous concluons qu'IP fonctionne mieux dans le réseau, grâce au modèle de référence dont il dispose, et que le modèle NDN a encore besoin de développements dans son système de fonctionnement.

### 1.4. Structure du rapport

Notre mémoire est organisée en Cinq chapitres :

Dans **le premier chapitre**, Nous avons donné l'introduction générale, le contexte, Objectifs de ce travail et Méthodologie et résultats, et Structure du rapport.

Dans **le deuxième chapitre**, Nous nous sommes concentrés sur le concept d'un réseau informatique, leurs types et les méthodes de communication dans ce réseau. Nous donnons aussi la définition de réseau Internet et leurs architectures. Enfin la définition d'une adresse IP et aussi leurs types.

Dans **le chapitre trois**, nous avons commencé par l'architecture de NDN et comment se passe le processus de communication, les caractéristiques de NDN et les bases de la mise en réseau des NDNs (Nommage, Routage, Cache, Sécurité et confidentialité).

Dans **le chapitre quatre**, nous avons fait une comparaison entre un réseau IP et le réseau NDN selon l'adressage et le nommage, routage et la transmission, contrôle de la congestion et la mise en cache en réseau. Nous avons vu aussi un aperçu des options de simulation pour les simulateurs NS-3/ndnSIM. Les termes et abstractions installent les simulateurs NS-3/ndnSIM. De plus, nous réalisons des scénarios de simulation sur lesquels notre étude a été basée et les résultats de la simulation.



Dans **le chapitre cinq**, nous avons donnée conclusion générale, contributions, critique du travail, travaux futurs et perspectives.

# Chapitre 02 : Compréhension sur les bases IP

## 2.1. Introduction

La plupart de notre travail dans ce monde est basé autour du grand réseau planétaire de base, qui est l'internet, aujourd'hui il connecte des millions d'utilisateurs qui partagent des milliards de ressources, d'informations et qui à leur tour forment notre réseau de médias automatisés autour des adresses IP.

Dans ce chapitre, nous aborderons d'abord l'architecture, classifications des types des réseaux et ses topologies. Puis, les bases de l'adresse IP, les types de ces adresses.

## 2.2. Réseau base sur le protocole IP

### 2.2.1. C'est quoi un Réseau Informatique ?

Un réseau informatique [1] est un groupe d'équipements qui sont reliés entre eux pour échanger des informations et partager des ressources entre eux comme les appareils informatiques : un ordinateur, un routeur, un concentrateur, un commutateur. Ces appareils en ce réseau utilisent un système de règles appelées protocoles de communication, pour transmettre des informations via des technologies physiques ou sans fil qu'est représenté par IP.

### 2.2.2. Classification des Réseaux

Plusieurs classifications ont été proposées pour les réseaux informatique. la classification plus utilisées présentée sur quatre classes : réseau personnel, réseau local, réseau métropolitain et réseau large. Ces réseaux sont appelés communément PAN (Personal Area Network), LAN (Local Area Network), MAN (Metropolitan Area Network) et WAN (Wide Area Network). Quant au type d'architecture matérielle : Réseaux en bus, réseaux en étoile, réseaux en anneau [2].

- **Réseau Personnel (PAN) :** Un réseau personnel connecte des appareils électroniques dans l'environnement immédiat d'un utilisateur. La taille du PAN varie de quelques centimètres à quelques mètres. Les PAN peuvent connecter des ordinateurs portables, des tablettes, des imprimantes, des claviers et d'autres appareils informatiques. Les connexions à un PAN peuvent être câblées ou sans fil. Les méthodes de connexion filaire incluent USB et FireWire ; Les méthodes de connexion sans fil incluent le Bluetooth (le plus courant) et le Wifi.
- **Réseau Local (LAN) :** est un réseau qui permet de connecter des ordinateurs dans une zone géographique relativement restreinte, par exemple, une entreprise ou une organisation, une résidence privée, un site d'affaires, pour l'échange d'informations.
- **Réseau Métropolitain (MAN) :** Un MAN est un réseau local de taille moyenne qui s'étend sur quelques kilomètres, par exemple pour être utilisé sur un campus universitaire. Un MAN relie généralement plusieurs réseaux locaux (LAN).
- **Réseau Large (WAN) :** Il couvre une vaste zone géographique, généralement à l'échelle d'un pays, d'un continent ou d'une planète entière. Le plus grand WAN est le réseau internet. Ces réseaux sont généralement l'interconnexion de plusieurs réseaux LAN et WAN.

### 2.2.3. Topologies des Réseaux

Les topologies de réseau sont :

- **Topologie en Bus :** Un réseau de type bus est ouvert à ses extrémités. Chaque PC y est connecté par l'intermédiaire d'un connecteur spécial. Certains périphériques, comme des imprimantes, peuvent également être directement reliés au réseau. Ils doivent alors comporter une carte adaptateur réseau. A chaque extrémité, le réseau est terminé par une résistance (appelé bouchon) pour empêcher l'apparition de signaux parasites. L'exemple le plus courant de ce type de réseau est le réseau Ethernet.
- **Topologie en étoile :** Dans un réseau en étoile, chaque nœud du réseau est relié à un contrôleur (ou hub) par un câble différent. Le contrôleur est un appareil

qui recevant un signal de données par une de ses entrées, va retransmettre ce signal à chacune des autres entrées sur lesquelles sont connectés des ordinateurs ou périphériques, voir d'autres contrôleurs.

- **Topologie en anneau :** Il s'agit d'un réseau local dans lequel les nœuds sont reliés en boucle fermée.

Dans la figure 1 ci-dessous, nous représentons les types des topologies d'un réseau informatique.

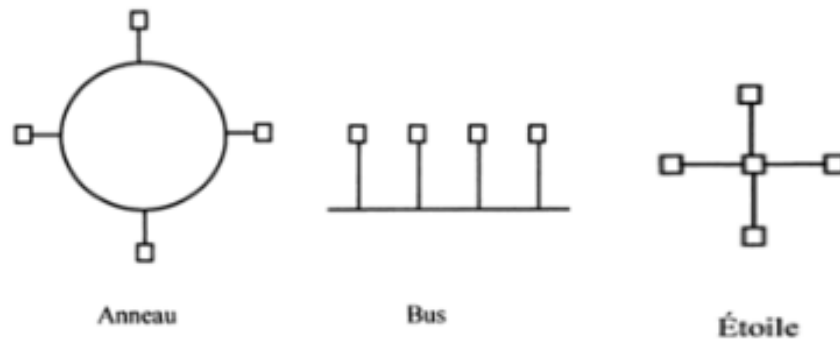


Figure 1 : Les types des Topologies des Réseaux [3]

#### 2.2.4. Le Routage

Le routage est le mécanisme par lequel le ou les meilleurs chemins d'accès dans un réseau sont choisis pour acheminer les données d'un expéditeur vers plusieurs destinataires. Le routage est mis en œuvre dans de nombreux réseaux, tels que le réseau téléphonique et les réseaux de données électroniques tels qu'Internet, et l'objectif optimal est de permettre à deux appareils de communiquer sur un chemin raisonnablement court. Cette technique est la plus connue, elle consiste à créer des tables, appelées tables de routage, dans chaque nœud ou machine du réseau. Il y a deux méthodes de routage :

- **Méthode Statique :** Dans cette méthode, les tables de routage sont remplies manuellement. Le responsable de réseau fait entrer les lignes des tables de routage, dans chaque nœud, par des commandes en ligne. Les systèmes des exploitations des machines offrent des commandes en ligne, ou même des interfaces graphiques pour remplir et mettre à jour les tables de routage.

- **Méthode Dynamique :** La méthode dynamique est proposée pour remplir tous l'inconvénient de la méthode statique. Dans ces méthodes, ce sont des algorithmes et des programmes qui remplissent et mettent à jour les tables de routages au lieu de l'intervention de l'administrateur du réseau. Ces algorithmes sont appelés protocoles de routage. La tâche principale de ce protocole est l'installation et la mise à jour des chemins dans les tables de routage.

### **2.2.5. Réseau Internet**

Internet est un réseau informatique mondial établi dans les années 1960 qui permet à des millions d'ordinateurs et de serveurs de communiquer et de partager des ressources à l'aide d'un protocole de communication commun TCP/IP (Transmission Control Protocol/Internet Protocol). Ces principaux services sont le web et le FTP (File Transfer Protocol), Ils sont utilisés pour communiquer et obtenir des informations, des ressources ou des services. Dans les années 1990, l'apparition du Web contribue à rendre Internet accessible au grand public. Depuis les années 2010, un nombre croissant de types d'objets divers y sont connectés, formant l'internet des objets.

Alors, la définition qu'on développe après ce qu'on a expliqué est que : l'internet est le réseau de réseau qui couvre toute la terre et qui utilise le modèle TCP/IP.

### **2.2.6. Architecture des Réseaux**

Architecture des réseaux est l'organisation des équipements de transmission, des logiciels, des modèles de protocoles de communication tels que TCP/IP et OSI (Open System Interconnection), appelé aussi modèle de référence de réseau. Il y a plusieurs modèles de références pour différents types de réseaux et selon différentes visions et objectives, ont été proposés. Dans cette section on va détailler deux modèles de référence, le modèle OSI et le modèle TCP/IP.

#### **2.2.6.1. Modèle OSI :**

Le modèle OSI [4] est une norme de communication en réseau pour tous les systèmes informatiques. Il s'agit d'un modèle de communication entre ordinateurs proposé par l'ISO (International Organization for Standardization) qui est utilisé pour décrire les fonctions

nécessaires d'un système en réseau. Leur objectif a défini comment les systèmes réseau communiquent et envoient des données d'un expéditeur à un destinataire, Le modèle de référence OSI en matière de réseaux sépare distinctement tous les services, interfaces et protocoles. Il est donc très flexible et les protocoles de chaque couche peuvent être remplacés facilement en fonction de la nature du réseau.

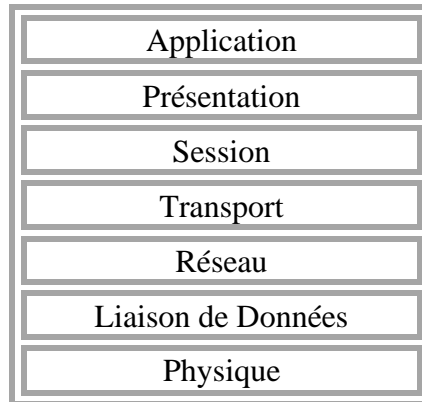


Figure 2 : Les Couches de modèle de référence OSI

Dans ce que suit on résume les tâches des couches du modèle OSI :

- **La couche physique** : La première couche d'un protocole est responsable de la Transmission efficace des signaux électriques, radio ou optiques entre les interlocuteurs pour transmettre des bits. Les équipements qui supportent la transmission sont : câble, paire torsadé, fibre, répéteurs, amplificateurs, modems, hubs.
- **La couche liaison de données** : La couche liaison de données est la seconde des sept couches du modèle OSI, cette couche constitue le premier contact avec le matériel. Parmi les tâches importantes de cette couche, on cite les suivantes :
  - ✓ **Le tramage** : la couche liaisons fragments les paquets de données qu'elle reçoit de la couche réseau en petite trame.
  - ✓ **Contrôle des erreurs** : à la réception de données, la couche liaison vérifie si les données sont correctement reçues ou non.

- ✓ **Contrôle de flux** : pour ne pas submerger une machine réceptrice par un flux de données qu'elle ne peut pas gérer, soit en raison de faible mémoire soit en raison de faible processeur, les deux machines communicantes négocient les paramètres de transmission.
  - ✓ **Gérer les acquittements** : pour s'assurer que les données sont bien reçues, la machine réceptrice doit retourner des messages de confirmation à la machine émettrice.
  - ✓ **Ordonner les paquets** : les paquets ne sont pas toujours reçus dans l'ordre de leurs émissions c'est à la charge de la couche liaison d'ordonner les paquets. Bien sûr du côté émission, la couche liaison doit numéroter les paquets.
  - ✓ **Gérer l'accès à un médium partagé** : si les machines utilisent un médium partagé pour la transmission et que plusieurs machines transmettent en même temps, les données sur le canal rentrent en collision. Pour remédier à ce problème, la couche liaison utilise une méthode d'accès au canal partagé pour éviter le maximum les collisions.
- **La couche réseau** : est la troisième couche de modèle OSI, qui correspond à la couche Internet du modèle TCP/IP. La fonction principale de cette couche est l'adressage des machines et le routage ou l'acheminement de données entre les communicantes.
  - **La couche transport** : constitue la quatrième couche du modèle OSI. La tâche principale de cette couche est l'établissement et le relâchement des connexions. La couche transport implémente le multiplexage afin de permettre à plusieurs processus de communiquer sur le même réseau. Les informations sont divisées en segments PDU (Protocol Data Unit) associés à une application spécifique. Concrètement cela consiste en l'utilisation d'un port associé à un socket pour chaque processus d'une machine. Ces informations sont contenues dans l'en-tête du segment de la couche transport.
  - **La couche session** : dans le modèle OSI, la couche session (parfois nommée « couche Port ») gère la mise en place et le retrait de l'association entre deux terminaux en communication, qu'on appelle une connexion. Des mécanismes de synchronisation sont

alors requis pour savoir par exemple qui répond à quoi. C'est le rôle des protocoles de cette couche. Quelques protocoles de la couche session : AppleTalk, NetBIOS, RPC.

- **La couche présentation** : le rôle de la couche présentation est d'établir une syntaxe de communication entre applications. Il permet la communication dans un langage commun entre différentes applications, en utilisant les données présentées de manière commune. Telle qu'elle est responsable du codage des données de l'application.
- **La couche application** : Fournit l'interface pour l'utilisateur final qui utilise un appareil connecté à un réseau. L'utilisateur voit tout dans cette couche lorsqu'il s'agit de charger une application comme le courrier électronique. Le protocole utilisé à la couche c'est Le protocole SNMP (Simple Network Management Protocol), protocole de gestion réseau simple permet d'afficher la configuration du réseau et le statut des machines-clés. SNMP permet également d'obtenir des statistiques réseaux complexes à partir d'un logiciel basé sur une interface graphique.

#### **2.2.6.2. Modèle TCP/IP (Transmission Control Protocol /Internet Protocol) :**

La première publication de TCP/IP remonte à septembre 1973 lors d'une conférence de l'INWG. Le réseau Arpanet adopte le 1er janvier 1983 la suite de protocoles TCP/IP qui sera la base d'Internet [5].

TCP/IP désigne communément une architecture réseau, mais cet acronyme désigne en fait deux protocoles étroitement liés : un protocole de transport TCP qu'on utilise un protocole réseau IP. Ce qu'on entend par modèle TCP/IP. Le modèle OSI forme un logiciel de réseau complet. Il gère le côté matériel et le côté logiciel. Par contre le modèle TCP/IP est proposé comme une solution pour l'interconnexion des réseaux.

Ce modèle contient quatre couches. Couche accès-réseau, couche internet, couche transport et couche application. Nous avons mentionné dans la figure 3.



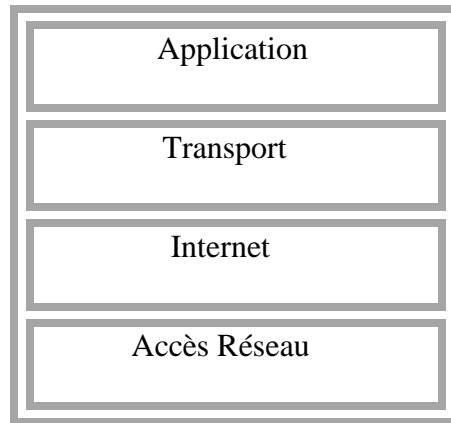


Figure 3 : Les Couches du Modèle de Référence TCP/IP

Dans ce qui suit, on résume les taches des couches du modèle TCP/IP :

- **La Couche Accès Réseau :** Couche accès réseau spécifie la forme sous laquelle les données doivent être acheminées quel que soit le type de réseau utilisé
- **La Couche Internet :** Les taches de cette couche sont l'adressage et le routage. Le Protocol le plus connu de cette couche est le Protocol IP.
- **La Couche Transport :** Cette couche définit deux protocoles TCP et UDP (User Datagram Protocol) :
  - ✓ **TCP :** est un protocole réseau qui permet à deux hôtes de se connecter et d'échanger des données.
  - ✓ **UDP :** Est l'un des autres principaux protocoles qui composent la suite de protocoles Internet. UDP est moins fiable que TCP, mais beaucoup plus simple. UDP est utilisé dans des situations où certaines pertes de données sont acceptables. Alors qu'UDP est similaire à TCP dans son utilisation pour envoyer et recevoir des données sur Internet.
- **La Couche Application :** La couche application est le contact entre l'utilisateur et le réseau. Elle offre donc des services via des protocoles (messagerie, accès au web...). Quelques protocoles utilisés par les couches. FTP, SSH (Secure Shell), SFTP (Secure File Transfer Protocol), DNS (Domain Name System), HTTP (HyperText Transfer

Protocol) Cette couche devra choisir un protocole de transport adapté au service Demandé.

## 2.2.7. Adressage IP

### 2.2.7.1. Définition d'une Adresse IP :

Une adresse IP permet d'identifier chaque équipement connecté à un réseau informatique utilisant le Protocol IP. Elle permet à l'équipement de communiqué sur le réseau auquel il est connecté [6].

### 2.2.7.2. Types des Adresse IP :

- **IPV4 (Version 4) :** notation décimale avec 4 valeurs comprises entre 0et255, séparées par des pointes.

Longueur de 32 Bits  4 Octets

- **IPV6 (Version 6) :** notation hexadécimale avec 8 valeurs sépaes par de points « : ».

### 2.2.7.3. Adresses IP Privés et Publiques :

Un des principes du protocole IP est que chaque hôte a une adresse IP unique. C'est facile dans un réseau local privé mais qu'en est-il sur Internet où plus de deux milliards d'hôtes sont interconnectés.

- **Adresse IP Privé :** sont les adresses IP que l'on peut utiliser pour les équipements sur un réseau local. Une adresse IP privé ne peut pas être utilisée sur internet, pour un site web par exemple, car elles ne sont pas accès au internet.

<b>Classe A</b>	10.0.0.0 à 10.255.255.255
<b>Classe B</b>	172.16.0.0 à 172.31.255.255
<b>Classe C</b>	192.168.1.0 à 192.168.255.255

Tableau 1 : Les classes des adresses IP privés

Le choix d'une adresse réseau s'effectue en fonction de l'usage, notamment le nombre d'hôtes à connecter sur le réseau.

- **Adresse IP Publique** : à l'inverse sont utilisées exclusivement sur internet : votre Box

Internet, un site web, etc. Une adresse IP publique est unique dans la monde contrairement à une adresse IP privé. C'est justement cela qui crée la pénurie d'IPv4 malgré la présence de 4 milliards d'adresses IP différentes.

<b>Classe A</b>	Tout sauf 10.0.0.0 à 10.255.255.255
<b>Classe B</b>	Tout sauf 172.16.0.0 à 172.31.255.255
<b>Classe C</b>	Tout sauf 192.168.1.0 à 192.168.255.255

Tableau 2 : Les classes des adresses IP publiques

## 2.3. Conclusion

Le réseau informatique qui est contrôlé par Internet et qui est contrôlé par le modèle de référence TCP/IP ont tous deux été mentionnés dans ce chapitre. Comme nous savons également que ces ordinateurs ne pouvaient se connecter entre eux que via des adresses IP, un nouveau mécanisme qui sera couvert au chapitre trois est nécessaire pour remplacer les adresses IP.

# Chapitre 03 : Compréhension sur les bases NDN

## 3.1. Introduction

Le nouveau mécanisme connu sous le nom de Named Data Networking (NDN) [7] représente un changement fondamental de paradigme de l'architecture Internet basée sur IP qui existe actuellement. Au lieu d'inclure l'adresse IP de destination dans chaque paquet, NDN insère un nom de données à la place. Lorsqu'un consommateur de données envoie un paquet d'intérêt avec un nom qui identifie les données souhaitées, la réponse est un paquet de données qui comprend le nom, les données demandées et une signature du producteur de données d'origine.

Dans ce chapitre, nous aborderons d'abord par l'architecture de NDN et comment se passe le processus de communication dans NDN, les caractéristiques de NDN et les bases de la mise en réseau (Nommage, Routage, Cache, Sécurité et confidentialité).

## 3.2. Named Data Networking

### 3.2.1. Architecture NDN

NDN [8] est un nouveau paradigme de communication et l'une des implémentations de Information Centric Network (ICN) la plus active qui utilise les noms comme composant principal pour récupérer le contenu au lieu des adresses IP.

NDN utilise deux types de paquets :

- **Paquets d'Intérêt** : est une demande déclenchée par l'utilisateur. Les intérêts contiennent le nom de la donnée, c'est à dire le préfixe de la donnée.
- **Paquets de Données** : est une réponse d'un paquet d'intérêt. Il est quant à eux composés du nom de la donnée (le même préfixe que le message d'intérêt).

Paquets dans l'architecture NDN, un paquet d'intérêt se compose du nom des données souhaitées et d'autres paramètres utilisés pour leur transmission par le réseau NDN.

Un paquet de données contient le nom des données, le contenu réel et la signature du producteur de données (figure 4).

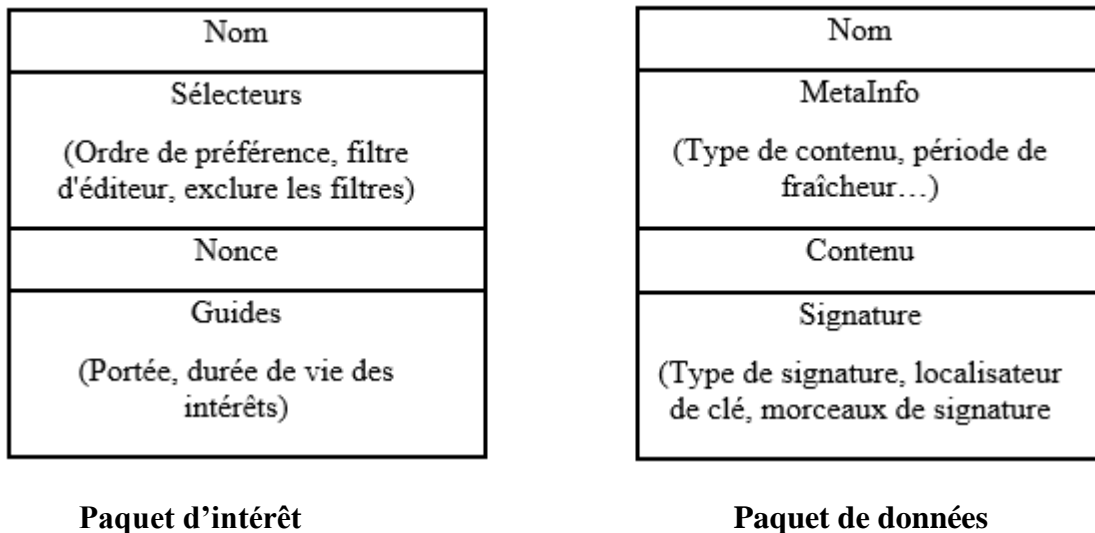


Figure 4 : Le paquet d'intérêt et le paquet de donnée dans NDN

Chaque nœud NDN comprend trois composants principaux (i.e. tableaux) :

- **Content Store (CS)** : permet de stocker des copies locales des contenus. Chaque nœud NDN pourra ainsi répondre directement aux requêtes (Intérêt) s'il possède les contenus dans son Content Store.
- **Pending Interest Table (PIT)** : permet à un nœud de conserver la liste des paquets d'intérêts qu'il a transmis et les interfaces sur lesquelles ils ont été reçus, de manière à éviter les retransmissions inutiles. Les entrées de la PIT sont aussi utilisées pour que les paquets de données puissent emprunter le chemin inverse des paquets d'intérêts.
- **Forwarding Information Base (FIB)** : conserve les entrées où doivent être transmis les paquets d'intérêts vers des sources de contenus potentielles.

### 3.2.2. Communication dans NDN

Dans NDN, le consommateur de données, ou l'extrémité réceptrice, contrôle la communication. Un consommateur envoie un paquet d'intérêt avec le nom des données demandées afin de l'obtenir [9].

Par exemple que nous avons montré dans la figure 5, un consommateur peut demander `/parc/vidéos/WidgetA.mpg`. Un routeur se souvient de l'interface de dans lequel la demande arrive, puis transmet le paquet d'intérêt en recherchant le nom dans son champ de transmission (FIB), qui est remplie par un protocole de routage basé sur le nom.

Une fois que l'intérêt atteint un nœud qui contient les données demandées, un paquet de données est renvoyé, qui porte à la fois le nom et le contenu des données, accompagnée d'une signature par la clé du producteur. Ce paquet de données trace en sens inverse le chemin créé par le paquet d'intérêt vers le consommateur. Notez que ni les paquets d'intérêts ni les paquets de données transporter toutes les adresses d'hôte ou d'interface (telles que les adresses IP).

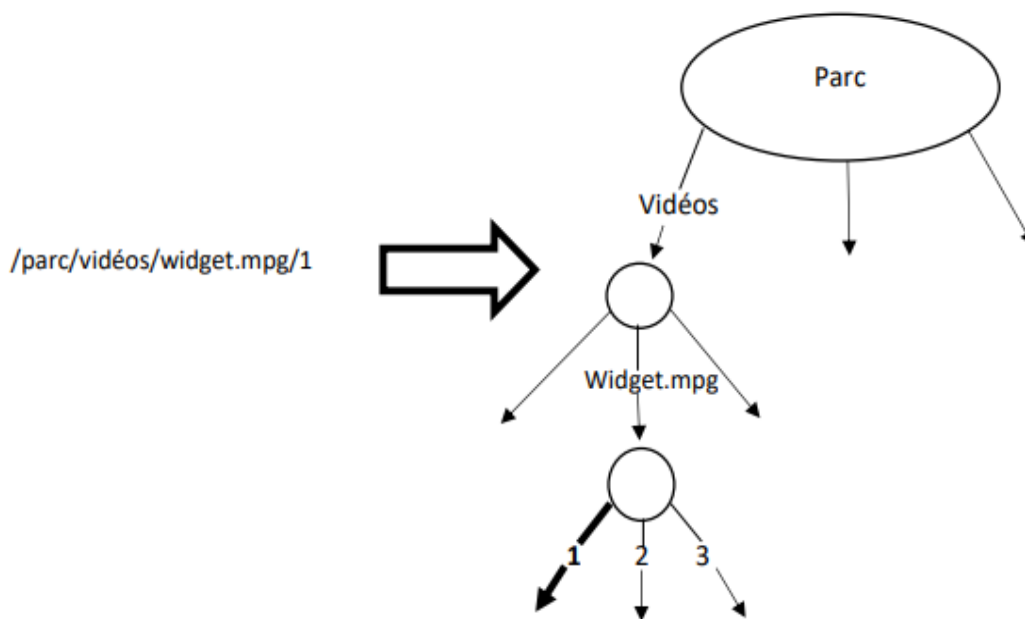


Figure 5 : Exemple sur la Communication en NDN

Les paquets d'intérêt sont acheminés vers les données producteurs basés sur les noms transportés dans les paquets d'intérêt, et les paquets de données sont renvoyés en fonction de l'information d'état configurées par les intérêts à chaque saut de routeur.

Les routeurs NDN conservent à la fois les intérêts et les données pendant un certain temps. Lorsque plusieurs intérêts reçoivent les mêmes données en aval, seul le premier intérêt est envoyé en amont vers la source de données. Le routeur stocke ensuite l'intérêt dans la table PIT, où chaque entrée contient le nom de l'intérêt et un ensemble d'interfaces à partir desquelles les intérêts correspondants ont été reçus. Lorsque le paquet de données arrive, le routeur trouve l'entrée PIT correspondante et transmet les données à toutes les interfaces répertoriées dans l'entrée PIT.

Le routeur supprime ensuite l'entrée PIT correspondante et met en cache les données dans le contenu Store, qui est essentiellement la mémoire tampon du routeur soumis à une politique de remplacement du cache. Les données prennent exactement le même chemin que l'Intérêt qui l'a sollicité, mais en sens inverse. Une donnée satisfait un intérêt à chaque saut, réalisant un équilibre de flux saut par saut.

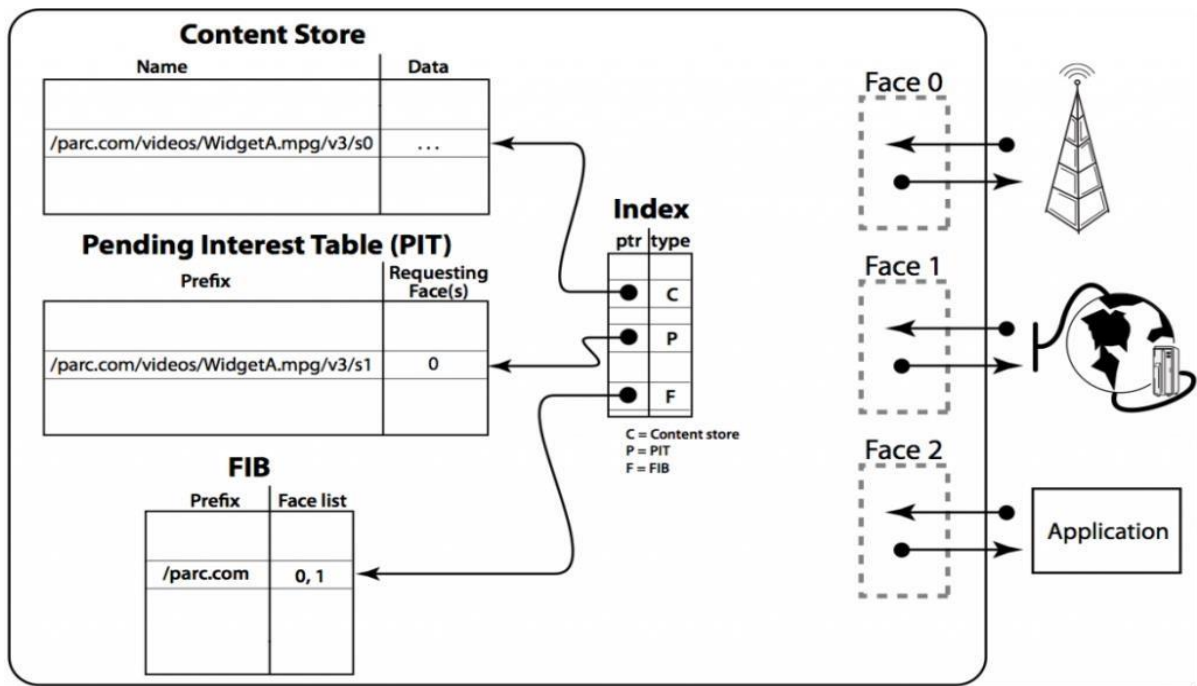


Figure 6 : Processus de transfert dans un nœud NDN [10]

### **3.2.3. Caractéristique dans NDN**

Un paquet de données NDN est significatif indépendamment de son origine ou de l'endroit où il peut être transféré, ainsi le routeur peut le mettre en cache pour satisfaire d'éventuelles demandes futures. Cela permet à NDN de prendre en charge automatiquement diverses fonctionnalités sans infrastructure supplémentaire, y compris :

- La distribution de contenu (de nombreux utilisateurs demandent la même donnée à des moments différents).
- Multidiffusion (de nombreux utilisateurs demandent les mêmes données en même temps).
- Mobilité (utilisateurs demandant des données à différents endroits).
- Réseau tolérant les retards (utilisateurs ayant des connectivités).

### **3.2.4. Les base de la mise en réseau des données Nommées**

Dans cette section, nous présenterons une étude détaillée du principal routage, mise en cache, transfert, sécurité, confidentialité et la confiance et les techniques de mobilité proposées pour le NDN ces dernières années.

#### **3.2.4.1. Nommage :**

Les noms NDN sont opaques pour le réseau, par conséquent les routeurs ne connaissent que les frontières entre les composants du nom mais pas leur signification. En conséquence, toute application peut sélectionner le schéma de nommage qui convient le mieux à ses besoins, et les schémas de nommage peuvent se développer séparément du réseau [11]. La conception NDN suppose des noms structurés hiérarchiquement, cette structure hiérarchique est utile pour les applications pour représenter les relations entre les éléments de données.

Les structures communes nécessaires pour permettre aux programmes de fonctionner sur des noms NDN peuvent être obtenues par des conventions convenues entre les producteurs de données et les consommateurs, par exemple, des conventions de nom indiquant version et segmentation. Les conventions de nom sont spécifiques aux applications et opaques aux réseaux.



Les noms n'ont pas besoin d'être globalement uniques, bien que la récupération de données globalement nécessite un certain degré d'unicité globale. Les noms destinés à la communication locale peuvent être fortement basés sur le contexte local et ne nécessitent qu'un routage local (ou une diffusion locale) pour trouver les données correspondantes.

Pour récupérer des données générées dynamiquement, les consommateurs doivent être en mesure de construire de manière déterministe le nom d'une donnée souhaitée sans avoir vu auparavant le nom ou les données. Soit un algorithme déterministe permet au producteur et au consommateur d'arriver au même nom sur la base des données disponibles pour les deux, et/ou les consommateurs peuvent récupérer des données sur la base de noms partiels. Le consommateur peut ensuite spécifier des segments ultérieurs et les demander, en utilisant une combinaison d'informations révélées par le premier paquet de données et la convention de dénomination convenue par les applications consommateur et producteur.

Le système de nommage est la pièce la plus importante de l'architecture NDN et fait toujours l'objet de recherches actives. En particulier, comment définir et allouer des noms de premier niveau reste un défi ouvert.

#### **3.2.4.2. Routage et Transmission :**

Le routage NDN se base sur les noms, ceci libère NDN des problèmes posés traditionnellement par l'architecture IP : la limitation du nombre d'adresses, Network Adresse Translation (NAT) et la mobilité des nœuds. Il n'y a pas de limitation des adresses disponibles car l'espace de noms NDN est hiérarchique et n'a aucune restriction (les noms ne sont pas dans des intervalles). Le NAT n'est pas nécessaire car les hôtes n'ont pas à communiquer leurs adresses pour échanger des données. La mobilité des nœuds n'affecte pas une communication tant qu'elle utilise des noms de données cohérents [12].

L'affectation et la gestion des adresses logiques n'est pas nécessaire dans les réseaux, ce qui est utile dans le cas des objets connectés dans une même maison par exemple.

Le protocole de routage peut fonctionner de manière similaire à celui de l'IP. Au lieu d'échanger des préfixes d'adresses IP, les routeurs NDN échangent des préfixes de noms de

données. Les annonces propagées dans le réseau grâce à ces protocoles permettent à chaque routeur de construire sa FIB. Donc les protocoles traditionnels de l'IP (Open Shortest Path First "OSPF", Border Gateway Protocol "BGP", etc.) peuvent être adaptés à NDN pour manipuler des noms au lieu de manipuler des adresses. Le principal protocole de routage NDN est Named-data Link State Routing Protocol (NLSR).

Les routeurs manipulent les noms comme une suite de composants. Pour router un intérêt, ils cherchent dans la FIB le plus long préfixe correspondant au nom (champ Name) du paquet. Par exemple : « /parc/vidéos/WidgetA.mpg » correspond à « /parc/vidéos » et « /parc » dans la FIB, mais « /parc/vidéos » est le plus long préfixe.

### **3.2.4.3. Cache :**

La mise en cache du contenu au niveau des nœuds intermédiaires, également appelée dans le stockage réseau, est fondamentalement importante pour prendre en charge le concept de base du modèle de livraison de données peer-to-peer centré sur le contenu de NDN à faible coût. La mise en cache dans NDN présente plusieurs avantages. La mise en cache des contenus produits par d'autres nœuds permet de vraiment dissocier les contenus de leurs producteurs. En outre, cela réduit les frais généraux du côté du producteur et évite un point de défaillance unique en mettant à disposition plusieurs copies du même contenu sur le réseau. Il offre des avantages élevés aux contenus dynamiques en cas de multidiffusion ou de retransmission due à la perte de paquets. En outre, il réduit considérablement la charge du réseau et la latence de diffusion des données.

La mesure des performances du cache est généralement effectuée à l'aide des quatre métriques suivantes : taux de réussite, délai de récupération du contenu (le temps total écoulé entre l'instant où une demande de contenu est générée et l'instant où elle est reçue par le consommateur), le nombre moyen de sauts parcourus (jusqu'à localiser et récupérer un contenu demandé est également mesuré pour mesurer la qualité de la distribution des contenus sur le réseau). Une autre mesure utile des performances du cache peut être la vitesse de diffusion, qui mesure le temps nécessaire pour diffuser le contenu jusqu'à la périphérie du réseau.

La politique de décision de cache détermine s'il faut ou non mettre en cache le paquet de donnée dans les nœuds intermédiaires. Pour un algorithme de mise en cache efficace, deux principales questions importantes sont où mettre en cache le contenu ? Et quel contenu remplacer en premier ? Par conséquent, nous avons largement classé les schémas de mise en cache en placement de cache (décision de placer ou non le contenu sur le réseau) et remplacement de cache (décision de mettre en cache le contenu sur le routeur).

- **Cache en Placement** : L'architecture NDN est livrée avec une stratégie de placement de cache intégrée appelée Leave Copy Everywhere (LCE). Dans LCE, chaque routeur entre le producteur (ou le fournisseur) et le consommateur mettra en cache un paquet de données. Le fait que le même élément soit mis en cache sur plusieurs nœuds dans le cadre de cette technique entraîne une redondance élevée du cache, ce qui réduit globalement la variété du contenu mis en cache du système. Laisser des copies avec probabilité (LCProb) et Laisser des copies avec probabilité uniforme (LCUniP) sont deux techniques actuelles de placement de cache pour minimiser la redondance du cache. Lors de la mise en cache du contenu sur le routeur, LCProb utilise la probabilité de mise en cache  $1/(\text{nombre de sauts})$ , tandis que LCUniP utilise la probabilité uniforme [13]. Afin d'augmenter l'utilité du contenu mis en cache dans l'ensemble du système, il est nécessaire de stocker les contenus populaires à la périphérie du réseau, ce qui minimisera la latence de téléchargement du contenu et améliorera la diversité du cache de l'ensemble du réseau. Il faut également un schéma de coordination efficace entre les routeurs pour réduire la redondance du cache et améliorer la diversité du cache.
- **Cache en Remplacement** : Une politique de remplacement de cache bien connue est Least Recently Used (LRU) (i.e. supprimer le contenu le moins récemment consulté) qui fonctionne bien et augmente la probabilité d'un accès au cache en stockant les données les plus récentes pendant un certain temps. Une autre politique importante de remplacement du cache est Least Frequently Used (LFU) où les contenus les moins fréquemment utilisés sont supprimés en premier. Le minutage de décision du cache peut être basé sur l'arrivée du contenu sur le routeur et le remplacement du

contenu. Pour de meilleures performances réseau, le contenu ne doit pas être supprimé du cache, il peut être déplacé d'un niveau en amont dans la hiérarchie du cache pour la mise en cache. Le remplacement du cache a été classifié en tant que popularité du contenu et hiérarchisation du contenu, où le contenu moins populaire et de faible priorité est remplacé en premier.

#### **3.2.4.4. Sécurité et confidentialité :**

La sécurité du contenu est l'une des exigences de base les plus importantes de tout réseau centré sur les données. Les principaux défis de la sécurité NDN sont des opérations de sécurité rentables, la gestion de la confiance et la protection de la vie privée.

- **Sécurité** : dans NDN, chaque paquet de donnée est signé avec une clé publique standard, de sorte que n'importe qui peut vérifier son authenticité. Les principales exigences d'un contenu de confiance sont : l'intégrité des données, l'authentification de l'origine et la pertinence des données (par rapport à celles demandées). Le producteur sélectionne un algorithme de signature approprié (comme RSA, DSA ou EC DSA) à partir d'un grand ensemble fixe pour minimiser la taille des données vérifiables, la latence de vérification et le coût de calcul de la génération de signature et vérification. Le processus de vérification de signature peut comporter plusieurs séries de récupération et de vérification de certificats. Par conséquent, la solution possible consiste à mettre en cache les certificats validés requis pour la vérification qui peuvent être utilisés jusqu'à leur expiration. Chaque paquet de donnée signé conserve les informations de la clé publique, nécessaires à la vérification. D'autres informations associées au paquet de donnée signé sont le résumé cryptographique de la clé publique, ou le localisateur d'empreintes digitales et de clé, qui indique l'emplacement de cette clé.
- **Confidentialité** : dans NDN, il est nécessaire de chiffrer à la fois le contenu et son nom pour assurer la confidentialité. Dans l'Internet actuel, grâce aux adresses IP et à la charge utile, il est facile de deviner qui consomme quel contenu. Dans NDN, grâce à la

dénomination et à la mise en cache, il est facile de suivre "quelles données sont demandées", mais sans adresse de destination, il est très difficile de savoir "qui a demandé le contenu" (à moins qu'il n'y ait une connexion directe). Les routeurs NDN stockent les requêtes et sont vulnérables à un attaquant. Par conséquent, NDN exige la confidentialité au niveau du routeur.

### **3.3. Conclusion**

Nous souhaitons donner un aperçu du Named Data Networking ce chapitre avant de mentionner son architecture et ses fonctionnalités. Par conséquent, nous comparons IP et NDN dans le chapitre suivant.

# Chapitre 04 : Évaluation et analyse des performances

## 4.1. Introduction

La recherche sur les réseaux IP et NDN dédiés nous a donné une idée que l'architecture NDN est un changement de paradigme fondamental par rapport à l'architecture Internet actuelle basée sur IP. Au lieu de transporter l'adresse IP de destination dans chaque paquet, NDN met un nom de données dans chaque paquet, un consommateur de données envoie un paquet d'intérêt dont le nom identifie les données souhaitées, et la réponse est un paquet de données contenant le nom, les données et une signature par le producteur de données d'origine. En nommant et en signant explicitement les données, NDN permet des fonctionnalités telles que la mise en cache dans le réseau, le transfert, la livraison des données.

Dans ce chapitre, la performance est étudiée à l'aide d'un simulateur réseau Network Simulator3 (NS-3) et Named Data Networking Simulator (ndnSIM). Le simulateur de réseau simule le comportement des différentes entités du réseau, vous permettant de gérer les interactions entre les différents nœuds du réseau.

## 4.2. NDN vs IP

### 4.2.1. Nommage et l'Adressage

Sur Internet, les paquets identifient l'hôte final à l'aide de l'adresse IP de destination, tandis que les paquets NDN renvoient les données à l'aide du Content Name (CN). Les routeurs IP utilisent des préfixes IP pour transférer les paquets IP et les routeurs NDN utilisent des préfixes de nom pour transférer le paquet d'intérêt. IP dépend du protocole de routage pour fournir des chemins sans boucle alors que, dans NDN, le nom du paquet de requête et ses informations d'entrée nonces suffisent pour éliminer le bouclage des paquets.

### **4.2.2. Routage et transmission**

Le transfert IP est sans état car les paquets ne sont pas enregistrés dans la table de routage. Mais, le transfert NDN est avec état car le routeur NDN conserve un enregistrement de chaque paquet d'intérêt qui les traverse, jusqu'à l'expiration de la durée de vie du paquet d'intérêt. NDN observe les performances du plan de données (c'est la partie d'un réseau qui transporte le trafic utilisateur) en continu à travers les états PIT, tandis qu'IP ne peut pas observer les performances du plan de données en raison du trafic unidirectionnel et du transfert sans état.

Les protocoles de routage internet gèrent les défaillances dans le transfert de paquets. Par conséquent, les protocoles de routage doivent maintenir des tables de routage cohérentes dans le réseau, ce qui induit une surcharge élevée dans un réseau à grande échelle, par exemple une convergence lente dans BGP et une évolutivité limitée dans OSPF. Cependant, dans NDN, les stratégies de transfert gèrent les pannes localement en sondant périodiquement les interfaces sortantes du routeur.

### **4.2.3. Contrôle de la congestion**

Sur internet, les paquets IP peuvent suivre n'importe quel chemin pour atteindre la destination, ce qui peut entraîner une congestion dans un environnement dynamique. Par conséquent, il est nécessaire de maintenir explicitement les schémas de contrôle de congestion. Alors que dans NDN, le routeur gère la congestion en régulant le taux de paquets d'intérêt en autorisant un seul paquet de données contre chaque paquet d'intérêt.

### **4.2.4. La mise en cache en réseau**

La mise en cache en internet est le stockage local de données réseau en vue de leur réutilisation, afin de réduire le temps de transfert pour les demandes futures. Dans le cas des pages web, la mise en cache statique sert simplement les objets, généralement des images, du JavaScript, des feuilles de style, tant qu'ils n'ont pas dépassé leur date d'expiration. Mais la mise en cache statique ne peut en général être utilisée que pour environ 30 % des requêtes http, qui typiquement n'incluent pas de données dynamiques à haute valeur.

Dans NDN consiste en un réseau de caches, ou les nœuds peuvent stocker les données qui transitent pour satisfaire les futures requêtes. La mémoire requise à chaque nœud, appelée Content Store, représente ainsi la majeure partie du cout de l'infrastructure.

Alors, Internet ne prend pas en charge la mise en cache des données demandées sur le réseau, tandis que NDN prend en charge la mise en cache sur le réseau des données demandées.

Nous avons analysé certains points clés de l'Internet actuel et NDN en termes de nommage, d'adressage, de routage, de transmission et de la mise en cache et les avons résumés dans le tableau suivant :

	<b>Internet</b>	<b>NDN</b>
Adressage/Nommage	Adresses IP	Données nommées
Routage/Transmission	-Préfixes IP -Contient le saut suivant unique -Ne contient rien d'autres que les informations du saut suivant	-Préfixes des noms -Contient une liste classée de plusieurs interfaces -l'entrée FIB enregistre les informations des plans de routage et de transfert
Contrôle de la congestion	Schémas explicites de contrôle de la congestion	Limite de taux FIB
Mise en cache	Non supporté	Supporté

Tableau 3 : Comparaison entre Internet actuelle et NDN

### 4.3. Évaluation des performances

Les résultats de l'évaluation des performances pour les modèles de communication NDN et IP sont présentés dans cette section. Nous commençons par décrire le cadre expérimental et les normes d'évaluation. Les résultats recueillis sont ensuite discutés.



### **4.3.1. Présentation des simulateurs NS-3/ndnSIM**

NS-3 [14] est conçu pour remplacer le Network Simulator2 (NS-2) courant populaire. Toutefois, NS-3 n'est pas une version mise à jour de NS-2.

NS-3 est un nouveau simulateur et il n'est pas rétro compatible avec NS-2. NS-3 est un simulateur réseau à évènement discrets. Il vise à remplacer son prédécesseur NS-2, écrit en C++, pour tenter de remédier à ses limites (mauvaise gestion des traces ou encore, plus gênant l'utilisation de multiples interfaces sur un nœud...). NS-3 est écrit en C++ et Python, et peut être utilisé sur les plateformes Linux/Unix, OS X(Mac).

Son développement a d'abord commencé en juillet 2006, et devait durer quatre ans, Il est financé par les instituts comme l'Université de Washington, Georgia Institute of Technologie et le Centre de l'TCSI pour la recherche sur Internet, la première version majeure publique et stable a été publiée en juin 2008.

Les développeurs de NS-3 ont décidé que l'architecture de simulation devait être remaniée complètement en partant de zéro. Dans cette optique, l'expérience tirée de NS-2 doit être associée avec les progrès des langages de programmation et du génie logiciel. L'idée de la rétrocompatibilité avec NS-2 a été abandonnée dès le départ. Cela libère NS-3 de contraintes héritées de NS-2 et permet la construction d'un simulateur qui est bien conçu depuis le début.

ndnSIM [15] est un module NS-3 qui est implémenté en tant que nouveau modèle de protocole de couche réseau, qui peut fonctionner au-dessus de tout modèle de protocole de couche liaison disponible, ainsi qu'au-dessus des protocoles de couche réseau et de couche transport. Cette flexibilité permet à ndnSIM de simuler différents scénarios de déploiement homogènes et hétérogènes.

#### **4.3.1.1. Terminologie et Abstractions**

Il est important de bien comprendre le sens des termes employés au sein du simulateur, ainsi que les abstractions qui ont été faites.

NS-3 utilise des termes largement employés dans le domaine des réseaux, mais qui peuvent avoir une signification particulière au sein du simulateur. Voici les principaux :

- **Un Nœud (Node) :** représente tout élément de réseau. La composition d'un Node peut être gérée (ajout de composants, de protocoles, d'applications).
- **Une Application :** représente un code exécuté par un utilisateur. Ce code peut être nécessaire au déroulement d'une simulation. L'échange de paquets durant une simulation nécessite par exemple la description d'une application au sein des nœuds participants (par exemple, `UdpEchoClientApplication` d'un côté et `UdpEchoServerApplication` de l'autre pour réaliser une application en mode client/serveur). NS-3 ne fait pas de distinction entre les « application système » (souvent exécutées par le noyau) et les applications des utilisateurs (exécutées dans l'user-space). Les applications peuvent ensuite être attachées à un Node.
- **Un canal de communication :** représente le lien qui relie des nœuds, ou plus exactement les `NetDevices` installés dans les nœuds. Des spécialisations de cette classe sont définies, comme par exemple `CsmaChannel` pour modéliser un lien Ethernet utilisant Carrier sense multiple access CSMA, ou encore `WifiChannel` pour modéliser un lien Wi-Fi.
- **Une interface de communication (interface réseau) :** appelée `NetDevice`, qui modélise à la fois l'équipement (carte réseau) et le pilote dont un ordinateur a besoin pour pouvoir communiquer avec d'autres. Des spécialisations de `NetDevice` sont fournies, comme par exemple `CsmaNetDevice` qui simule une carte réseau Ethernet et peut être reliée à un `CsmaChannel`, ou encore `WifiNetDevice` pour un lien à un canal de type `WifiChannel` [16].

Pour établir une connexion entre deux nœuds, il faut alors :

- Equiper chaque nœud de `NetDevice`.
- Configurer la couche protocolaire sur chaque nœud (élément non représenté qui se situe entre les applications et les `NetDevice`).

- Dans le cas d'une couche protocolaire TCP/IP, configurer les adresses MAC et IP sur chaque NetDevice.
- Créer le canal de communication correspondant.
- Connecter chaque NetDevice au canal de communication.

#### 4.3.1.2. Installation des Simulateurs NS-3/ndnSIM

Pour installer NS-3[17] nous devons avoir une interface Linux, et cela se fait installant linux sur la machine. Les étapes suivantes sont des commandes pour télécharger et installer les différents paquets utilisés dans Ubuntu :

- Les packages suivant doit être télécharger pour la version Ubuntu 20.04 :

```
sudo apt-get install build-essential sudo apt install libsqlite3-dev sudo apt install libboost-all-dev
sudo apt install libssl-dev sudo apt install pip sudo apt install git python3-setuptools castxml
sudo apt install gir1.2-goocanvas-2.0 sudo apt install gir1.2-gtk-3.0 sudo apt install
libgirepository1.0-dev sudo apt install python3-dev sudo apt install python3-gi sudo apt install
python3-gi-cairo sudo apt install python3-pip sudo apt install python3-pygraphviz sudo apt
install python3-pygccxml sudo apt install kiwi.
```

- Après télécharger les package nous allons passer à l'installation de ndnSIM, en commençant par le téléchargement des packages ndnSIM qui sont composés de trois éléments : une branche personnalisée de NS-3 qui contient quelques correctifs utiles, une bibliothèque de génération de liaisons python personnalisée, le code source du module ndnSIM

✓ *mkdir ndnSIM*

✓ *cd ndnSIM*

✓ *git clone <https://github.com/named-data-ndnSIM/ns-3-dev.git> ns-3*

✓ *git clone <https://github.com/named-data-ndnSIM/pybindgen.git> pybindgen*

✓ *git clone --recursive <https://github.com/named-data-ndnSIM/ndnSIM.git> ns-3/src/ndnSIM*

✓ *git submodule update --init*

- ndnSIM utilise une procédure de compilation standard NS-3. Les commandes suivantes sont suffisantes pour configurer et construire ndnSIM avec les liaisons python activées :

✓ *cd ns-3*

✓ *./waf configure --enable-examples*

✓ *./waf*

## 4.4. Scénarios de simulation

Dans la topologie simulée que nous avons choisie, il y a 2 réseaux locaux, reliés entre eux par des liens de transmission, dans le premier réseau il y a quatre consommateurs (c1, c2, c3, c4) et dans le second il y a quatre producteurs (p1, p2, p3, p4). Dans lequel tous les liens consommateur-routeur et producteur-routeur sont de 10 Mbqs. Comme illustré à la figure 7.

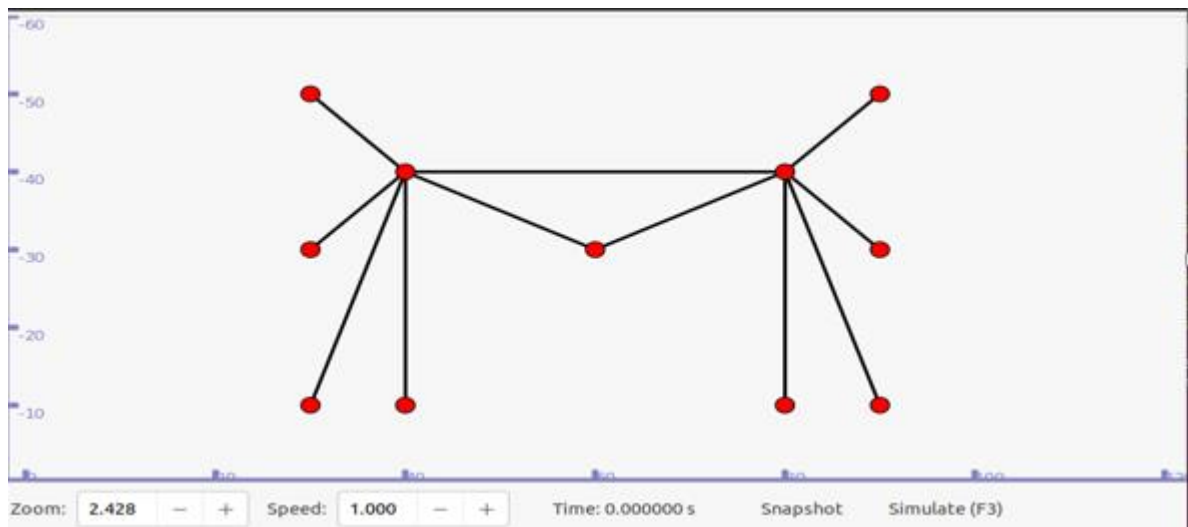


Figure 7 : Topologie d'un cas d'étude

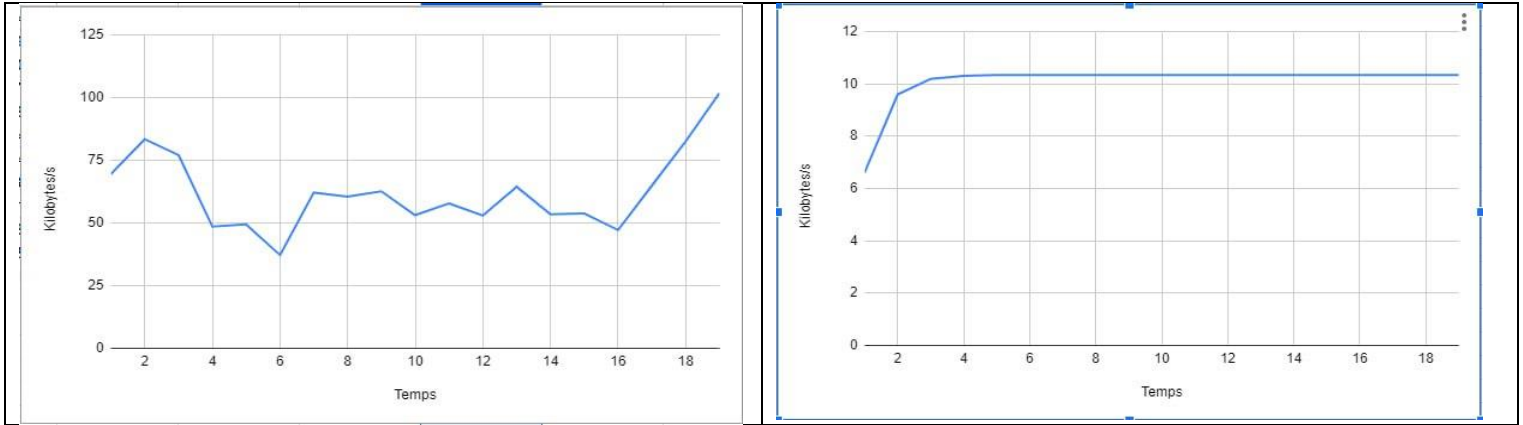
Dans cette évaluation, nous utilisons les métriques suivantes :

- **Throughput** : le débit est défini comme le nombre de bits, à l'exclusion des bits de contrôle, transmis avec succès dans un intervalle de temps.
- **Delay** : le délai peut être défini de plusieurs manières : la période nécessaire pour créer des services et la période de réception ou d'envoi d'informations.
- **Packet loss** : l'abandon de paquets ou la perte de paquets fait référence au pourcentage de paquets qui ne sont pas livrés à leur destination.

## 4.5. Résultats de la Simulation

Dans cette sous-section, nous discutons des résultats après avoir réalisé deux scénarios. Dans le premier scénario NDN, 100 paquets utilitaires par seconde avec une taille de charge pratique de 1024. Il y a quatre producteurs et quatre consommateurs. De tous les nœuds, le nœud consommateur (c1) est simulé. Dans la deuxième situation, le même nœud (c1) utilise 10 paquets. Le temps de simulation est de 20 secondes. Nous simulons et calculons trois paramètres importants : Throughput, Packet loss et Delay.

- **Throughput** : Pour le scénario NDN, nous calculons le débit pour chaque nœud en utilisant la classe `ndn::L3RateTracer`. La sortie peut être vue dans `rate-trace.txt`. Pour visualiser le texte, nous utilisons Microsoft Excel pour convertir du texte au graphique, comme montré dans le graphique NDN présenté en figure 8.



(a)

(b)

Figure 8: Throughput 100 paquets (a) / Throughput 10 paquets (b) en NDN

D'après la figure 8 ci-dessous présente Throughput (Kilobytes) en fonction de temps(s), nous remarquons que pour les deux graphes a et b qu'à des moments de temps la courbe prend une valeur élevée du débit, et c'est ce que nous expliquons que le système est capable de traiter plus de données à la fois, ce qui améliore les performances du système.

Il diminue également par instants, et cela nous explique que le réseau ne fournit pas une grande quantité de données par seconde. Cela représente de mauvaises performances réseau et peut être dû à des taux élevés de latence, perte de paquets et d'instabilité.

- Delay :** Pour le scénario NDN, nous calculons le délai pour chaque nœud en utilisant la classe `ndn :: AppDelayTracer`. La sortie du traceur est un fichier `.txt` qui comprend le paramètre de retard et les caractéristiques. Certaines des caractéristiques sont le type de retard, c'est-à-dire le retard complet (Full-Delay) et le dernier retard (Last-Delay). D'après notre compréhension, le délai complet est le temps nécessaire pour qu'un paquet arrive de l'expéditeur à la destination en calculant également la longueur du paquet lui-même, ce qui entraîne un retard plus important. Le dernier délai est le temps nécessaire pour qu'un paquet arrive de l'expéditeur à la destination sans tenir compte de sa longueur, ce qui entraîne un délai plus petit. Nous pensons que le dernier délai est plus représentatif car il n'implique pas le paquet mais

uniquement le chemin qu'il traverse. Par conséquent, nous utilisons le dernier retard dans notre mesure.

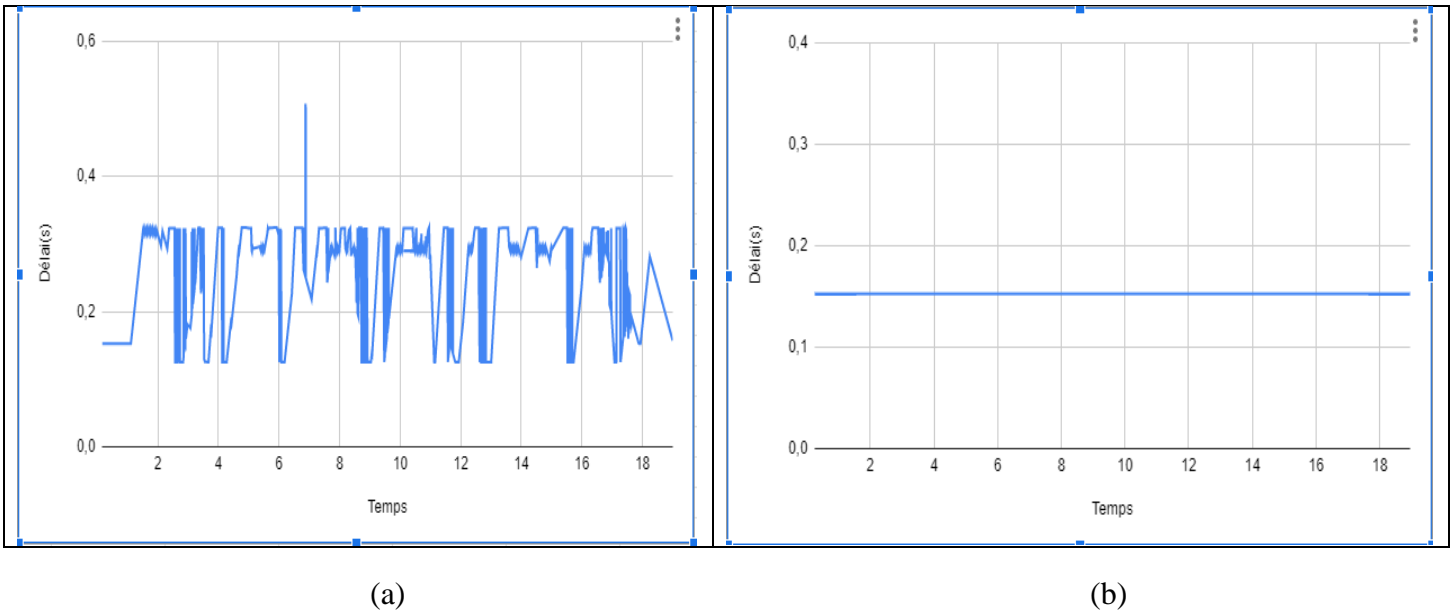


Figure 9: Delay 100 paquets (a) /Delay 10 paquets (b) en NDN

D'après la figure 9 (a) ci-dessous présente Delay (s) en fonction de temps(s), on note l'instabilité du délai, et la valeur la plus élevée qu'il atteint dans 7s est de 0.5 s. Cette instabilité est affectée soit par le délai de propagation d'intérêt, i.e. lorsqu'un nœud NDN envoie un paquet d'intérêt pour demander un élément de contenu spécifique, le paquet d'intérêt doit se propager à travers le réseau pour atteindre le producteur de contenu approprié. Le temps pris pour cette propagation peut contribuer au retard global. Soit par le délai de récupération de contenu c'est-à-dire dès qu'un paquet intéressant le producteur de contenu fournit les données demandées, le producteur génère un paquet de données et le renvoie au nœud demandeur. Les employés temporaires contribuent aux retards dans les demandes.

D'après la figure 9 (b) ci-dessous présente Delay (s) en fonction de temps(s), on remarque que le délai est similaire dans toutes les 20 secondes et cette distribution uniforme des délais suggère que les délais de livraison des paquets de données sont cohérents et prévisibles. Cela peut être le résultat d'une combinaison de facteurs, tels que : cache efficace (si les données demandées sont souvent présentes dans le cache des nœuds du réseau NDN, les délais de livraison seront

généralement courts et cohérents), faible congestion (lorsque le réseau NDN est peu encombré, les paquets peuvent être acheminés rapidement et sans délai significatif).

- **Packet loss** : Pour le scénario NDN, nous calculons la perte de paquets pour chaque nœud à l'aide de la classe L2Tracer. Le résultat peut être vu dans le fichier drop-trace.txt.

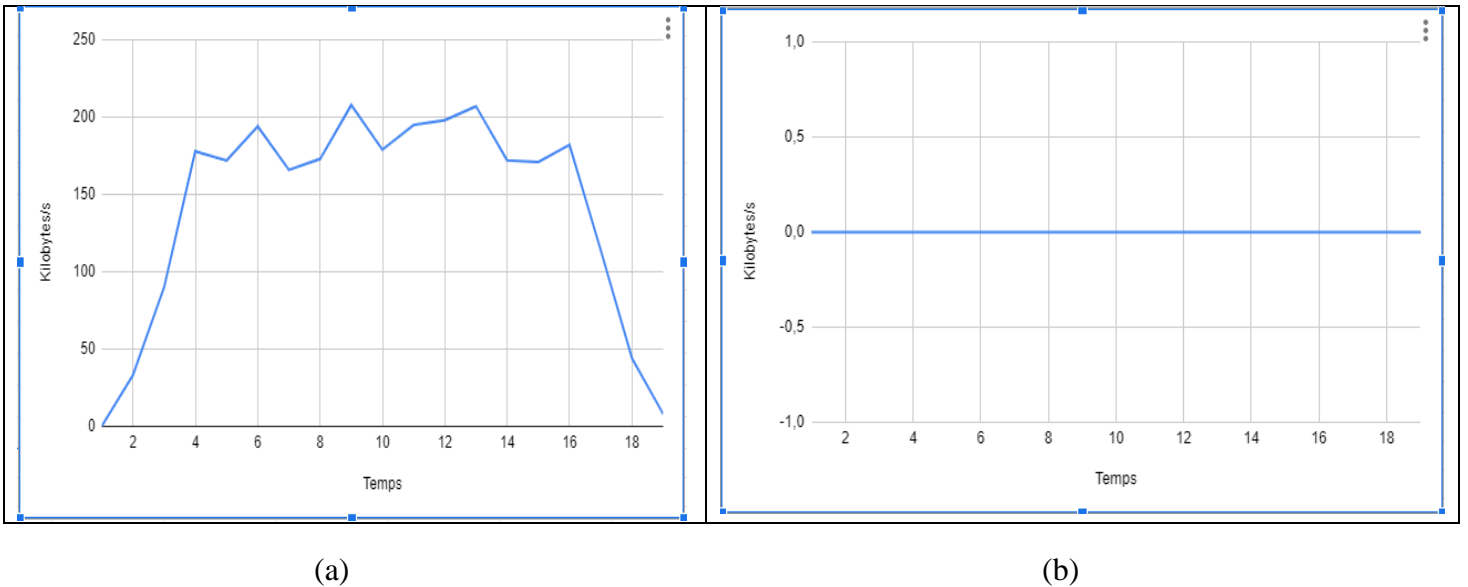


Figure 10: Packet loss 100 paquets (a) / Packet loss 10 paquets (b) en NDN

D'après la figure 10 (a) présente Delay (s) en fonction de temps(s), on observe une augmentation de la 0s vers 4s, et de la seconde 4 jusqu'à 16s elle est à son plus haut niveau, en raison de l'énorme volume de trafic qui inonde les liens. Cela peut congestionner les liens, cette congestion entraîne une perte du paquet. A partir de la 16 seconde, il commence à diminuer progressivement et cela est dû à la fonction de Content Store qui permet de réduire l'énorme volume de trafic dans les liens. Cela peut arriver en stockant le contenu requis dans le nœud le plus proche du consommateur. Dans le graphe (b) et d'après nos simulations, nous constatons que les paquets ne sont pas abandonnés en raison d'une utilisation inefficace de la bande passante.



Pour obtenir le même traitement et le même environnement pour NDN dans le scénario IP, nous créons un script de programmation pour IP et l'exécutons sur le logiciel ndnSIM, similaire à ce qui est fait dans le scénario NDN. Grâce à ce script, nous simulons et calculons trois paramètres importants : Throughput, Packet loss et Delay.

Pour réaliser cette simulation ont été appliqués deux scénarios du premier utilisant 100 paquets et du second utilisant 10 paquets, mais les résultats étaient identiques en débit, délai et perte de paquets. Tous les paquets subissent des conditions de réseau et des événements de congestion similaires, cela peut entraîner des résultats de performances similaires pour différentes tailles de paquets.

- **Throughput** : pour le scénario IP, le débit est obtenu en utilisant le logiciel Wireshark à travers ces paramètres.

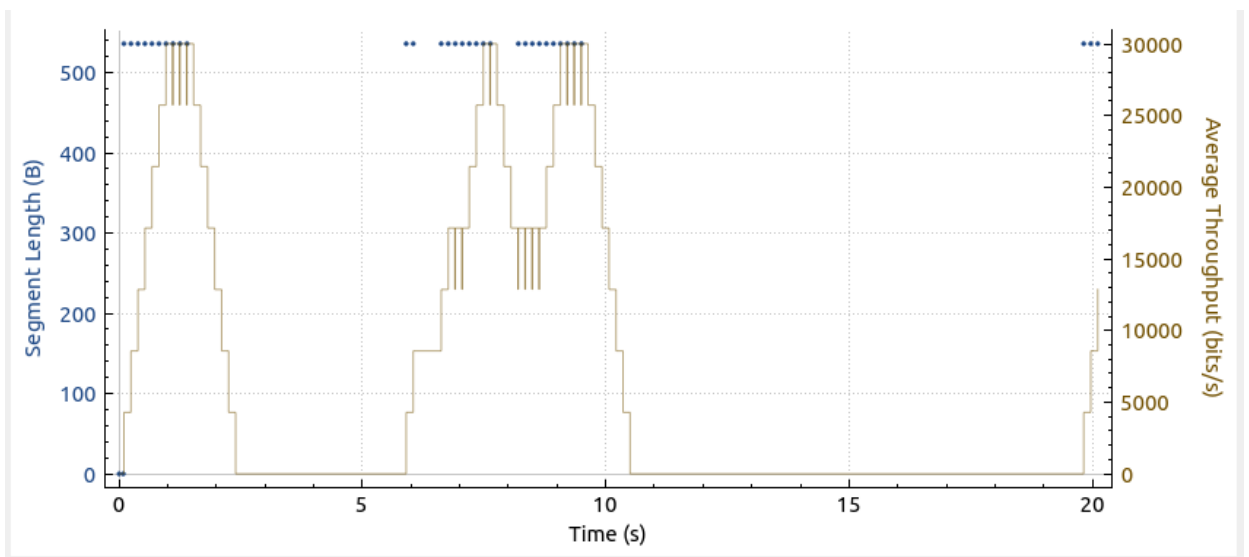


Figure 11 : Throughput scénario IP

Ce graphique (figure 11) montre la quantité de débit pendant chaque seconde de l'exécution du scénario. On remarque que le débit est relativement similaire, nous expliquons l'augmentation du débit en raison de la grande quantité de données transmises, et sa diminution en raison de la diminution de la quantité de données transmises.

- **Delay** : pour le scénario IP, nous calculons le délai en utilisant l'instruction suivant : `PointToPoint.EnablePcapAll ("nom-fichier")`; pour créer des fichiers pour chaque interface, ces fichiers ont l'extension ".pcap". Grâce au logiciel Wireshark, nous avons pu accéder à ces fichiers et en fonction du temps écoulé depuis la trame précédente dans le flux TCP, nous avons pris les données. Afin de convertir ces données en graphe nous avons utilisé Microsoft Excel.

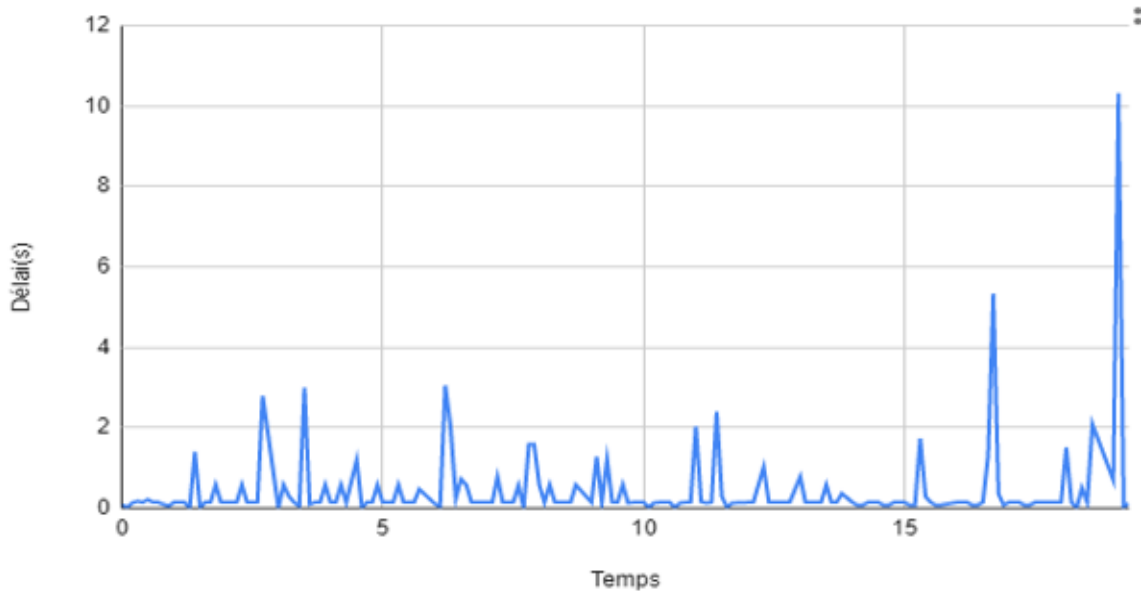


Figure 12 : Delay scénario IP

D'après la figure 12 présente Delay (s) en fonction de temps(s), On remarque un changement dans les valeurs du graphe de délai parfois décroissante et parfois croissante jusqu'à ce qu'elle atteigne sa valeur la plus élevée en 19s car la transmission des paquets de données est soit ralentie par un routeur parce que le réseau ne possède pas une bande passante suffisante, c'est-à-dire qu'il ne peut faire face au nombre de paquets envoyés ou demandés soit les paquets passent par un grand nombre de route différents avant d'atteindre leur destination.

- **Packet loss** : pour le scénario IP, la perte de paquets est obtenue en utilisant le logiciel Wireshark, qui est équipé d'un compteur de perte de paquet.

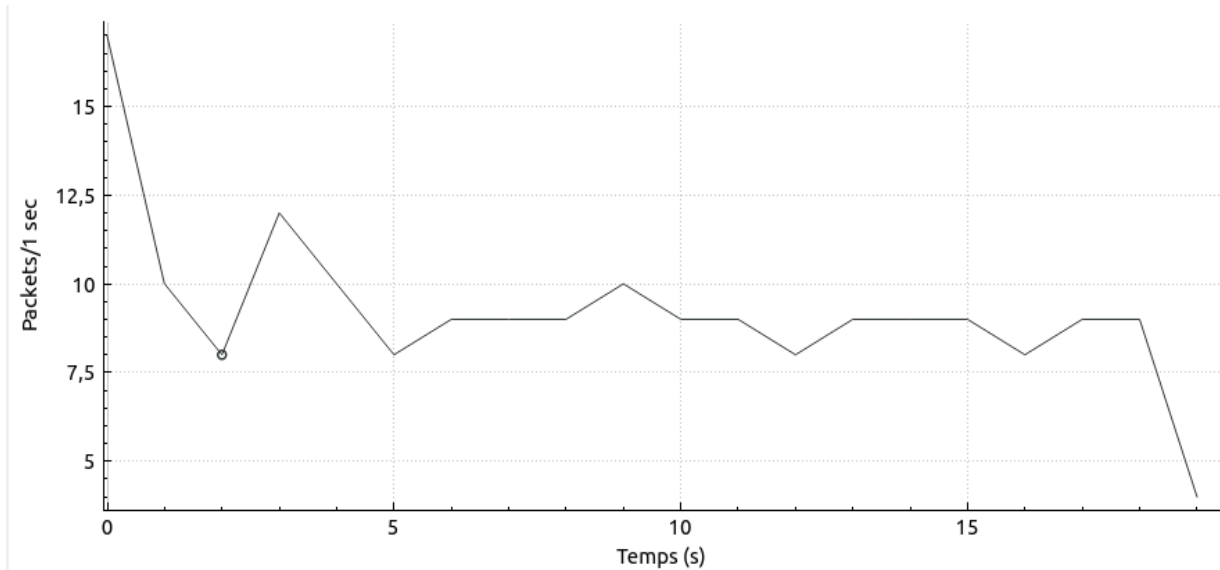


Figure 13 : Packet loss scenario IP

D'après la figure 13 présente Packet loss (s) en fonction de temps(s), le graphe nous montre une diminution relative des paquets perdu, et cela est dû à la fonction de contrôle d'erreur afin de vérifier si les données ont été reçues correctement ou non et de s'assurer que les données ont été reçues correctement, l'appareil récepteur doit renvoyer les messages de confirmation à l'appareil émetteur.

## 4.6. Évaluation

La comparaison des trois paramètres de base de NDN et IP à partir de simulations sur le réseau peut être résumée comme suit :

Les paramètres de débit dans NDN sont principalement affectés par content store et le débit de paquets d'intérêt. L'un des avantages de NDN est content store qui ressemble au mécanisme de mise en cache. Une plus grande taille de content store réduit le trafic de paquets sur le réseau car les paquets avec le même contenu ont été demandés et livrés au consommateur auparavant seront stockés dans le nœud qu'ils traversent, permettant à ce consommateur particulier ou à d'autres

consommateurs d'obtenir les données du nœud le plus proche au lieu de redemander le contenu au producteur.

Le débit dans NDN et IP est principalement influencé par l'existence du content store. Dans le système basé sur IP, il n'y a pas de content store, par conséquent, le paquet sera demandé et transféré vers et depuis le consommateur et le producteur à chaque fois, quelle que soit la fréquence de demande du paquet. En général, le débit des liaisons NDN doit être inférieur à celui des liaisons IP car NDN stocke le même contenu demandé au nœud le plus proche du côté consommateur.

D'après la simulation de données, le délai moyen dans NDN est inférieur au retard dans IP, de manière concise. L'existence content store n'a pas besoin de demander un contenu au producteur situé loin en continu, le même contenu peut être demandé au nœud le plus proche, de sorte que le délai élevé peut être réduit.

Une congestion NDN peut se produire lorsque le volume de messages d'intérêt dépasse les ressources réseau disponibles ou lorsque plusieurs intérêts sont en concurrence pour les mêmes données. Dans les scénarios encombrés, les routeurs ou les nœuds peuvent laisser tomber les préoccupations pour gérer la charge de trafic et éviter la surcharge du réseau. D'autre part, dans IP il existe des périphériques spécifiques au réseau, tels que des routeurs ou des commutateurs, qui disposent d'un espace de stockage temporaire limité pour stocker temporairement les paquets entrants. Si les mémoires tampons se remplissent et que de nouveaux paquets arrivent, les appareils devront peut-être abandonner les paquets pour faire de la place au trafic entrant le débordement de la mémoire tampon peut se produire pendant les périodes de trafic intense, de flux de données ou lorsque les périphériques réseau rencontrent des problèmes de performances.

## **4.7. Conclusion**

Dans ce chapitre, nous avons présenté toutes les données sous forme de courbes et les avons analysées. Les deux modèles ont bien fonctionné, mais NDN a excellé à la fois en débit et en délai et IP a excellé dans le contrôle de la congestion. Il est important de noter que les performances spécifiques en termes de débit, délai et perte de paquets dépendent de nombreux facteurs, tels que la mise en œuvre spécifique des protocoles, la configuration du réseau, la charge du trafic et les

conditions du réseau. Les comparaisons entre NDN et IP dans ces aspects peuvent varier selon les scénarios et les conditions spécifiques d'utilisation.

# Chapitre 5 : Conclusion Générale

## 5.1. Contributions

Ce rapport présente une simulation qui compare deux mécanismes de livraison de paquets : un mécanisme IP plus ancien qui livre des paquets de manière point à point, source à destination, et un mécanisme nouvellement proposé appelé réseau de données, qui implémente le nom comme un identification du système de réseau et utilise un mécanisme de mise en cache appelé content store. L'objectif est de comparer trois paramètres importants : le débit, la perte de paquets et le délai, entre un réseau basé sur NDN et un réseau basé sur IP. Cet objectif doit être atteint en calculant les paramètres dans NDN et IP, en comparant les graphiques dans les deux réseaux, puis en analysant et en discutant les principaux aspects affectant les paramètres.

Avoir un content store améliore le délai et le débit dans un réseau basé sur NDN. Cette fonctionnalité de NDN aide le réseau à stocker le paquet de contenu dans le nœud le plus proche du nœud consommateur, ainsi le réseau ne sera pas surchargé. De la simulation, on peut conclure qu'en moyenne, les résultats de NDN sont meilleurs que les résultats de IP. En termes de débit, NDN est également supérieur à IP. Il s'avère que le débit NDN est inférieur au débit IP, tandis que certains nœuds obtiennent les mêmes résultats qu'IP. Pendant ce temps, le paramètre de perte de paquets entraîne une perte NDN nulle dans le deuxième scénario (10 paquets) et une perte plus importante dans le premier scénario avec moins de perte IP pour les deux scénarios.

L'explication de la perte nulle est que le paquet de requête utilisé dans cette simulation n'est pas assez grand par rapport à la bande passante utilisée, alors la congestion ne s'est pas produite et la perte de paquets n'a pas été trouvée contrairement à d'autres scénarios. En général, les performances NDN sont meilleures que celles du réseau IP, cependant, le réseau basé sur IP reste le meilleur en termes de contrôle de la congestion.

## **5.2. Critique du travail**

Dans ce projet, nous avons analysé les fichiers de trace de ndn et ip pour les comparer et déterminer lequel est le meilleur en matière de contrôle de la congestion au sein du réseau. Nous avons utilisé Wireshark et Microsoft Excel pour étudier ces données et les représenter sous forme de graphiques. Ensuite, nous les avons analysés et interprétés et extrait des informations utiles.

Mais ils ne fonctionnent pas bien sur des données volumineuses. NDN ne fonctionne pas bien dans le contrôle de la congestion et le contrôle des erreurs contrairement à IP. Pour le débit, NDN fonctionne mieux qu'IP.

## **5.3. Travaux futurs et perspectives**

Cette recherche peut être améliorée sur certains points que nous considérons comme des perspectives pour notre travail :

- Créez une combinaison entre NDN et IP pour débarrasser le manque des adresses IP et incluez la propriété des noms dans le réseau à côté de des adresses IP.
- Utilisation de nouveaux mécanismes pour remplacer l'absence de couche de transport dans NDN.
- Utilisation de modèles d'apprentissage en profondeur qui permettent de prédire la congestion et de prédire le mouvement des paquets au sein du réseau pour éviter les pertes.

# Les références

- [1] Atelin, Philippe. Réseaux informatiques : notions fondamentales : normes, architecture, modèle OSI, TCP/IP, Ethernet, Wi-Fi,... Editions ENI, 2009.
- [2] Livre : Réseaux de communication / Auteur : Dr, Ali Moussaoui
- [3] Rahoual, Malek, and Patrick Siarry. Réseaux informatiques:... Editions OPHRYS.
- [4] Prieur, Benoît. "Initiation aux réseaux informatiques." (2020) : 52
- [5] Casad, Joe, and Bob Willsey. TCP/IP. CampusPress France, 1999.
- [6] Duchemin, Yann. "TCP/IP." (2000).
- [7] Syambas, Nana Rachmana, et al. "Performance Comparison of Named Data and IP-based Network-Case Study on the Indonesia Higher Education Network." *J. Commun.* 13.10 (2018): 612-617.
- [8] “Named Data Networking Architecture “, [Online] <https://named-data-.net/project/archoverview/> Consultée 27-02-2023
- [9] Zhang, Lixia, et al. "Named data networking (ndn) project." *Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC 157* (2010): 158.
- [10] Aubry, Elian, Thomas Silverston, and Isabelle Chrismen. "Implementation and evaluation of a controller-based forwarding scheme for NDN." *2017 IEEE 31st International Conference of Advanced Information Networking and Applications (AINA)*. IEEE, 2017.
- [11] Zhang, Lixia, et al. "Named data networking." *ACM SIGCOMM Computer Communication Review* 44.3 (2014): 66-73.
- [12] Saxena, D., Raychoudhury, V., Suri, N., Becker, C., & Cao, J. (2016). Named data networking: a survey. *Computer Science Review*, 19, 15-55.



[13] Gui, Yiqi, and Yongkang Chen. "A cache placement strategy based on compound popularity in named data networking." *IEEE Access* 8 (2020): 196002-196012.

[14] H. M. Bouchentouf and M. R. B. Stambouli, « Etude Des Performances des réseaux », 4G S, p. 76, 2013

[15] "ndnSIM documentation," "[Online] <https://ndnsim.net/current/getting-started.html>  
[Consultée 18-03-2023](#)

[16] Mcours, "leilclic949," Chapitre 4 Simul. Interprétation des résultats 63, p. 17, 2013, doi : 10.7868/s0044457x13060081.

[17] "NS3," "[Online] [https://www.nsnam.org/releases/ns-3-34/download/Consultee\\_10-03-2023](https://www.nsnam.org/releases/ns-3-34/download/Consultee_10-03-2023)