A Search Engine implementation for ICN

THIBAULT CHOLEZ - ERWAN KESSLER - YANN MEYER

18/05/2020 - PIDR

ICN Search Engine
PIDR

UNIVERSITÉ DE LORRAINE

Loria

TELECOM Nancy

Institut Mines-Télécom

Lorraine INP

Summary

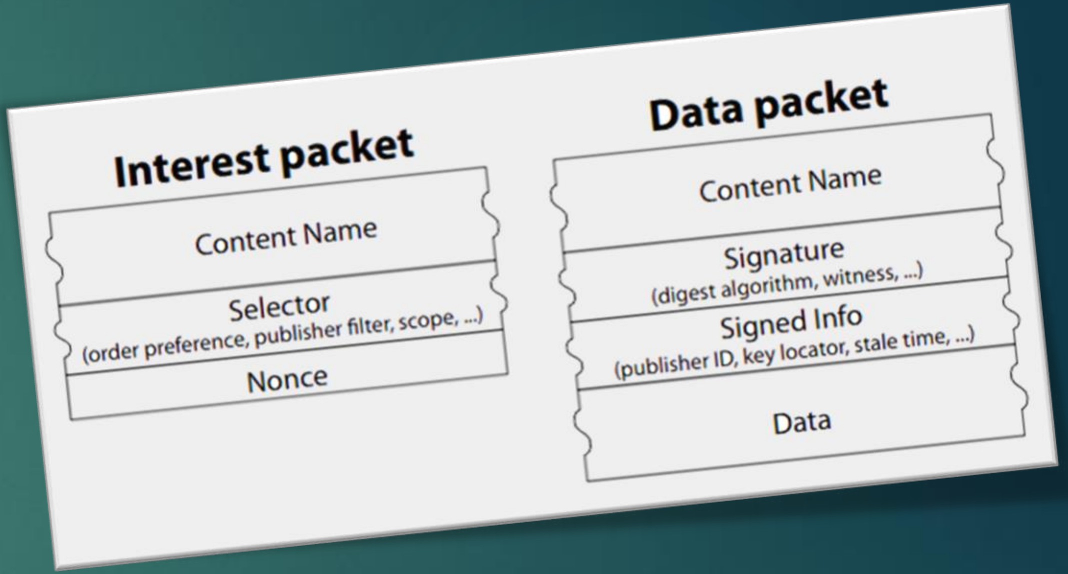
- An ICN introduction
- The Search Engine problem
- An architecture for a Search Engine Network
  - Logical structure
  - Physical structure
- Conclusion

An ICN Introduction: overview

- Information-Centric Networking (ICN) aims at introducing new communication protocols better suited for current Internet usage such as massive content broadcast and mobile use.

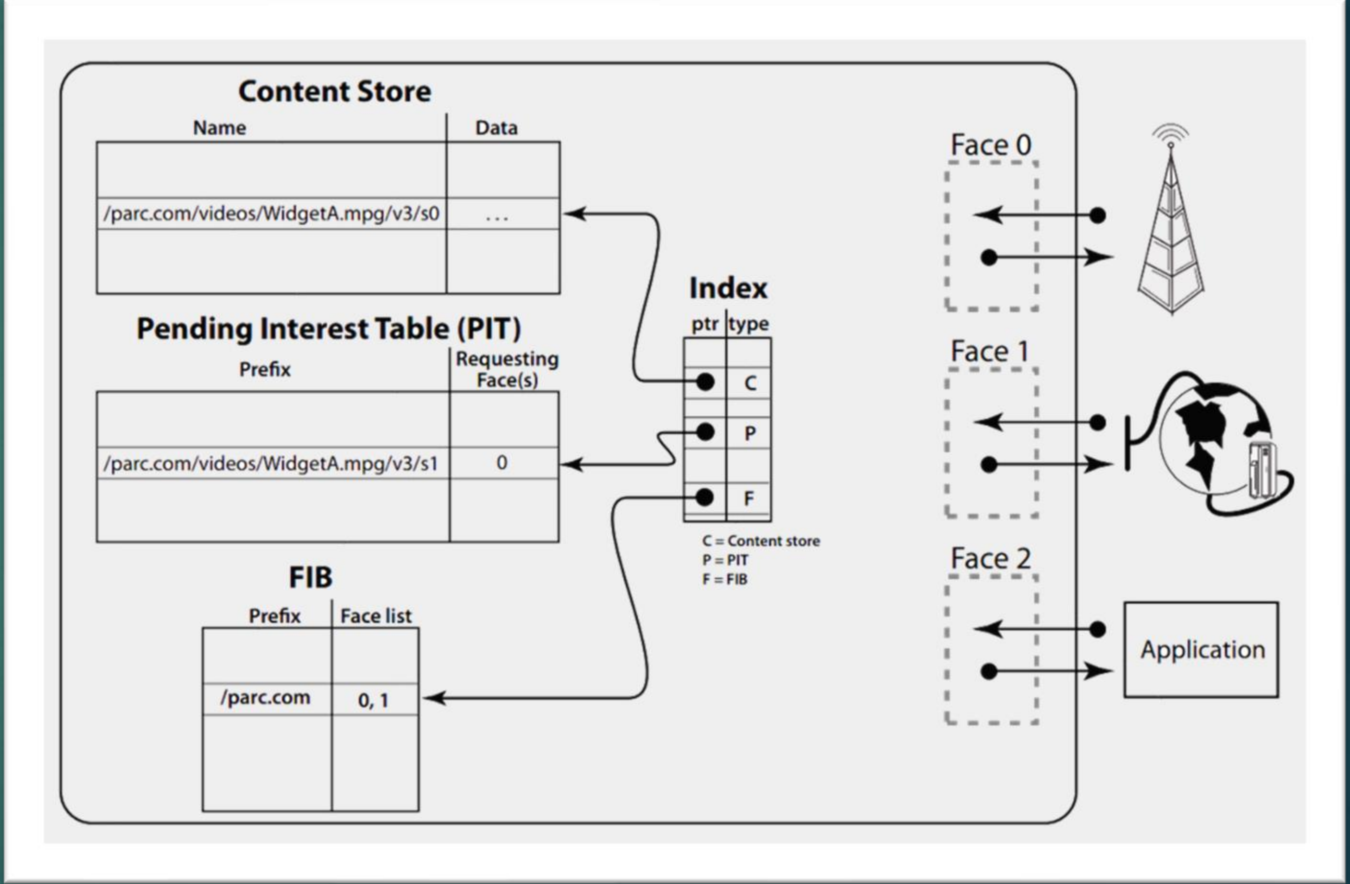
- Replaces IP addresses with named data, allowing more flexibility and improving efficiency.

- The NDN (named data networking) protocol implements this concept with two types of packets: Interests and Data.


Interest packet
Content Name
Selector
(order preference, publisher filter, scope, ...)
Nonce
Data packet
Content Name
Signature
(digest algorithm, witness, ...)
Signed Info
(publisher ID, key locator, stale time, ...)
Data

An ICN Introduction: the components

- The NDN forwarding engine model contains 3 main components :
  - The Content Store (CS): caches data to improve speed and reduce bandwidth usage
  - The Pending Interest Table (PIT): allows backtracking to the packet's emitter(s)
  - The Forwarding Information Base (FIB): get the best route to a given data



Content Store

Name	Data
/parc.com/videos/WidgetA.mpg/v3/s0	. . .

Pending Interest Table (PIT)

Prefix	Requesting Face(s)
/parc.com/videos/WidgetA.mpg/v3/s1	0

FIB

Prefix	Face list
/parc.com	0, 1

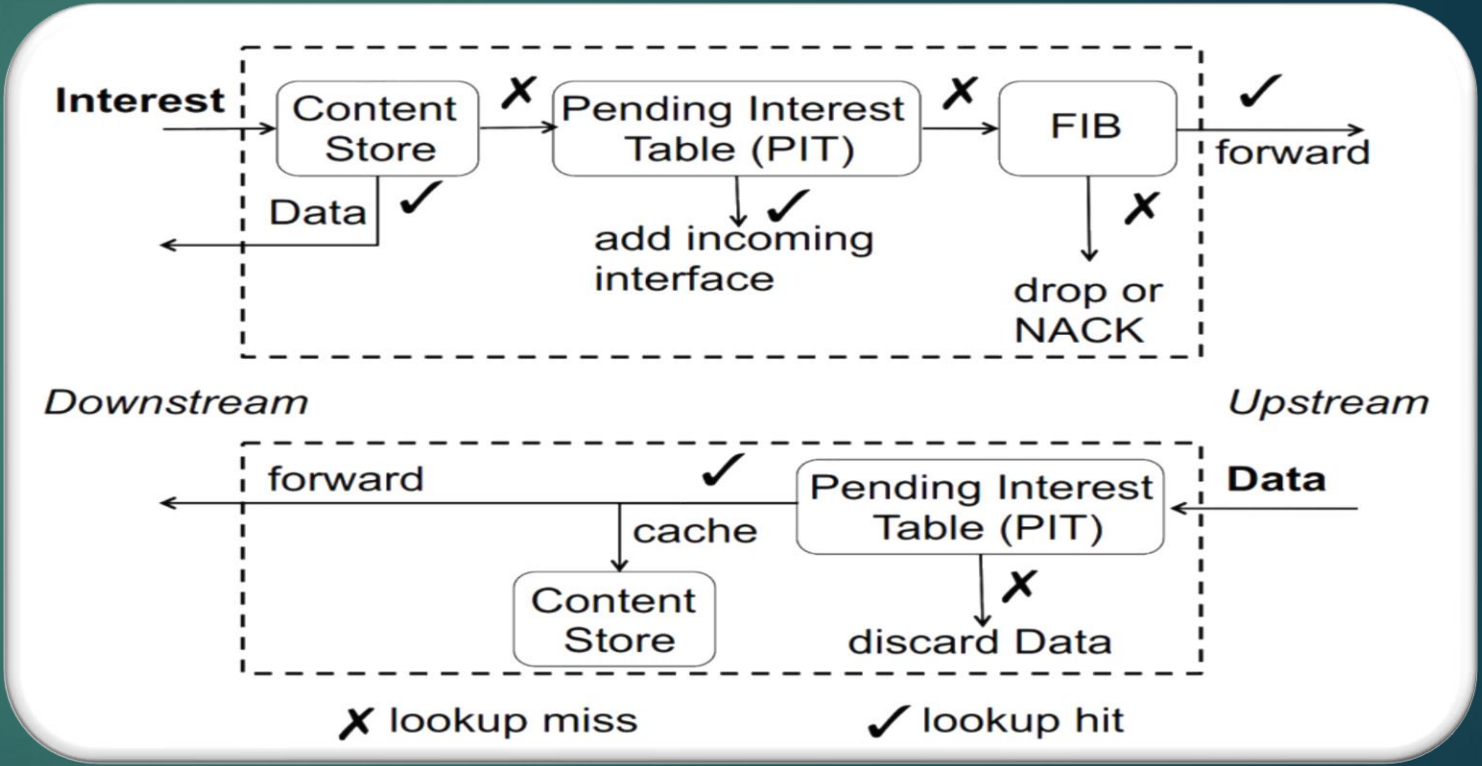
Index

ptr	type
●	C
●	P
●	F

C = Content store
P = PIT
F = FIB

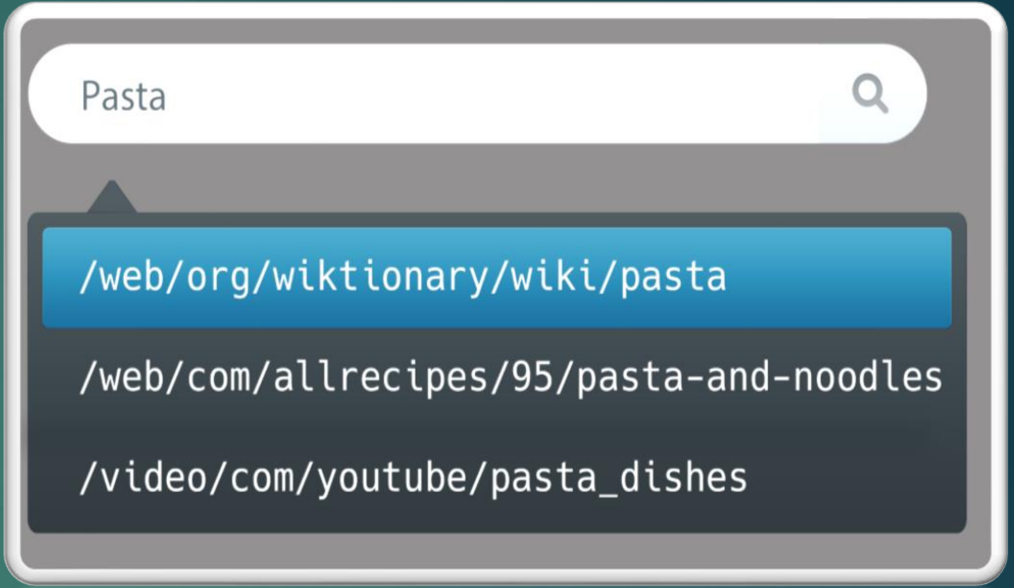
An ICN Introduction: Data retrieval

- Interest reaches a NDN router:
  - Checks CS -> if Data exists, return Data it
  - Else, check PIT : if interest already exists : append reception interface and exit
  - Else, check in FIB the best route(s) and forward or drop.
- Data reaches a NDN back
  - Check CS -> if exists, discard data else stores it
  - Check PIT -> if an entry is found, forward to all listed faces



Problem introduction: Search Engine

- To search specific names one needs a search engine , however a Search Engine in ICN needs to:
  - Be decentralized, a centralized index would defeats the purpose of the ICN.
  - Be highly scalable.
  - Be deterministic, fast and reliable.



Pasta

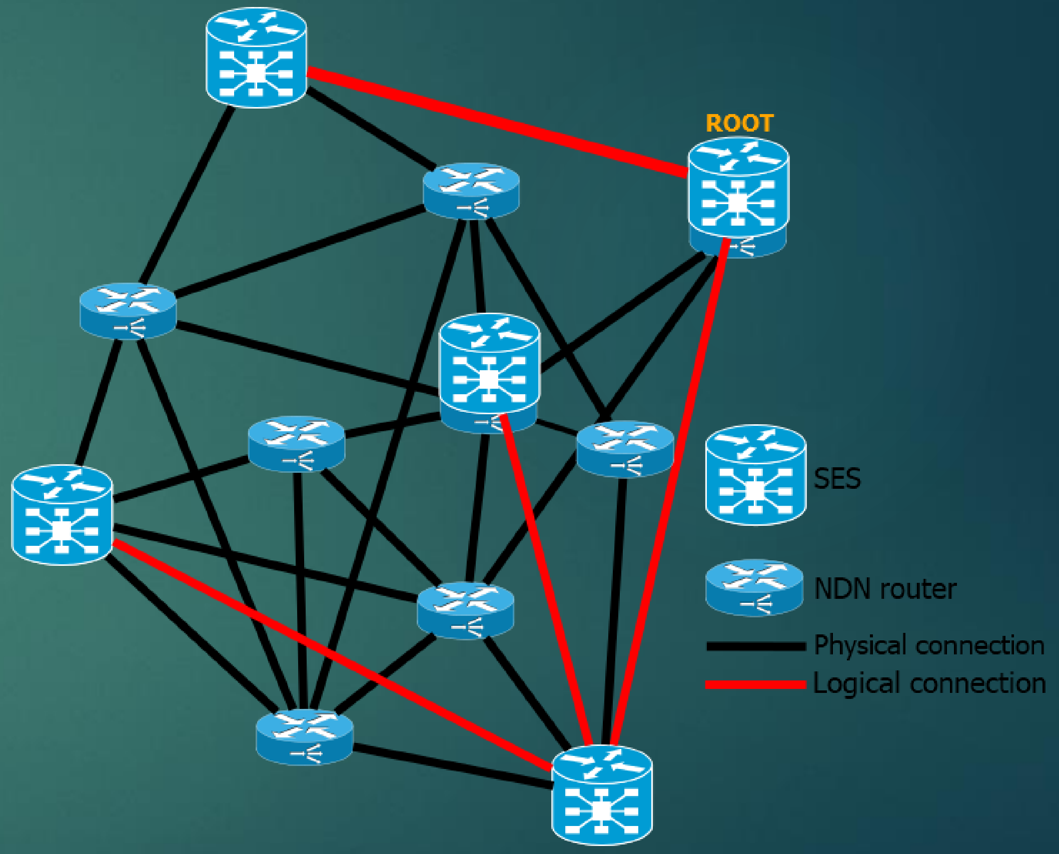
/web/org/wiktionary/wiki/pasta

/web/com/allrecipes/95/pasta-and-noodles

/video/com/youtube/pasta_dishes

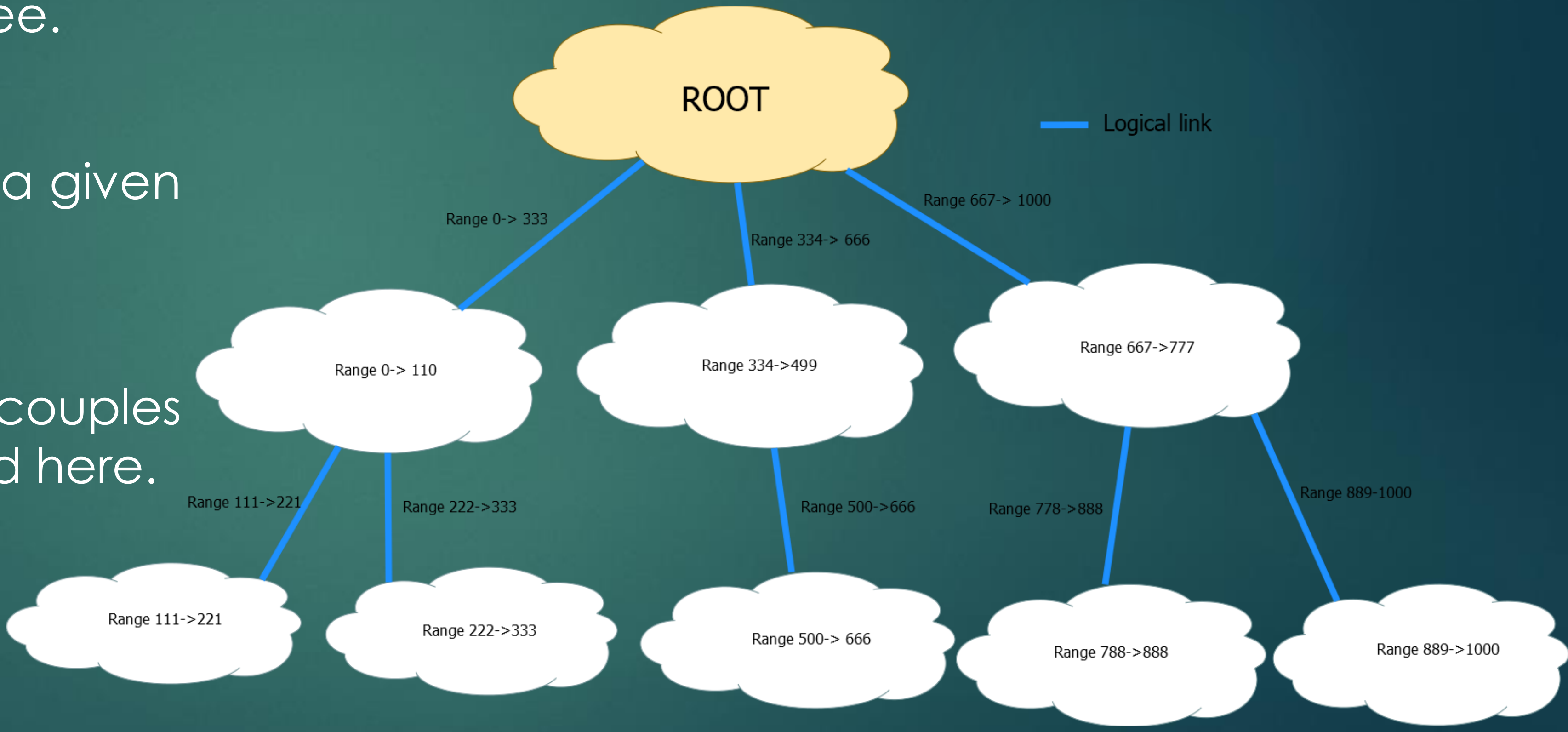
Architecture for a Search Engine Network

- Separation of the logical and physical structure (better flexibility, reliability and security, easier to deploy)
- Tree shaped Distributed Hash Table (DHT) network
- Decentralized index distributed with a hash function
- Fully deterministic
- Fast and scalable



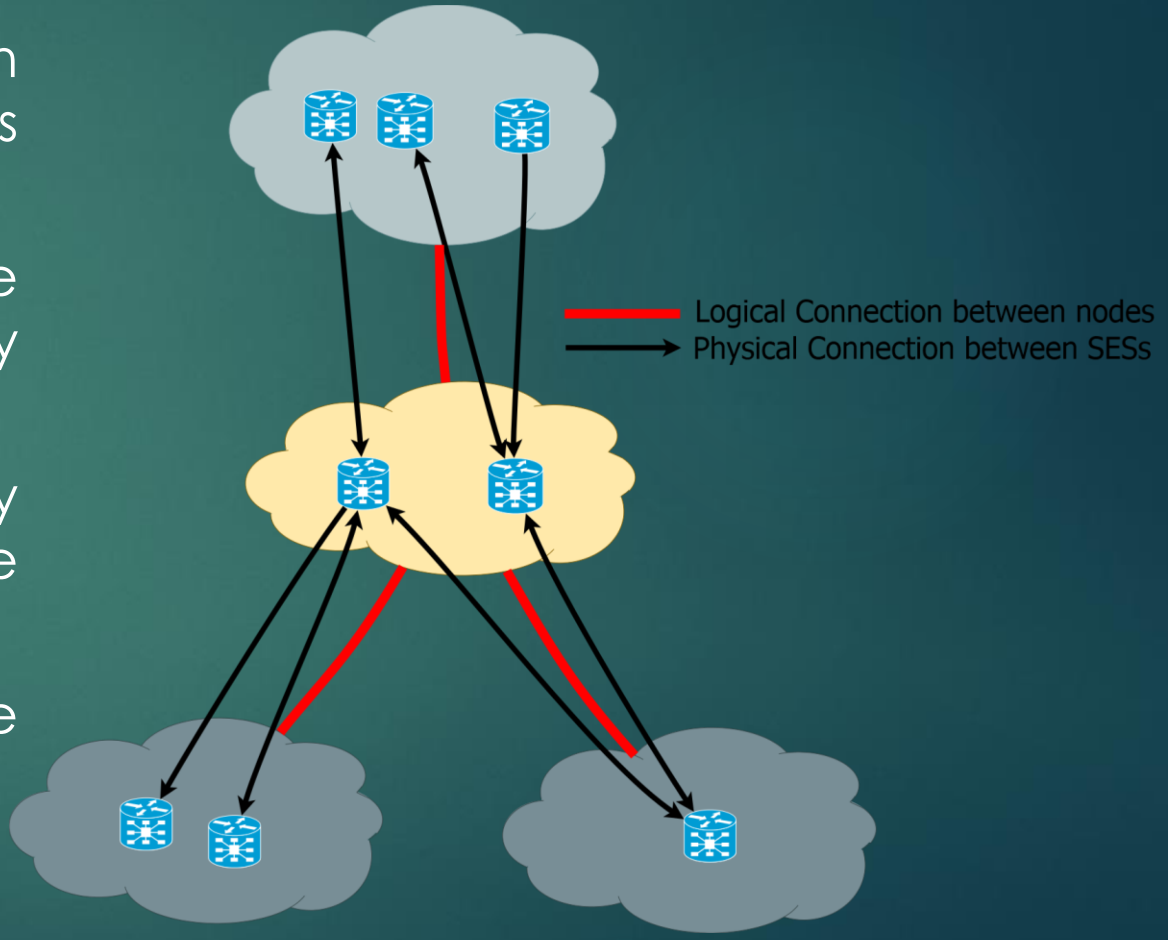
Logical network architecture

- The logical structure of the SEN can be represented as a N-ary tree.
- Each node is responsible for a given hash range.
- The range represents which couples (keyword,<datas>) are stored here.



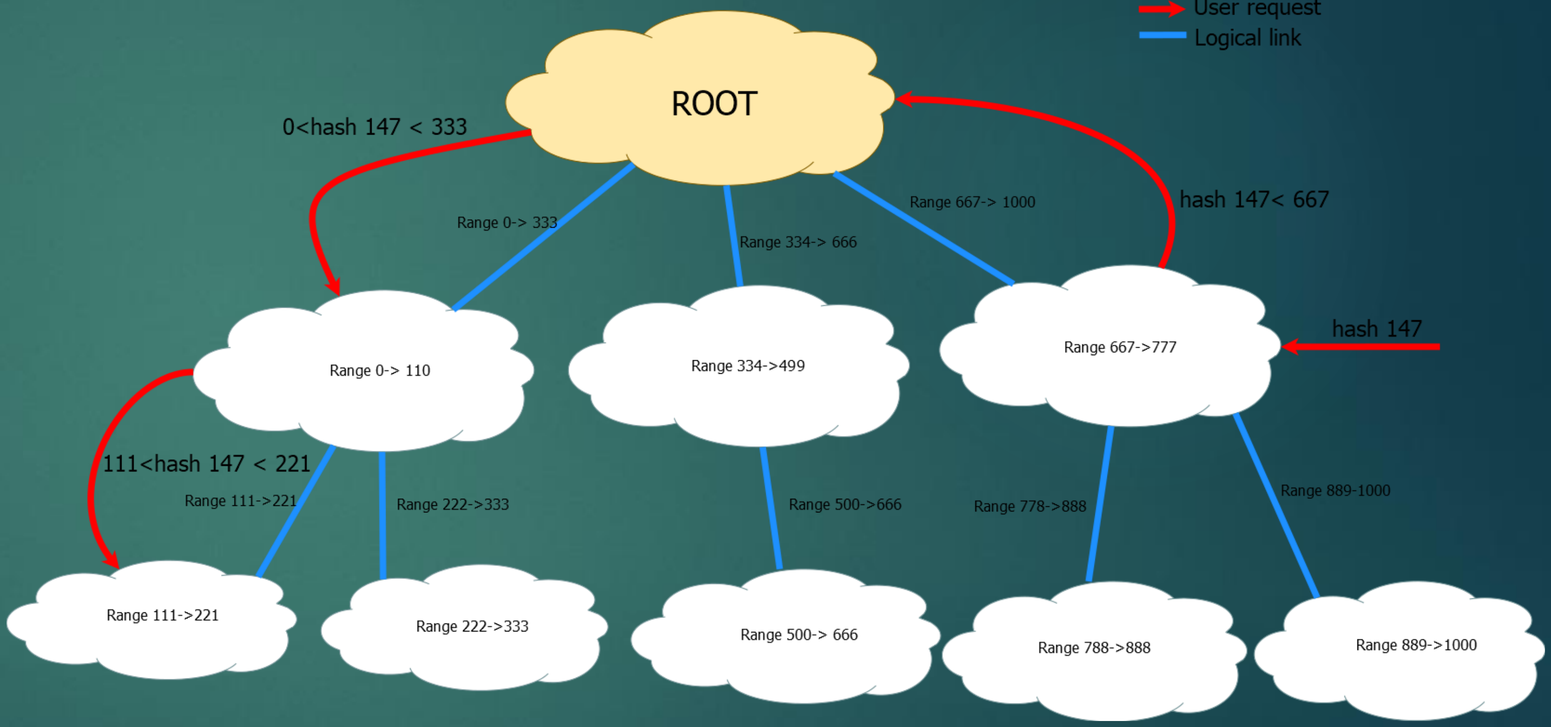
Node structure

- Each node of the tree can be broken down in a pool of Search Engine Servers (SESs).
- Each server in a node holds the same data and has the same responsibility within the network.
- The number of servers is determined by how much network traffic has to be handled.
- Cardinality differences are a result of the network balancing.

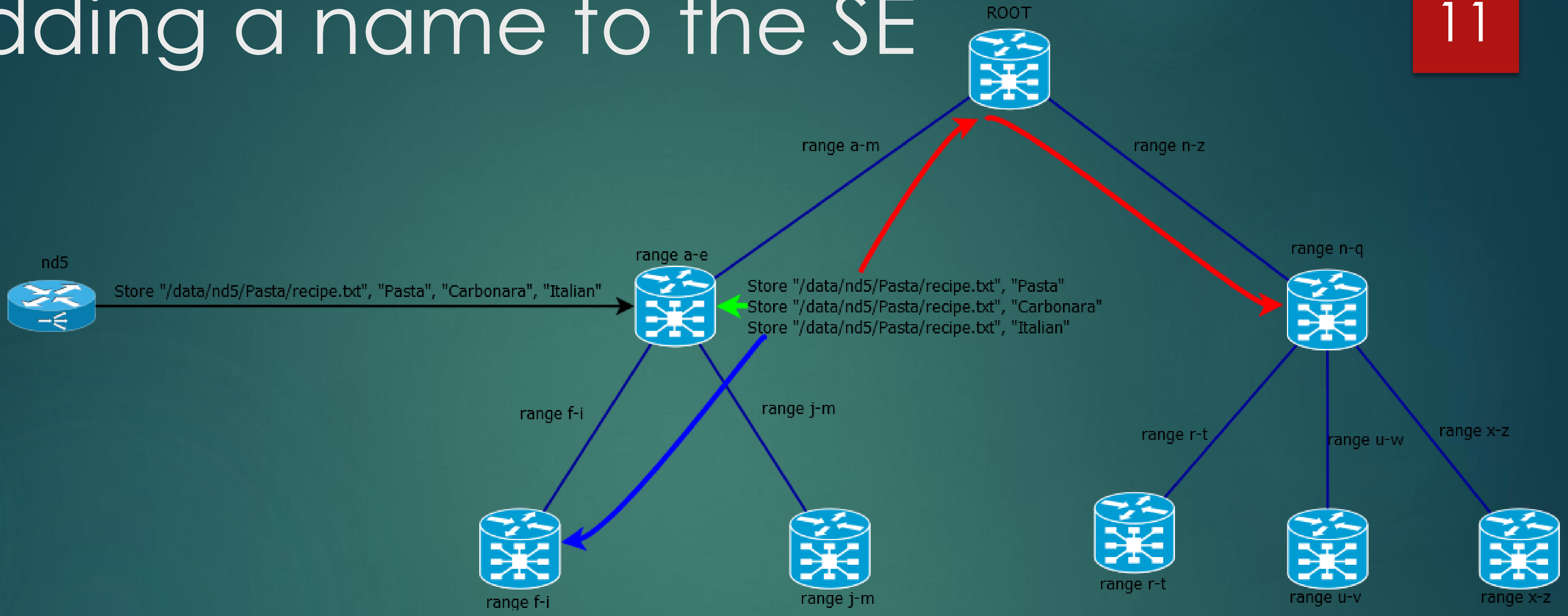

Logical Connection between nodes
Physical Connection between SESs

Query propagation within the network

- For each required keyword, a query is sent to the SEN.
- The query is sent recursively through the network by comparing the required hash to know hash ranges.
- Complexity in O(log n) in the worst case



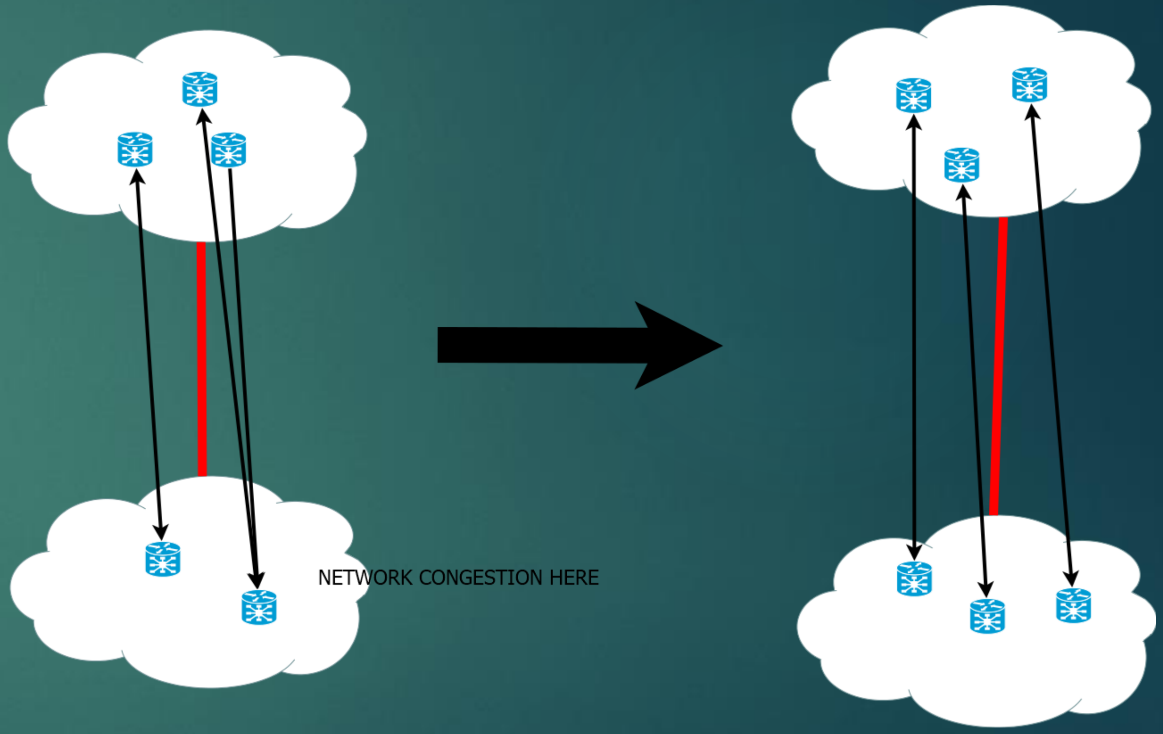
Adding a name to the SE


ROOT
range a-m
range n-z
nd5
Store "/data/nd5/Pasta/recipe.txt", "Pasta", "Carbonara", "Italian"
range a-e
Store "/data/nd5/Pasta/recipe.txt", "Pasta"
Store "/data/nd5/Pasta/recipe.txt", "Carbonara"
Store "/data/nd5/Pasta/recipe.txt", "Italian"
range n-q
range f-i
range j-m
range r-t
range u-w
range x-z
range f-i
range j-m
range r-t
range u-v
range x-z

- The producer publishes the name of the data, a set of main keywords, a set of secondary keywords and possibly an abstract.
- This data will be published once in the network for each main keyword associated.
- The routing process is analog to a query.

Network congestion management

- When a segment between 2 nodes is under heavy traffic, the node will see his size increased.
- The amount of possible connections increases, the average load decreases and the data replication increases.
- This operation is completely reversible and will be used frequently.

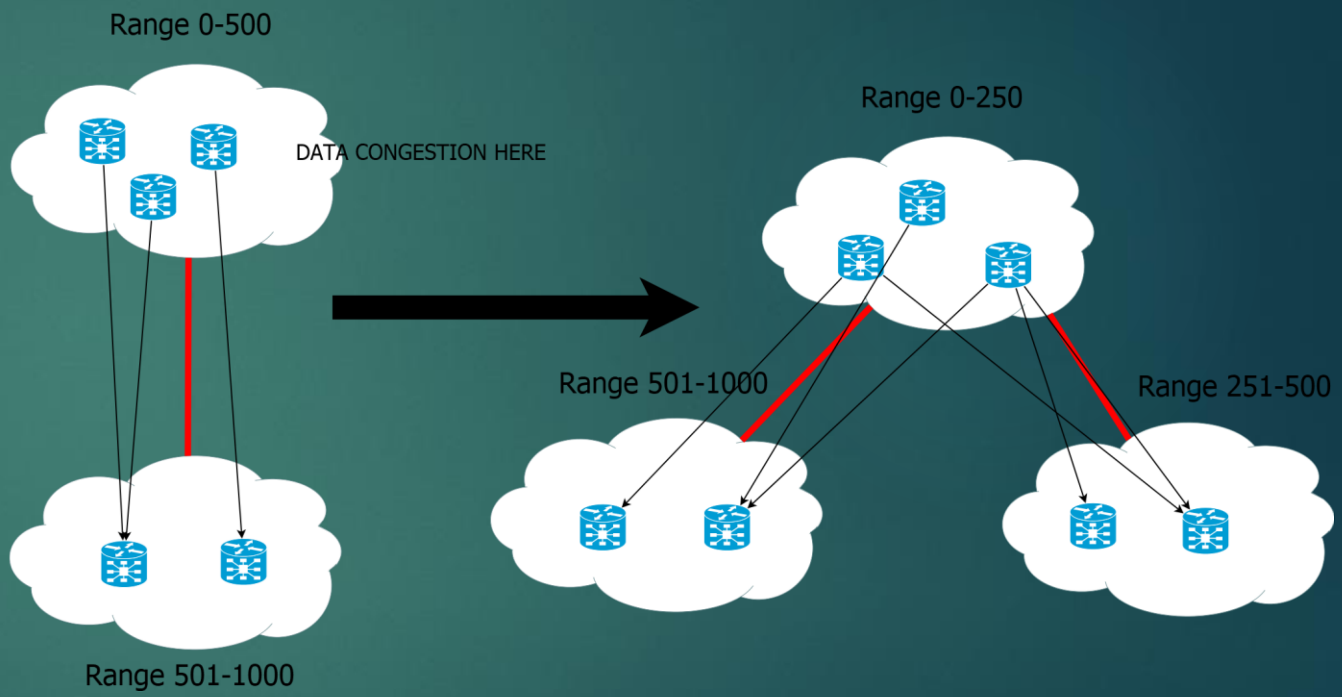


Data congestion management

- When a node is overloaded, it splits his hash range by giving a portion of it to its newly created child, lowering the load.

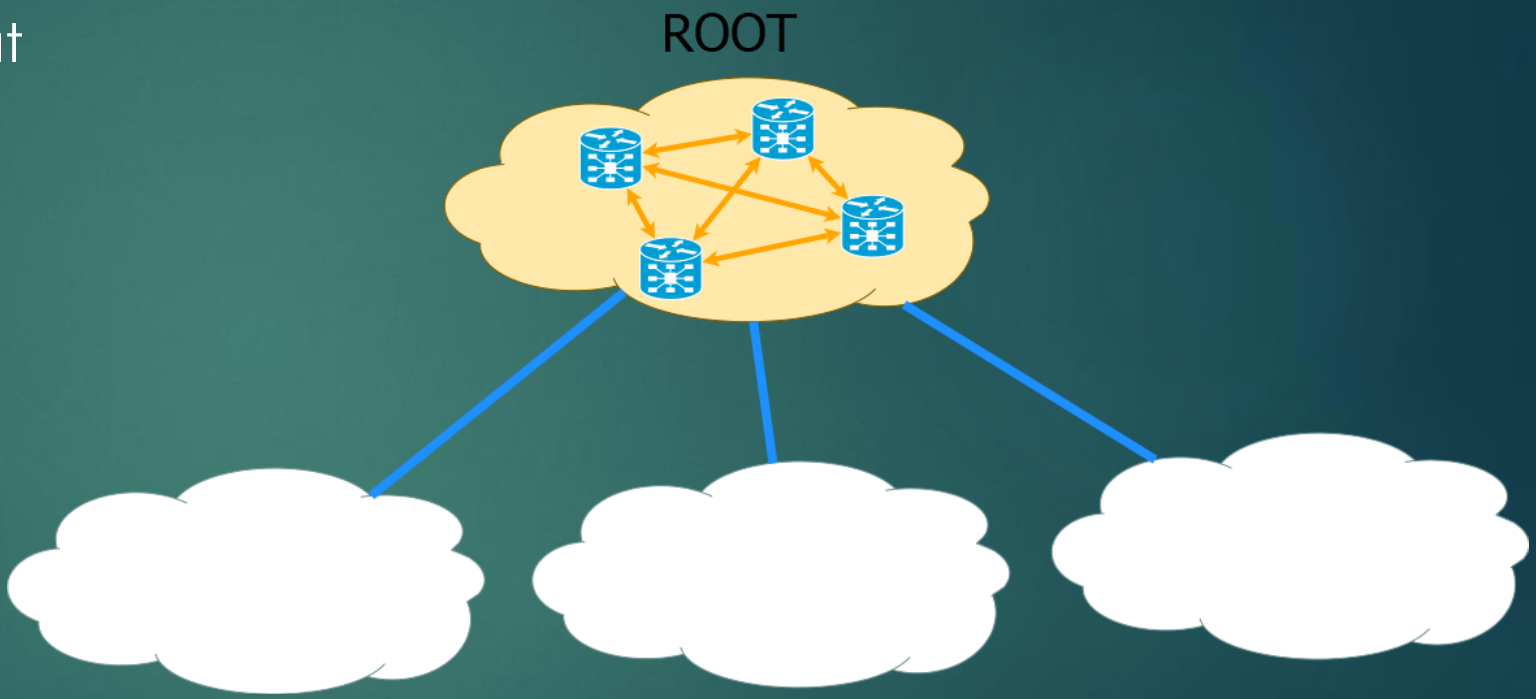
- This operation is non reversible, it has to be used with caution.

- This operation does not change addressing in the tree.



Root responsibilities

- The root is the only node that does not hold data.
- It handles the new servers distribution and their reallocation.
- It periodicly gathers the network's status.

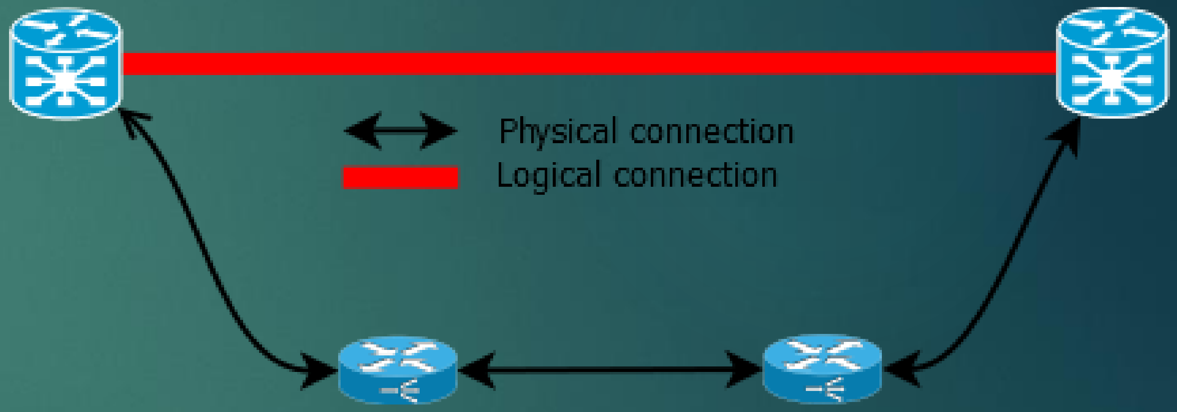


Physical network architecture

- In reality, ICN nodes are not connected to one another in any particular way.

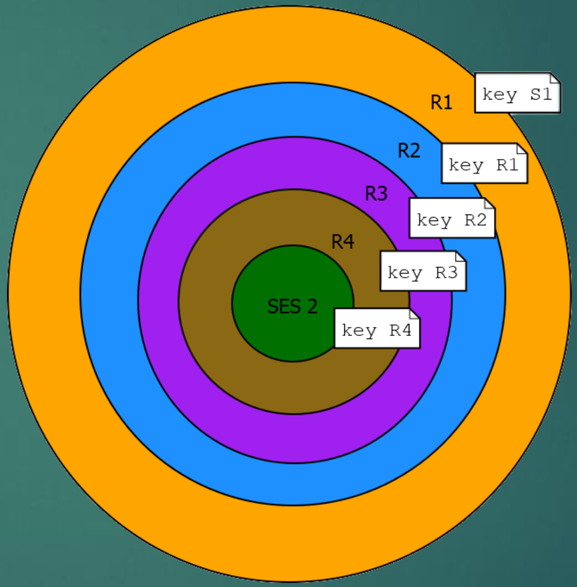
- In order for the SEN to work as intended, physical connections between logical links have to be established.

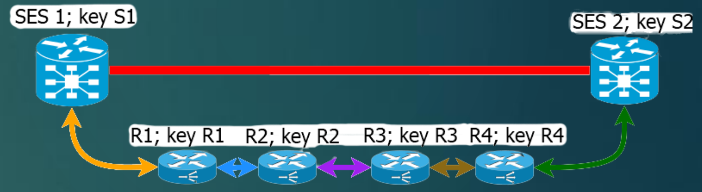
- Such connections have to be exploited to create the logical graph structure.



Establishment of a physical connection between 2 routers

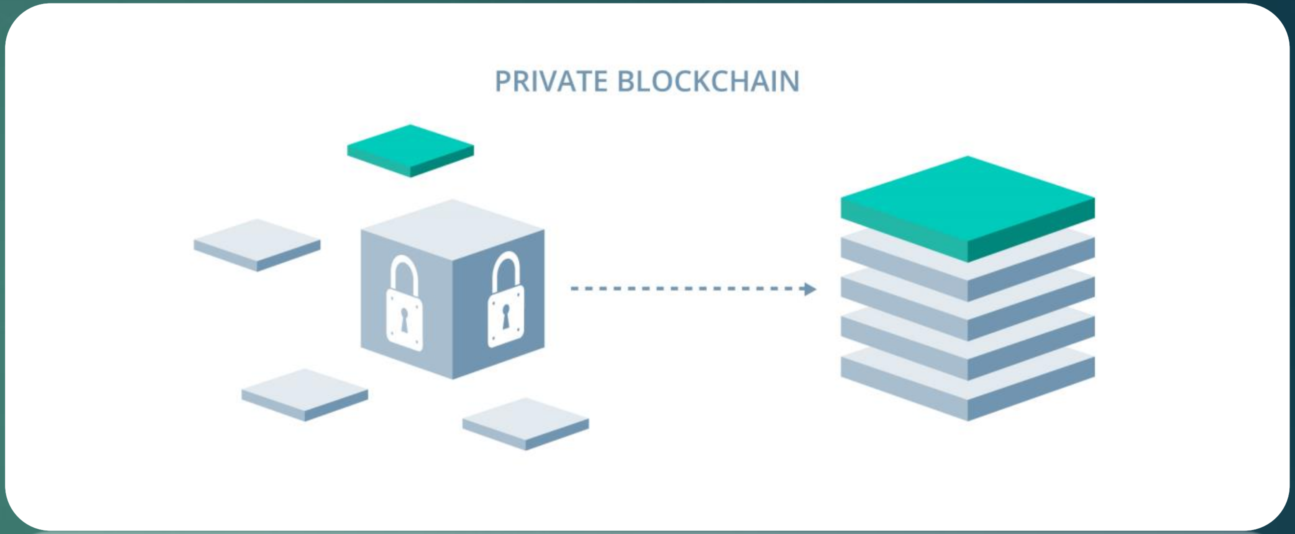
- To be able to establish a physical routing, a SES needs to know the public key of the destination.
- Send an Interest packet on a specific prefix with a cryptographic challenge.
- The onion route established is stored on the SES. It will use it to contact his logically linked SES.





Root: Private BlockChain

- The Root uses a private blockchain that executes smart contracts to trigger events such as data or network congestion.
- All the actions will then be recorded in a distributed and immutable ledger.
- The Root is thus a trustable distributed entity.



Conclusion

- We have presented a novel index scheme for keyword search in ICN, unlike existing approaches that are based on flooding or flat DHT, we use a tree shaped DHT structure which allows for a better network management and spread of the data.

- In contrast with some other design, we made sure to account for potential security vulnerabilities at the core of our implementation using zero knowledge tactics and a physical/logical separation by design.

- Although we cannot present extensive results, our initial tests have led us to be confident that the architecture is scalable, reliable and usable as a search engine.