future internet

MDPI

Article

An ICN-Based IPFS High-Availability Architecture

Ruibin Zeng [1,2], Jiali You [1,2], Yang Li [1,2,*] and Rui Han [1,2]

[1] National Network New Media Engineering Research Center, Institute of Acoustics, Chinese Academy of Sciences, No. 21, North Fourth Ring Road, Haidian District, Beijing 100190, China; zengrb@dsp.ac.cn (R.Z.); youjl@dsp.ac.cn (J.Y.); hanr@dsp.ac.cn (R.H.)

[2] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, No. 19(A), Yuquan Road, Shijingshan District, Beijing 100049, China

* Correspondence: liyang@dsp.ac.cn

Abstract: The Interplanetary File System (IPFS), a new type of P2P file system, enables people to obtain data from other peer nodes in a distributed system without the need to establish a connection with a distant server. However, IPFS suffers from low resolution efficiency and duplicate data delivery, resulting in poor system availability. The new Information-Centric Networking (ICN), on the other hand, applies the features of name resolution service and caching to achieve fast location and delivery of content. Therefore, there is a potential to optimize the availability of IPFS systems from the network layer. In this paper, we propose an ICN-based IPFS high-availability architecture, called IBIHA, which introduces enhanced nodes and information tables to manage data delivery based on the original IPFS network, and uses the algorithm of selecting high-impact nodes from the entitled network (PwRank) as the basis for deploying enhanced nodes in the network, thus achieving the effect of optimizing IPFS availability. The experimental results show that this architecture outperforms the IPFS network in terms of improving node resolution efficiency, reducing network redundant packets, and improving the rational utilization of network link resources.

Keywords: IPFS; ICN; availability

Citation: Zeng, R.; You, J.; Li, Y.; Han, R. An ICN-Based IPFS High-Availability Architecture. *Future Internet* **2022**, *14*, 122. https://doi.org/10.3390/fi14050122

Academic Editors: Ammar Muthanna and Mohammed Abo-Zahhad

Received: 6 March 2022
Accepted: 15 April 2022
Published: 19 April 2022

1. Introduction

The HTTP protocol has had a huge impact on people's Internet behavior. However, with the emergence of new network application scenarios, such as 5G and IoT, which not only bring about a surge in traffic but also make the distribution of network data more fragmented and decentralized, the browser/server (B/S) and client/server (C/S) architectures followed by HTTP have gradually revealed its drawbacks [1]. Under the existing network architecture, content service providers need to build dedicated server clusters for their own data, but such systems rely not only on systems and machines, but also on a few managers. In addition, service providers mostly use the help of third-party Content Delivery Network (CDN) to guarantee the quality of their data delivery [2]. In essence, it is a distributed deployment of servers to provide low-latency and reliable data services to users at locations close to them, and then stores their data by charging content service providers a significant price [3]. However, CDNs also face many sophisticated cyber-attacks, and adversaries can also weaponize CDN resources to launch more sophisticated attacks against end users and source servers [4]. We are entering a new era of data distribution, where people are eager to have low-latency, highly available distributed networks where data is securely distributed across the network, rather than in the hands of certain organizations or companies.

P2P networks have been considered as a research direction to replace centralized networks, and since their inception, researchers have come up with many relevant and excellent technologies. Especially with the development of distributed ledger technologies such as Bitcoin [5] and Ether [6], the attention is again directed to P2P networks, of which

The Interplanetary File System (IPFS) is a representative. IPFS is a new P2P distributed file system proposed by Juan Bennett in 2014, which incorporates many excellent technologies of traditional distributed systems and aims to replace the traditional HTTP protocol [7]. Although IPFS has been used by many researchers to address the storage challenges of blockchain [8–11], the Internet of Things [12,13], federated learning [14], and edge computing [15], and has also been used to resist censorship threats on websites, such as Wikipedia, research on IPFS's own technology is limited.

By reviewing the relevant literature, we believe that IPFS itself is actually still in the development stage, and there are still many usability challenges to be addressed. In terms of data delivery, Shen et al. studied the I/O performance of IPFS retrieval and storage from the user's perspective by comparing IPFS with HTTP and pointed out that IPFS may have performance bottlenecks in resolution and downloading [16]. Abdullah Lajam O et al. conducted experiments comparing IPFS with FTP within a private network and pointed out that private networks in IPFS do not perform as well as the C/S alternative, and IPFS needs more improvements to compete with existing C/S file sharing technologies [17]. In terms of data availability and reliability, Henningsen et al. studied the code and network topology of IPFS, and their experiments showed that about 52.19% of all nodes were private nodes located behind NATs, and that these private nodes were unable to meet long online times [18].

However, ICN, as a representative of a new type of network, can complement IPFS in many characteristics. Its efficient data distribution capabilities and directed resolution services can improve the availability of IPFS. Onur Ascigil et al. successfully solved the redundant packet delivery problem by deploying IPFS over NDNs to optimize the link resources on the user side [3]. In this paper, we improve the availability of IPFS by combining IPFS with an ICN with an Enhanced Resolution System (ENRS) to exploit the resolution and caching capabilities of the network. IPFS, like HTTP, is an application layer protocol, and many features of ICN networks are transparent to IPFS. One of the main challenges is to design the architecture of the network with hybrid deployment mechanisms. The main contributions of this paper are as follows.

- An ICN-based IPFS high-availability architecture (IBIHA) is proposed, introducing the concept of enhanced nodes and information management tables, and analyzing the advantages that can be brought.
- A traffic-based influential node selection algorithm for complex networks is designed to solve the deployment problem of augmented nodes.
- We simulate the implementation of the IBIHA architecture and the traditional IPFS network in NS3 and compare their performance gap in terms of resolution and delivery. We validate the effectiveness of the proposed algorithm on the example network and the real network dataset, respectively.

The remainder of this paper is organized as follows: Section 2 analyzes related research in IPFS availability and reviews related work in ICN; Section 3 describes the composition architecture of IBIHA and illustrates the advantages of enhanced nodes and information management tables; in Section 4, the PwRank algorithm is designed as the basis for node deployment in IPFS; in Section 5, we present the results through different perspectives and different data to conduct comparative simulation experiments and analyze the results. In Section 6, we conclude the work.

2. Related Work

2.1. Availability of IPFS

Traditional service availability is usually expressed by the percentage of time the service is available [19], which can be given by the following equation.

$$Availability = \frac{MTTF}{MTTF + MTTR} \quad (1)$$

where *MTTF* indicates the service mean time to failure, and *MTTR* indicates the service mean time to repair. However, the factors that cause service failure are very complex, such as: service node downtime, link congestion, data non-existence, and many other factors; so, defining the availability of IPFS by the above equation cannot explain the intrinsic meaning of availability well. For a clearer exposition of the availability study of IPFS, we introduce the definition of availability by On, G., et al. [19].

$$Avail_{Service} = Avail_{Data} \times Avail_{System} \tag{2}$$

$$Avail_{System} = Avail_{Node} \times Avail_{Link} \tag{3}$$

$$Avail_{Node} = Avail_{dynamic} \times Avail_{intrinsics} \tag{4}$$

We describe the meaning of these availability metrics in the background of IPFS.

$Avail_{Service}$: Service availability is composed of IPFS data availability and IPFS system availability, and only a stable and available system and data can guarantee the completion of the service.

$Avail_{Data}$: Data availability means that the user always has access to the latest data published by the service provider, regardless of the time and place. Mechanisms such as IPNS and DNSLink are provided in IPFS to ensure data availability.

$Avail_{System}$: System availability is influenced by a combination of the dynamic availability of IPFS nodes, intrinsics availability, and link availability.

$Avail_{Link}$: Link availability means that the link between the requester and the server is reachable within a tolerable delay while the IPFS node obtains data and has a relatively rich bandwidth resource.

$Avail_{Node}$: Node availability indicates whether the performance and state of the service node can support the completion of data delivery.

$Avail_{dynamic}$: Dynamic availability indicates that the IPFS nodes providing the service must have a stable online rate, with frequent offlines leading to data acquisition failures.

$Avail_{intrinsics}$: Intrinsics availability indicates the performance of the IPFS node, including: processing power, storage space, hardware configuration, etc.

2.1.1. Data Availability

Data availability is mainly expressed as data being available at the time of request, while the following reasons may exist for data unavailability in IPFS networks.

- Data is not available due to nodes going offline.
- Data retrieval information is not updated in a timely manner, resulting in the inaccessibility of the latest data.

IPFS uses the typical owner replication technology, which means that the nodes that have downloaded the data are able to store the data information in the local database. When a node publishes data with high popularity, a large number of requesting nodes in the network are able to realize the proliferation of hot data, and the higher the number of downloads increases, the more copies of data blocks in the net, and the higher the availability of data.

However, for cold data such as family photos, fewer people request the data, so the number of copies within the network is limited, and may even only be stored on the node where the data are published. When the data node fails or goes offline, then these data may face the problem of not being able to be requested. The IPFS data cache usually has a survival time, and when the survival time is exceeded, the node needs to delete the expired data unless the data are pin-operated by the node.

Therefore, IPFS networks should be persistent and permanent in order to overcome data availability problems. In traditional distributed networks, replication techniques are usually used in order to improve network availability and reliability [20], so IPFS proposes IPFS clustering to help service providers to achieve multiple copies of data management. However, Barbara Guidi et al. argue that in the case of mobile networks or other distributed

networks affected by high churn rates, it is not enough to achieve data redundancy and availability by building a private network through IPFS clusters, and it may be more convenient to implement data replication through cloud servers or resource nodes in the network [21].

Compared with the cloud server solution, the resource nodes in the network are closer to the user side, and therefore more suitable for working scenarios that provide low latency and light data transmission. However, traditional intermediate routing nodes of IP networks are only concerned with data forwarding performance and cannot support complex decisions, and new types of network devices are required to enable data replication with all the associated complex policies, such as ICN network nodes.

In addition, since the data transmission in IPFS are encrypted by the hash algorithm, different data will generate different hash values. Although the correctness of data are guaranteed in IPFS networks, this mechanism also poses some challenges for data updates in IPFS. Whenever the service provider's published data are updated, a completely different content identifier (CID) is generated, which is not perceived by users. Some users continue to use the previous CIDs for data retrieval, and the information obtained will not be real-time, so IPFS uses the URL mechanism to publish updated data through Interplanetary Name System (IPNS) [22] and Domain Name System Link (DNSLink) technologies.

IPFS maintains link information for all publishers via IPNS, which is in the format of /ipns/data provider's public key hash. The data provider stores variable, signed records under this link, and the user is then able to access the CID of the latest content published by the other party through this information. While DNSLink and IPNS work in a similar way, IPNS relies primarily on public keys, and while DNSLink uses the same readable links as DNS, DNSLink for IPFS is an incremental evolution based on the existing DNS network.

2.1.2. Link Availability

The link availability for IPFS is mainly affected by the resolution and downloading mechanism. IPFS retrieves information through a content identifier (CID), and each CID is a unique identifier generated from the data content, which means that different content necessarily leads to different generated CIDs, and the CIDs have the same length as the node IDs. The data producer uploads the data to the network and the IPFS system breaks the data into blocks and uses the Merkle-DAG [23] structure to ensure the integrity of the data. These data blocks are still stored locally, but the index information of the data will be placed on multiple nodes that have the maximum Hamming distance from the CID. The requestor of the data gets the CID of the required data block through IPNS, uses the distributed hash table to find the index information of the storage peer of the data block, and then gets the data from the peer, in which IPFS supports various transmission protocols, such as TCP, UDP, QUIC, etc. The most used protocol is still TCP. Finding the index information of CID data blocks through DHT is called resolution. Getting data from a peer is called downloading.

Shen et al. pointed out that the current IPFS use of routing protocol (Kademlia [24]) causes bottlenecks in IPFS in terms of the lookup latency of information [15]. IPFS routing is done by maintaining a distributed hash table (DHT) with 256 K-bukets, which can store up to 20 peer points of information in each K-buket, to ensure the connection of nodes to the network, and based on the ping operation to confirm whether the node is online or not. However, the lookup operation of the Kademlia algorithm takes $\lceil \log n \rceil + c$ time, so the time of resolution delay increases with the number of nodes [24]. At the same time, we must recognize that through DHTs as routing mechanisms, each access implies a series of routing steps to retrieve it. Although most DHTs define efficient routing algorithms, the traffic on the network increases by a factor proportional to the number of routing hops required to access the content and cannot guarantee a low latency for data retrieval [25].

In the traditional DHT routing mechanism, each request is considered as a separate task. Even when the same content is requested, it needs to go through the same route. These tasks can obviously be simplified with historical information, reducing the latency

and bandwidth required for the request. Therefore, Protocol Lab designed Bitswap for IPFS [26]. Bitswap is a P2P network data block exchange protocol that focuses on the rational use of historical information from the DHT routing mechanism, allowing each node to maintain a unique list of peer points.

Through the Bitswap protocol, IPFS nodes are able to filter out IPFS nodes that meet certain criteria from the history of communication and include them in their peer list. The judging criteria can be based on a combination of latency with the peer point, completion of historical requests, etc. Before data acquisition, a data acquisition request is made to the peer in the Bitswap mechanism, and if the acquisition fails, a DHT is used to obtain data from the network, but the peer can also keep an eye on the data for the requester. The Bitswap mechanism can effectively increase the success rate of data retrieval and reduce the time of data retrieval.

However, the Bitswap mechanism requires the establishment of a session window between the requesting node and the peer node, which leads to a proliferation of duplicate packets in the network by increasing the interaction information between nodes in the network, and these packets consume a large amount of bandwidth resources. In terms of data delivery, users also acquire only one copy of the data and discard the same packets reached later. Currently IPFS is using deletion signaling to reduce duplicate packets transmitted between nodes. When a requesting node receives a data block, it sends a deletion signaling to the peer that had initiated the request, and when the peer receives the deletion signaling, it updates its own record table and does not send duplicate packets to the request when it receives the corresponding data block next time. However, the introduction of deletion signaling is not ideal for the optimization of duplicate packets in IPFS [26].

In addition, when IPFS as a proxy node typically needs to serve multiple nodes, redundant packets can significantly limit the data delivery performance and cause competition for network bandwidth resources, making it particularly important to address IPFS's link availability challenges.

2.1.3. Node Availability

Henningsen et al. showed that most of the nodes in IPFS networks are private nodes. Most of these private nodes are unable to meet the node active availability and the intrinsic availability [17]. Although IPFS supports flexible joining and exiting behavior of users, it causes the storage of data sources to become extremely unstable. To tackle this problem, Protocol Lab designed a new virtual currency, Filecoin, in an attempt to increase the online rate of idle nodes from an incentive layer perspective, while contributing with their own storage resources. At present, Filecoin is still in the development stage and lacks a standard service protocol, so a huge amount of storage resources are piled up in the Filecoin mining pool and cannot be provided to the public [27].

In addition, from the perspective of content service providers [28], they cannot rely entirely on all users in the network for data storage, and if users lose interest in the service provider's historical information, the service provider may face the threat of losing useful historical information. Therefore, Protocol Lab also proposed the concept of IPFS clusters [29], which is a private IPFS network that interacts with the main IPFS network mainly through proxies and has the same IPFS characteristics inside the cluster. Through IPFS clusters, service providers are able to manage the data within the cluster, while the data in the network are driven by the user's behavior as a driver. Two consensus mechanisms are already supported by IPFS clusters: CRDT [30] and Raft [31]. Service providers can set replication factors to adjust the number of data copies in the cluster. IPFS clusters replicate content and maintain pinsets through multiple nodes to enable fast resets when one or more nodes crash, get corrupted, disappear, or fail.

Although service nodes with high performance and a high online rate are introduced to the IPFS network through IPFS clustering, the interaction between IPFS clusters and the main network relies on proxy nodes, which therefore need to face the challenge of high

throughput. In addition, for the proximity acquisition vision of the IPFS network, IPFS clustering only improves the data availability for data servers, while the performance on the user side remains inadequate.

2.2. Information-Centric Networking

In today's networks, there is more focus on the data than on where the data are located. Information-Centric Networking (ICN) is a new type of network architecture that aims to solve the problem of semantic overload in traditional IP networks. Currently, some of the more successful ICN cases include NDN [32], CCN [33], MobilityFirst [34], DONA [35], SEANET [36], etc. The core feature of these ICN networks is the separation of identifiers and locators, and therefore, further name resolution services between data IDs and NAs are needed. Typically, two name resolution service approaches are used, the name-based routing approach and the independent name resolution approach.

Among them, NDN and CCN are using the name routing-based approach, which is done by coupling routing with name resolution without the need for an additional resolution system. However, this approach completely reverses the existing data routing method, which requires changes to most of the existing network infrastructure, resulting in a non-negligible cost overhead. MF and DONA, on the other hand, both use an independent name resolution approach, which is done by pre-establishing an independent resolution system. The requestor obtains the service address information of the data source through the resolution system, and then accesses the data through normal address routing. Compared with the name routing-based approach, the independent name resolution approach can help ICN networks to make incremental deployments on existing networks with lower cost overhead.

Onur Ascigil et al. first proposed the improvement of IPFS data delivery performance by using NDN network to reduce the number of redundant packets caused by the Bitswap mechanism [18]. The routing of the NDN network mainly relies on the content base (CS), forwarding information base (FIB), and pending interest table (PIT) to record the data forwarding path, when the data are successfully retrieved, and the data are delivered to the requester along the reverse path according to the forwarding records in the PIT table. The DHT routing mechanism of IPFS is still able to provide forwarding hints to the network layer. For redundant packets, each node updates its own PIT information after receiving the first delivered data, and does not send data to the requester path again when the subsequent duplicate data reach it. However, the NDN routing method may require the existence of flood lookups due to the lack of precise localization, and the retrieval information needs to be bounded by adding a survival time.

Current independent name resolution systems have cloud-based architecture systems [34], multi-layered DHT architectures (MDHT and HSkip) [37], and determined latency (DLNR) [38] implementation schemes. While this paper focuses on combining IPFS networks with ICN network architectures with deterministic time-delayed resolution systems. As shown in Figure 1, DLNR proposes an Enhanced Name Resolution System (ENRS) by dividing the network into multiple deterministic time-delayed resolution domains with resolution delay as a constraint and deploying independent resolution nodes within the resolution domains to maintain the one-to-many relationship between identifiers and locators. Logical nesting is achieved by dividing the physical network into multi-layer coverage networks and deploying different levels of delay constraints in different coverage networks. All the resolving nodes jointly maintain a tree organization structure, and the leaf nodes of the tree represent the resolving domains with the lowest level of experimental constraints. The resolution nodes within these resolution domains can help users in the region to achieve a fast location of information data.

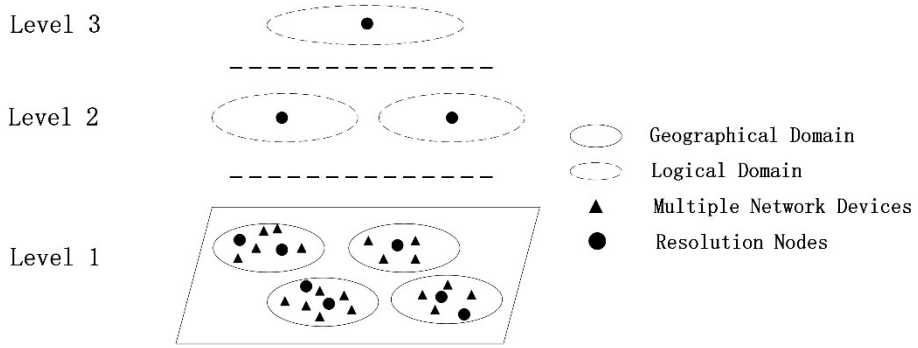


Figure 1. Determined Latency Name Resolution (DLNR) Architecture.

3. Design Overview

In this section, we first introduce the overall optimization framework of the ICN-based IPFS system, and then describe the changes of IPFS under the new framework in accordance with the existing IPFS. The overall idea of the framework is that by deploying IPFS applications on ICN nodes, these enhanced nodes can satisfy the existing IPFS interaction capabilities while using the characteristics of the ICN network. We will then provide a full description.

3.1. IBIHA Overview

Figure 2 shows an overview of the architecture of IBIHA, which relies on an ICN network with an enhanced name resolution system to enable a seamless evolution with the IP network, and includes three main types of nodes: IPFS nodes, routing nodes, and enhanced nodes. Their functions are described as follows:

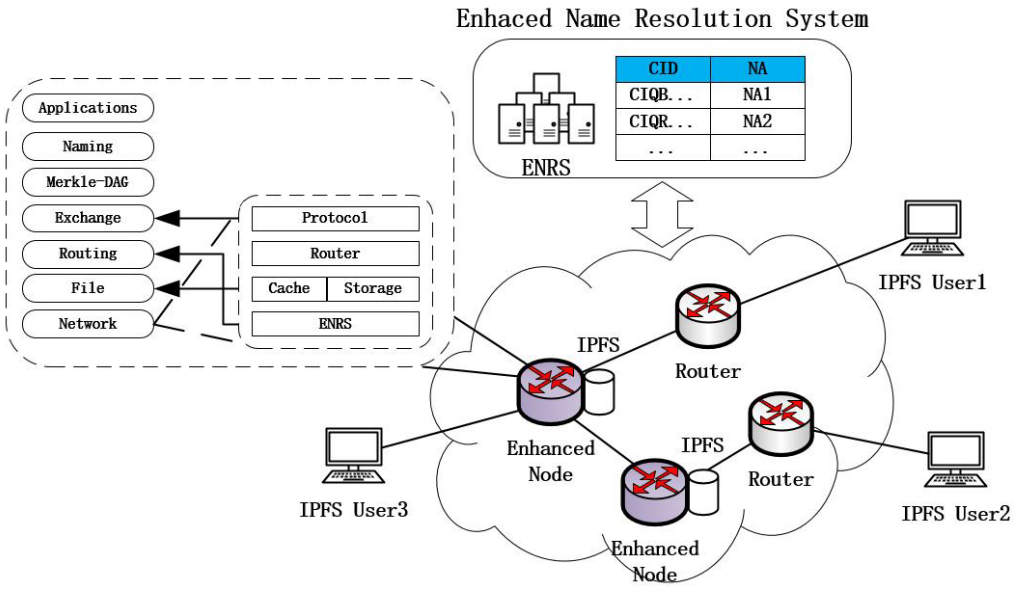


Figure 2. ICN-Based IPFS High Availability (IBIHA) Architecture.

- IPFS Node: These are many IPFS network nodes that initiate data requests from the edge of the network and enable data retrieval and data exchange through the DHT and Bitswap mechanisms.
- Routing Node: completes only the data forwarding function.
- Enhanced Node: ICN node for deploying IPFS applications. It is the peer node for all IPFS application nodes in the IPFS network, and it is also the router node in the transmission process.

3.1.1. Resolution Mechanism

As shown in Figure 3, since the network is composed of different functional nodes, IBIHA uses two methods of resolution. The first is the resolution method of DHT and Bitswap, which is supported by both IPFS nodes and augmented nodes, and this mechanism does not focus on the underlying network. Secondly, the other resolution mechanism is ENRS, which can only be supported by the enhanced nodes. In ICN, each routing node is partitioned by latency, and each region has ENRS to manage the registration and cancellation information of all nodes in that latency domain. These ENRS are guaranteeing the resolution delay of the nodes in the resolution domain, and usually guarantee that the resolution delay is at a very small delay constraint.

When an IPFS node receives a request signaling, the IPFS node that has the data sends the requested block of data to the requester, regardless of whether the requester goes through DHT or Bitswap. If not, Bitswap chooses whether to forward the data based on the TTL, while the DHT mechanism sends to the requester some nodes in its own K-bucket that are logically closest to the requested data. However, if the enhanced node does not own the data, it first checks whether the requested data exist in the time resolution domain to which it belongs, and if it does, it returns the information of the node storing the data or directly pulls the data back to complete the delivery. Otherwise, it routes the data through the application layer resolution mechanism of IPFS. the resolution in IBIHA is done by using the ENRS of ICN to enhance the IPFS node for the neighboring nodes in the time resolution domain contact, which improves the resolution success rate of the node. Meanwhile, the DHT and Bitswap mechanisms can help to realize the cross-zone search of ENRS, and in data acquisition, although ENRS may record the address information of multiple data sources, it is able to design the copy selection policy in the protocol to select the appropriate data source [39].

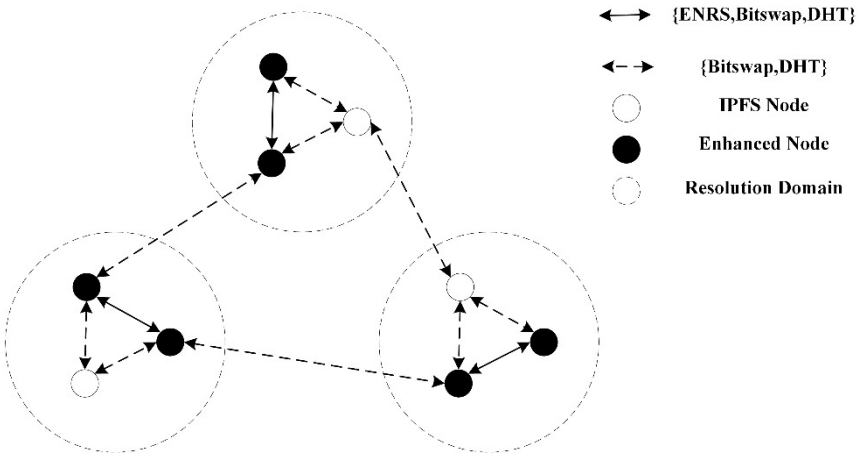


Figure 3. Resolution method in IBIHA.

3.1.2. Data Distribution

Enhanced nodes are the core components of the architecture and can bring the following advantages to IPFS:

- Nodes are high-performance network nodes with stable online rates;
- Nodes support ENRS, which can meet users' needs for nearby access;
- The network composed of nodes can provide caching and storage services, reducing the deployment costs of content service providers.

The connection between the IPFS application stack architecture and the ICN architecture is also depicted in Figure 2. It is obvious that the enhanced nodes still maintain the upper modules of the IPFS stack and that there is no heterogeneity in the structure of the data between the nodes. The main purpose is to give the IPFS network a more powerful performance by using ICN's transport protocol, cache, ENRS, and other features.

Traditional IPFS networks are built based on IP networks, where the nodes in the network have only forwarding functions. When an IPFS node initiates a data communication, it needs to complete an end-to-end interaction process. Caching is the most iconic feature of ICN, and with the corresponding ENRS it can provide more powerful resolution and delivery capabilities for IPFS networks, break the end-to-end data transmission mode, and improve the availability of the network.

The base unit of data in IPFS is the data block, so the enhanced nodes are able to take advantage of the caching capability to store the data and help subsequent requestors to achieve a fast delivery of data, because in reality there is likely to be a situation where, for high heat events, the publishers who publish them may be different, but they may use the same images or videos. Therefore, caching the core data blocks of these high-heat files can effectively enhance the quality of data delivery.

3.1.3. Information Management Table

Since the traditional IP network nodes can only complete the forwarding operation of data, the routing nodes cannot determine the current state of the requesting node. After an IPFS node sends a request information to multiple peers through the Bitswap mechanism, when the node has successfully accepted the required data block, the routing node will still send all the duplicate data blocks flowing through to the requester, thus creating a great load pressure on the user's link resources.

Compared to traditional IP network routing nodes, ICN routing nodes are more capable of implementing more powerful routing and forwarding policies. To better manage the flow-through data, IBIHA manages the flow-through request data by introducing an information management table. The information shown in Figure 4 includes the source and destination of the signaling, the request CID, the current resolution status, and the survival time. There are three main states characterizing the resolution: success, pending, and failure. Each piece of information in the information table has a survival time. When within the survival time, the enhanced node is able to determine whether the requester succeeds in obtaining the required data block, and if it succeeds in obtaining it, no multiple deliveries are required. If the requested information is in a failed state, the enhanced node takes direct forwarding for the same request during the TTL time period until the data in the management table is deleted.

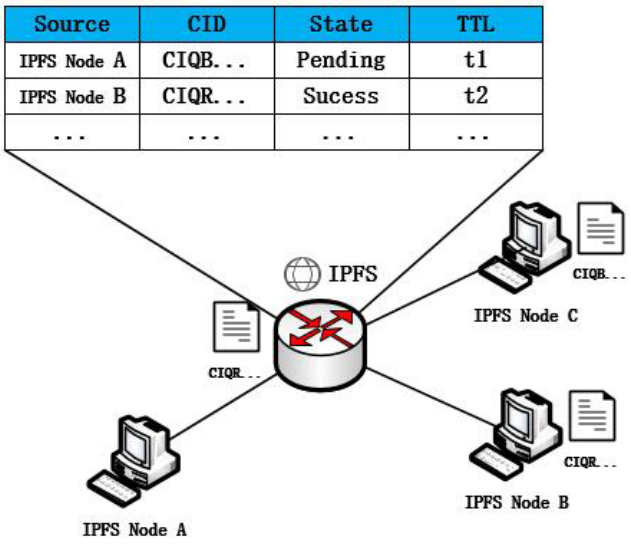


Figure 4. Scenarios for information management tables.

3.2. Enhanced Node Performance Analysis

3.2.1. Resolution Performance Analysis

First we define that each retrieval behavior of the IPFS node is independent and that the underlying DHT resolution success rate is 1. Without considering the DHT mechanism, the protocol lab gives the resolution success rate of IPFS through the Bitswap mechanism as $P_{Bitswap}$ [26]:

$$P_{Bitswap} = \frac{\gamma * l * (1 - \alpha)}{n} \tag{5}$$

where n denotes the total number of nodes in the IPFS network, γ denotes the popularity of the data blocks retrieved by the IPFS node in the network, which is related to the number of replicas r, while l denotes the number of peers of IPFS nodes, and α denotes the degree of overlap of peer connections. From the above equation, we can see that in the case of a determined network size, the resolution success rate of the Bitswap mechanism is mainly affected by both the popularity of the retrieved content and the number of nodes in the peer list.

However, the Protocol Lab analyzes the success rate formula for the Bitswap mechanism from the perspective of the whole network and does not give a detailed construction process for the above formula. Our architecture introduces the design of resolution domains, so we further analyze IPFS using a combinatorial model within a limited number of resolution domains.

With Equation (5), it is clear that the success rate of information retrieval by the Bitswap mechanism within the public network, $n \gg r$, $n \gg l$, depends on the retrieval by the DHT. However, considering a limited organizational LAN or IPFS cluster environment, e.g., enterprises, neighborhoods, schools, etc., the retrieved data of users within these local networks may be highly repetitive. ICN networks can be divided into separate resolution domains and use ENRS to improve the sensitivity of IPFS nodes to data within the local network.

Assume that there are n IPFS nodes in the IPFS network in the resolution domain and the number of replicas in the IPFS network is r. All IPFSs store l peers through the Bitswap mechanism, and L_{max} is the capacity of the peer list for Bitswap in IPFS. Since in the real case, IPFS nodes may store nodes outside the resolution domain, there exists $l \leq L_{max}$. Let B denote the event of a successful information retrieval by IPFS nodes in the resolution network through the Bitswap mechanism, whose probability is denoted as P_B.

$$P_B = 1 - P_{\overline{B}} \tag{6}$$

where $P_{\overline{B}}$ denotes the probability of a failure in information retrieval of the IPFS node through the Bitswap mechanism, and we can obtain P_B by calculating:

$$P_B = 1 - \frac{C_{n-r}^{l}}{C_n^{l}} = 1 - \prod_{i=n-r+1}^{n} (1 - \frac{l}{i}) \tag{7}$$

From the above equation we find that P_B is related to the size of the IPFS network, the number of peers of Bitswap, and the number of replicas in the network.

In turn, we analyze the resolution success rate of the enhanced node, which is set to P_M. The resolution success rate of the enhanced node consists of the success probability of the two-part resolution mechanism.

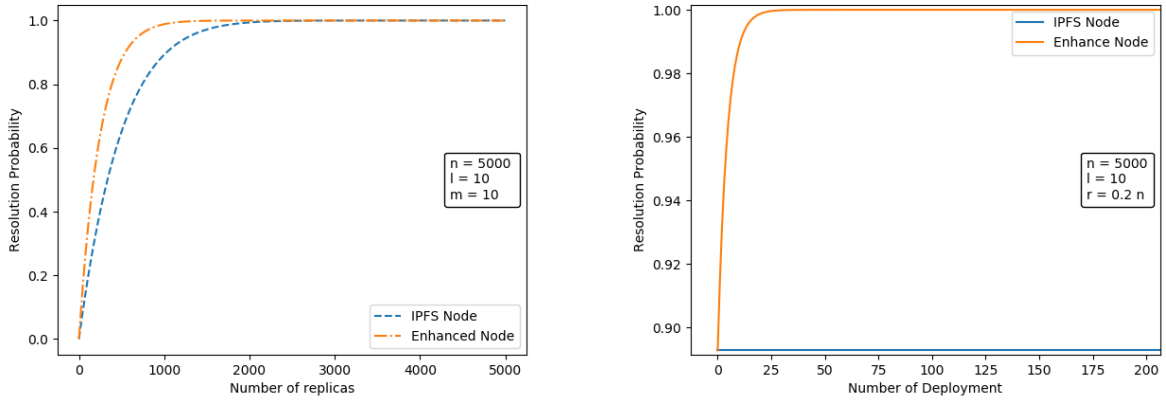
$$P_M = P_E + (1 - P_E) * P_B \tag{8}$$

where P_M denotes the resolution success rate of the enhanced node under the hybrid resolution mechanism and P_E denotes the resolution success rate of the enhanced node through the ENRS system in the resolution domain, which can be expressed as follows:

$$P_E = 1 - \prod_{i=n-r+1}^{n} (1 - \frac{m}{i}) \tag{9}$$

where m is denoted as the number of deployed enhanced nodes in the IPFS network. Thus, it is known that the resolution success rate of the enhanced nodes is related to the size of the IPFS network, the number of peers of Bitswap, the number of replicas in the network, and the number of enhanced nodes deployed in the network. In a limited local network, the number of replicas r in the network is critical to the success rate of IPFS information retrieval, and it a dynamic variable that is related to the interest of users in the network.

Figure 5a depicts the predicted values calculated by Equations (7) and (8). From the figure, it can be seen that the resolution success rate of the enhanced node has been improved more significantly compared with the traditional IPFS node for a different number of replicas. At the same time, the introduction of the resolution node makes the enhanced node more sensitive to the replicas, and the growth of the resolution success probability is more rapid.



(**a**) (**b**)

Figure 5. Comparison of the resolution probability of IPFS nodes and enhanced nodes in different scenarios. (**a**) The effect of the number of replicas on the probability of resolution; (**b**) The effect of the number of enhanced node deployments on the probability of resolution.

We also studied the effect of the number of deployed augmented nodes m on P_B and P_M, and the results are shown in Figure 5b. The resolution success rate of IPFS nodes using the Bitswap mechanism is independent of m and is therefore constant. At $m = 0$, no enhanced node exists in the network at this time, so the resolution success rate at this time is equal to the resolution success probability of the Bitswap mechanism. As the number of deployed nodes increases, only a small number of augmented nodes needs to be deployed to keep the performance of augmented nodes at a high level, and as the deployed nodes are enhanced, the resolution success rate tends to 1.

To quantify the performance growth benefits of the enhanced nodes, we characterize them by the performance improvement ratio β:

$$\beta = \frac{P_M - P_B}{1 - P_B} \tag{10}$$

By bringing Equation (8) into the above equation, it is obtained that:

$$\beta = P_E \tag{11}$$

We take β = 90%, that is, when the enhanced IPFS resolution performance in the Bitswap-based resolution performance improvement space is further optimized by 90% as

the goal, we can get P_E = 90%. Where P_E is related to the deployment of m, according to the above equation, we can express P_E as:

$$1 - \prod_{i=n-r+1}^{n} \left(1 - \frac{m}{i}\right) = 90\% \tag{12}$$

To facilitate the analysis, we define the upper and lower bounds of P_M by:

$$1 - \left(1 - \frac{m}{n-r+1}\right)^r \geq 1 - \prod_{i=n-r+1}^{n} \left(1 - \frac{m}{i}\right) \geq 1 - \left(1 - \frac{m}{n}\right)^r \tag{13}$$

The equality sign holds if and only if $r = 1$.

The above equation shows that the smaller the r, the greater and closer the approximation of the upper and lower bounds to P_E. For different retrieval data, there may be different numbers of copies within the network, but it can be seen from the above figure that both P_B and P_M show an increasing relationship with the growth of the number of copies, and when the number of replicas within the network reaches a certain level, the probability of retrieval has converged to 1. Therefore, we further constrained our goal to be to increase the probability of information retrieval from nodes while guaranteeing a small number of replicas. Therefore, we denote the number of replicas $r = 0.2n$ according to the zipf law of network data access [40].

When P_E is expressed as an upper bound formula model, an upper bound value m_1 for the number of enhanced node deployments is obtained as follows:

$$m_1 = \left(1 - \sqrt[0.2n]{0.1}\right) * (0.8n + 1) \tag{14}$$

When P_E is expressed as the lower bound formula model, the lower bound value m_2 for the number of enhanced node deployments is obtained as follows:

$$m_2 = \left(1 - \sqrt[0.2n]{0.1}\right) * n \tag{15}$$

And the actual number of augmented nodes deployed, m, can take m_1 and m_2 as reference, and here let m be the average of m_1 and m_2:

$$m = \left(1 - \sqrt[0.2n]{0.1}\right) * \frac{(1.8n + 1)}{2} \tag{16}$$

As can be seen from Figure 6, the number of enhanced nodes to be deployed is not the same for different IPFS network scales under the constraint that $\beta = 90\%$ is guaranteed, but the relationship between the two is not linearly increasing. After the number of augmented nodes deployed reaches a certain size, each augmented node is able to satisfy the better performance.

Note that m represents the minimum number of deployments required to satisfy the performance of the enhanced nodes. Analyzed from the perspective of resolution performance, the resolution probability of the augmented nodes tends to 1 when the number continues to increase, but if the number of enhanced nodes is reduced, it leads to a decrease in its performance.

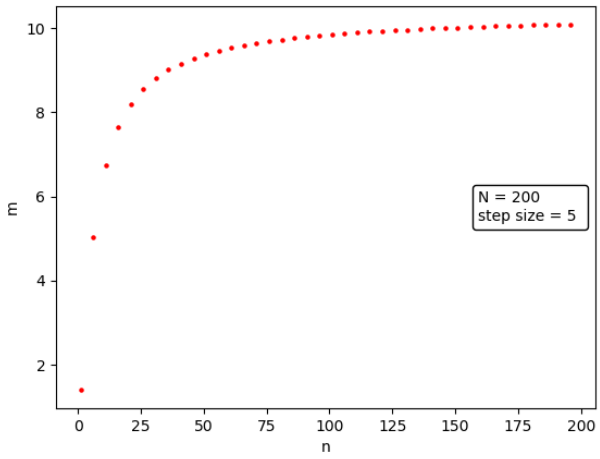


Figure 6. The impact of network scale on the overall resolution performance of the enhanced nodes.

3.2.2. Distribution Performance Analysis

The duplicate data delivery problem of the traditional IPFS network is mainly due to the introduction of the Bitswap mechanism. However, although the enhanced node integrates the resolution capability of ENRS and Bitswap, it prioritizes the resolution through ENRS and the resolution through the capability of ENRS. The enhanced node only needs to send one resolution message to ENRS, so it only needs to perform the delivery task once if the success rate of ENRS resolution can be guaranteed.

Based on the principle of the Bitswap mechanism described above, we found that the duplicate packets generated by the Bitswap mechanism are closely related to the number of peer points on the IPFS nodes. First, we disregard the deletion signaling of Bitswap and consider that one delivery signaling is generated for each resolution signaling. The resolution signaling delivery process of IPFS nodes is first retrieved through Bitswap, and when the Bitswap mechanism fails to retrieve, it is retrieved through the DHT mechanism. Therefore, let L_B be the expectation of the number of duplicate packets generated by the IPFS node through the Bitswap mechanism.

$$L_B = l * P_B + (l + 1) * P_B \tag{17}$$

Since the enhanced nodes are deployed by deploying IPFS applications on the routing nodes of ICN, the enhanced nodes not only act as logical peer nodes for IPFS applications, but also play the function of routing transmissions in the network. Among them, the ICN routing node supports the separation of the control plane and the data plane, so the enhanced node is able to make different routing decisions by analyzing the data packets passed by different applications. In IPFS networks, it is then possible to reduce the transmission of duplicate packets through information management tables.

As shown in the red request path in Figure 7, the enhanced node acts as a gateway node for the IPFS client. As the logical peer node of the IPFS application, the number of duplicate packets L_M for the enhanced node is as follows:

$$L_M = P_E + L_B * (1 - P_E) \tag{18}$$

From the above equation, we can see that the higher the P_E, the smaller the packet L_M generated by the transmission.

As shown in Figure 7 for the blue request path, when the enhanced node acts as a routing node, the effect on L_B is as follows:

$$L_B = l * P_B * (1 - \alpha) + (l + 1) * P_B \tag{19}$$

where α denotes the ratio of the number of duplicate enhanced nodes that the route passes through to the peers in the list when the IPFS node resolves the request through the Bitswap mechanism. The core mechanism is the information management table maintained by the enhanced nodes to manage the IPFS request signaling data passing through the same enhanced nodes. Packets with the same source and CID over a period of time are intercepted.

Therefore, it is a challenge to enhance the performance of the IPFS network to select node locations with high influential and high traffic volume as deployment nodes for enhanced nodes in the network. Deploying these nodes in the network as enhanced nodes can maximize the capability of information management tables and effectively improve the data delivery capability of the underlying IPFS network, thus increasing the availability of the IPFS network.

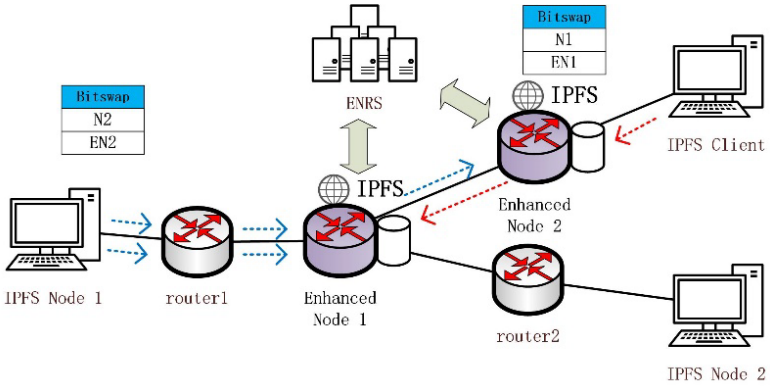


Figure 7. Enhanced nodes for different roles.

4. Enhanced Node Selection Algorithm

4.1. Analysis of Influential Nodes in Complex Networks

Selecting a set of high-impact nodes from a complex network has good applications in advertising, epidemic analysis, and network information diffusion, and related researchers have proposed many excellent algorithms, among which the VoteRank [41] algorithm, which has received further attention for its lower complexity and faster propagation speed. However, the VoteRank algorithm assigns the same voting probability to all nodes and does not make a distinction based on the local information of each voting node. To solve this problem, Guo et al. introduced the information entropy to measure the voting ability among nodes based on the VoteRank algorithm and proposed the Enrenew algorithm [42].

But both VoteRank and Enrenew still simply adopt the degree information, which is analyzed on unweighted graphs and cannot accurately solve some models with complex requirements. Real-world networks are more in line with the model of weighted graphs [43], and in our proposed network model, further information about the data transmission traffic in the network needs to be considered, since the augmentation nodes, when acting as routing nodes, are able to analyze the information carried in the transmitted data and implement the management of duplicate packets based on the management information table. Therefore, when the enhanced nodes are deployed in high traffic critical nodes, they are able to provide services to more IPFS nodes.

4.2. High-Influence Node Selection Algorithm Based on Node Traffic

Our algorithm is called PwRank is mainly optimized for VoteRank and Enrenew in weighted application scenarios. We represent the IPFS network in the resolution domain as a weighted network $G(V, E, W)$, where the weight ω_{ij} denotes the number of packets sent to each other between n node i and node j over a period of time, $\omega_{ij} = \omega_{ji}$. To consider

local information, we still use information entropy to calculate the mutual voting ability between nodes.

$$E_i = \sum_{j \in \Gamma i} H_{ji} = \sum_{j \in \Gamma i} -p_{ji} log p_{ji} \tag{20}$$

This is the formula for calculating node scores based on information entropy proposed by Enrenew based on VoteRank [42]. E_i denotes the voting score of node i, while H_{ji} denotes the voting power given to node i by each neighbor node j. p_{ji} is related to the node degree, denoted as $p_{ji} = \frac{d_{ji}}{\sum_{k \in \Gamma i} d_{ki}}$, $\sum_{k \in \Gamma i} d_{ki}$ is denoted as the sum of all degrees of node i's directly connected neighbor node k, d_{ji} denotes the degree of node i's neighbor node j, and Γi denotes the set of node i's direct neighbors, so $\sum_{j \in \Gamma i} p_{ji} = 1$. However, this formula has the limitation that it does not reflect the weight influence when selecting the influence nodes of the weighted network.

$$S_i = \sum_{j \in \Gamma i} H_{\omega_{ji}} = \sum_{j \in \Gamma i} \omega_{ji} H_{ji} = \sum_{j \in \Gamma i} -\omega_{ji} p_{ji} log p_{ji} \tag{21}$$

Equation (21) is used for calculating the voting fraction of a weighted network node i, where ω denotes the weight, and ω_{ji} denotes the weight between node j and node i. $H_{\omega_{ji}}$ denotes the ability of node j to vote on node i in the weighted network. For any $\omega = 1$, the network is denoted as an unweighted network. In the IBIHA architecture, the enhanced nodes have a higher reputation after a stable operation because of their stronger resolving ability and better delivery quality. There exists a higher probability that the Bitswap mechanism of ordinary IPFS nodes in the network will include them in the peer list. A reasonable deployment scheme for an enhanced node must prevent the coverage of the node's influence domain. If, for an ordinary node, multiple enhanced nodes appear within a one-hop distance, then there is bound to be an overlap of influence. Therefore, when an augmented node is deployed, it must be optimal for its neighboring nodes within at least one hop. In a realistic network, this can be expressed as a voting process in which all persons vote for those with close relationships, so the voting power accumulates as the closeness is passed, and when the first round of voting is over, a winner is bound to emerge. Therefore, for the winner's surrounding people, intimate voting ability must be suppressed.

When the node with the highest influence is filtered, for its directly connected neighboring nodes, we can call it the influenced node. The influential node needs to suppress the voting ability of its neighboring nodes to the maximum, and its suppression ability decreases with the spread of the range, so the voting ability suppression model for neighboring nodes can be expressed by the following equation.

$$f * H_{\omega_{ji}} = (1 - P_M)^{\sigma} \omega_{ji} H_{ji} \tag{22}$$

In the above equation, $f = (1 - P_M)^{\sigma}$, where the suppression factor f indicates that the voting ability of node j for i is suppressed, P_M denotes the resolution ability of the enhanced node, and σ denotes the influence range of the enhanced node. When $\sigma = 2$, the suppression factor of the enhanced node for the directly connected neighbor nodes is $(1 - P_M)^2$, while the suppression factor of the neighbor nodes of the directly connected neighbor nodes is $(1 - P_M)$, see Algorithm 1.

Algorithm 1. PwRank

Input: A network $G = (V, E, W)$, and the number of nodes n, and resolve probability p
Output: A set I including n influential nodes.

function Updata(i, H_ω, $\sigma' + 1$, p)
  if $\sigma' > \sigma$ **then**
    return *Null*
  end if
  for j in Γ_i **do**
    Updata(j, H_ω, $\sigma' + 1$, p)
    $H_{\omega_{ij}} = (1-p)^{\sigma-\sigma'} H_{\omega_{ij}}$
  end for
  return *Null*
end function
//PwRank main function
$I = O$
$S = O$
for i in V **do**
  for j in Γ_i **do**
    $p_{ji} = \frac{dji}{\sum_{k \in \Gamma i} d_{ki}}$
    $H_{\omega_{ji}} = -\omega_{ji} * p_{ji} \log_2 p_{ji}$
  end for
end for
while $|I| < n$ **do**
  for i, j **in** E **do**
    $S_i = S_i + H_{\omega_{ji}}$
    $S_j = S_j + H_{\omega_{ij}}$
  end for
  for i in I **do**
    $S_i = 0$
  end for
  add v to I, where $v = argmaxS_i$
  $\sigma' = 0$
  Updata(v, H_ω, σ', p)
end while
return I

5. Simulation Results and Analysis

In this section we compare the performance improvement of the IBIHA architecture for IPFS networks when the enhanced nodes are used as application nodes and routing nodes, respectively. In addition, we compare the PwRank algorithm with some classical influence node selection strategies. First, the simulation scheme is presented, and then the simulation results are given. The main metrics analyzed are as follows:

Resolution latency: The average latency of all nodes across the network from the time a resolution request is initiated to the time the request is routed to the node with the requested data information.

Duplicate packet ratio: the ratio of the number of received network packets to the number of initiated resolution requests.

Resolution probability: The ratio of the number of successful resolutions of all nodes in the network to the number of total resolution signaling initiated.

5.1. Application Layer Performance Comparison

5.1.1. Setting of Experimental Environment

In order to evaluate the performance enhancement of ICNs with field resolution systems on IPFS systems, we used a simulation scheme similar to the one mentioned in the protocol lab [26] and introduced IBIHA for comparison. This scheme focuses on the performance improvement of the enhanced nodes compared to the traditional IPFS nodes

at the application level. Thus a fully connected network consisting of n nodes is constructed through NS3, where n is set to [10,15,20,25,30]. Since the network uses a fully connected topology with direct connections between the nodes, the enhanced nodes are only used as peer points in the IPFS network and no routing feature is used. One of the nodes acts as a service node for storing information, and the rest are request nodes. The bandwidth of all links is 100 Mbps, and the propagation delay is 100 ms. We simulate three different route resolution systems to evaluate the network data delivery capability, including: DHT, IPFS, and hybrid resolution architectures. Initially, we deploy DHT (Kademlia) as the underlying routing system in the network. Then we introduce Bitswap of IPFS as a supplement to the routing system, where we set the number of peer neighbors of each node to $l = 5$ and do not directly connect with peers that have data. Moreover, the TTL of retrieval signaling of Bitswap is 1. Finally, the on-site resolution system is introduced as a supplement to the system, and the number of enhanced nodes within the coverage of the resolution system is set according to Equation (16).

5.1.2. Performance Analysis

It is observed through Figure 8a that the resolution time taken to retrieve the data increases with the number of nodes for the network with DHT as the route resolution system. And the increase in time is greater than that of the network based on bitswap and hybrid resolution architecture. As the number of nodes increases, the improvement in resolution performance of the bitswap and hybrid resolution architectures is more obvious. At our experimental scale, the hybrid resolution reduces the average resolution latency of the IPFS network by about 25% compared to the Bitswap mechanism.

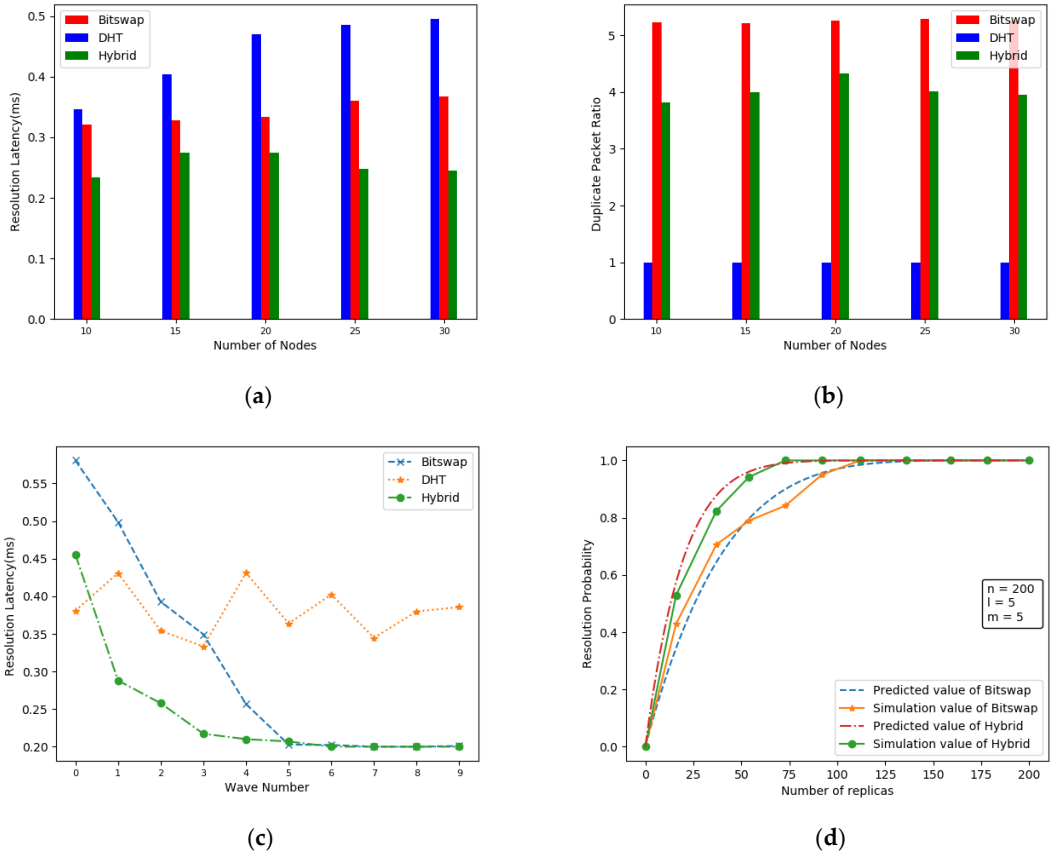


Figure 8. Performance comparison. (**a**) Average resolution delay of the network at different network sizes; (**b**) proportion of duplicate packets in the network at different network sizes; (**c**) average resolution delay at different wave numbers; (**d**) Comparison of simulated data with model predicted values.

As shown in Figure 8b, the cost of using the Bitswap block-switching protocol is that it increases the request packet traffic within the network by a factor of 5. This is consistent with the simulation results from the protocol lab. Although Bitswap has introduced CANCEL messages, the data delivery time is much longer than the resolution time in a limited network, so there is still a large number of duplicate packets in the network. In contrast, the enhanced nodes prioritize the resolution to ENRS, so no additional packets will result for requests that are successfully resolved by ENRS.

The simulation platform simulates a total of 10 data retrieval scenarios, and Figure 8c shows the average resolution latency of the network at different wave counts, where the wave number mainly indicates the batch of retrieval requests initiated by the network nodes in the simulation experiment. When the simulated network starts, due to the setting that all nodes are not directly connected to the data owner, neither Bitswap nor Hybrid can successfully parse the data compared to the DHT resolution, thus causing an additional communication overhead, but as the number of replicas gradually increases, it is clear that the Hybrid resolution can provide a lower resolution latency. Later, as the number of waves increases, leading to an increasing number of replicas in the network, both Bitswap and Hybrid resolution will tend to successfully parse the replicas within a one-hop distance, but the Hybrid resolution is clearly able to reach this state faster. This is because, compared to the Bitswap mechanism, the Hybrid resolution introduces ENRS, which enhances the sensitivity of IPFS nodes to in-region replicas.

Finally, Figure 8d shows the comparison between the predicted values of the Bitswap resolution mechanism of IPFS and the hybrid resolution mechanism of IBIHA and the actual simulated data. In terms of the results, although there are some fluctuations between the predicted and simulated values, the probability of the resolution results in the simulation experiments varies with the growth of the number of copies, which is consistent with our proposed model.

5.2. Network Layer Performance Comparison

5.2.1. Experimental Setup

As shown in Figure 9, we use an example network structure to verify the soundness of the proposed PwRank algorithm. In the above network topology, we divide it into three main regions, and each region uses a different network topology, respectively. The main areas include: star topology, ring topology, and fat tree topology.

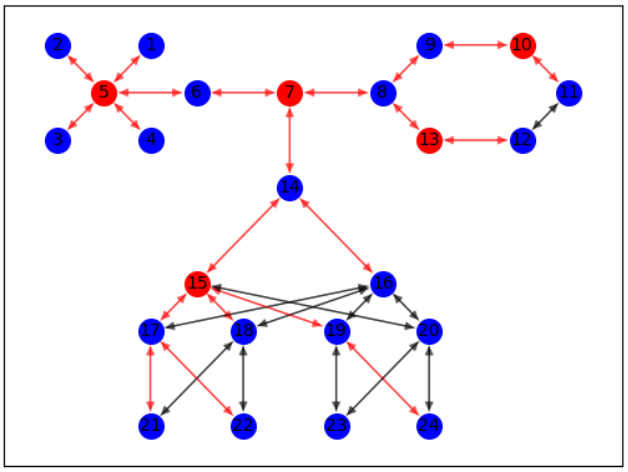


Figure 9. This is an example of a network topology diagram that contains 24 nodes with node numbers 1–24.

Where the blue nodes are normal routing nodes, while the red nodes indicate high-impact nodes filtered by our algorithm, and the color of the connection between nodes represents the amount of data transmitted between nodes: red represents a high amount of

data transmission between nodes, and therefore has a high weight, while black represents a low amount of data transmission and has a relatively small weight. We set the transmission delay between nodes to 20 ms.

We performed a network simulation by python, using an IPFS network as the baseline, and then used k-shel, VoteRank, and Enrenew as the comparison scheme. We set the deployment number n of enhanced nodes as [2–6], by simulating a request process for a large number of nodes, and finally got the following results.

5.2.2. Performance Analysis

As shown in Figure 10a, the proportion of the number of duplicate packets in the network is significantly improved by the deployment of the enhanced nodes. Among them, VoteRank, Enrenew, and PwRank have the most obvious effect. Combined with the analysis in Table 1, the Enrenew algorithm, due to the excessive focus on the degree information of the network in node selection, leads to the fat tree region with more complex links. However, it is obvious that PwRank pays more attention to the historical traffic information of the network compared with other algorithms, and the performance increase effect after deployment is more significant.

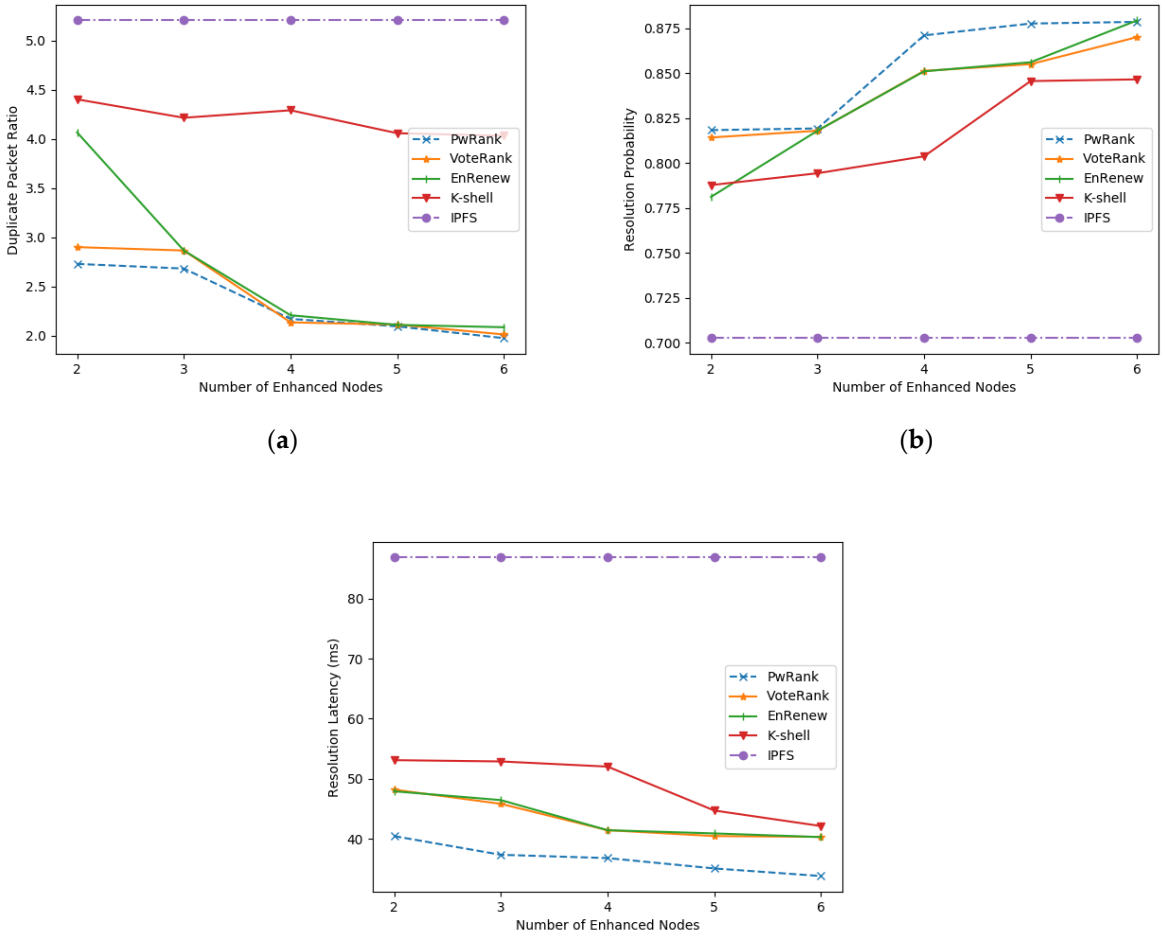


Figure 10. Comparison of algorithm performance in example networks. (**a**) comparison of the ratio of the number of duplicate packets; (**b**) comparison of the resolution probability; (**c**) comparison of the resolution delay.

Table 1. The set of influence nodes selected by different algorithms (n = 5).

Method	Influential Nodes				
	1	**2**	**3**	**4**	**5**
k-shell [44]	5	1	3	2	4
VoteRank [41]	5	15	16	8	11
Enrenew [42]	15	16	5	7	17
PwRank	7	15	10	5	13

Figure 10b shows that the average resolution rate increases as the number of deployed nodes is enhanced. The resolution performance improvement for the whole network tends to level off after a certain number of enhanced nodes are deployed in the network. This is basically consistent with our previous description of the relationship between the resolution performance and the number of enhanced nodes deployed in the network, where the resolution performance improves significantly with the increase in the number of deployed nodes in the early stage and remains at a stable state with a high resolution rate in the later stage. The PwRank algorithm can help the network converge to this stable state faster.

Figure 10c depicts the relationship between the number of augmented nodes deployed and the resolution delay. The traditional IPFS network does not introduce the augmented nodes, so it takes the longest time, more than 4 hops, to query the nodes with data, while after the introduction of the augmented nodes, the resolution latency of the whole network decreases significantly, regardless of the algorithm. The PwRank algorithm, which introduces the amount of data transmitted between nodes as a parameter, is more sensitive to the nodes that are active in the network and can therefore be better deployed in active areas of the network, thus improving the network resolution latency more significantly.

5.2.3. Real Network Performance Analysis

We also performed the same simulation experiments on a real network dataset provided by Ryan A. Rossi and Nesreen K. Ahmed [45], to further validate the performance benefits of our proposed architecture. The Figure 11 shows the real network topology that we used, which contains 2113 nodes. In order to introduce the weight parameter, we randomly selected a certain number of active request nodes from their dataset with the storage nodes used to place high hot data, and they all conform to the zipf law [40]. All nodes in the whole network randomly generated a large number of data requests, went through DHT and Bitswap mechanisms to retrieve data for data delivery, and recorded the number of data interactions between nodes as network weight information. Then, the same simulation experiments as in the example network were performed.

As shown in Figure 12, the performance of the network improves as the number of deployed nodes increases. Although the network size increases, it still indicates that our proposed PwRank algorithm shows better advantages for the augmented node deployment problem. The focus on deploying augmented nodes at high traffic makes the information management table more useful for managing redundant packets in the network. In addition, there is an improvement in the probability and time of resolution for the network. The experiment verifies that our proposed IBIHA architecture has a significant increase effect on the availability of the IPFS network.

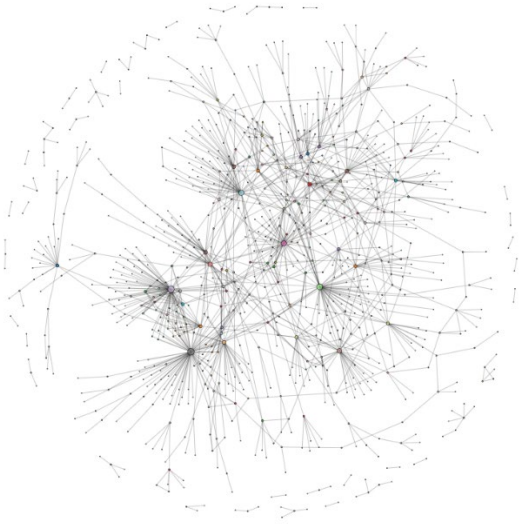
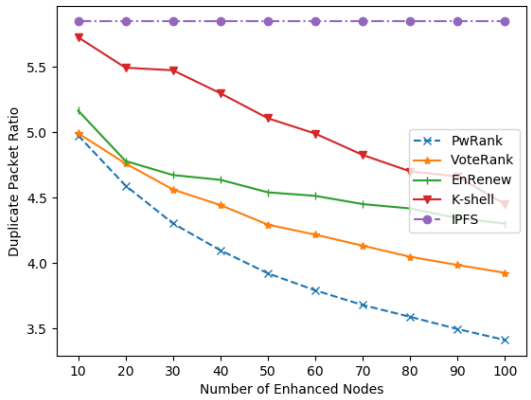
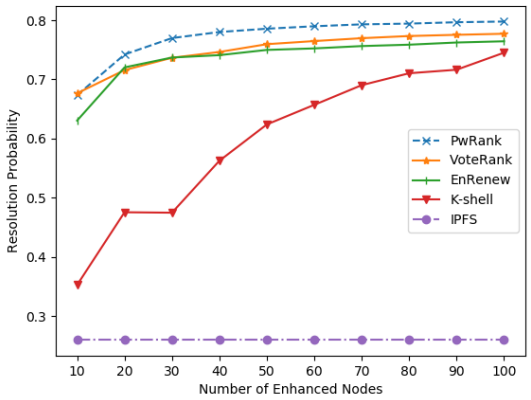


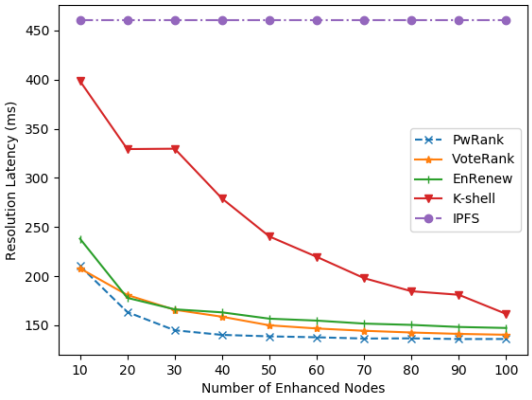
Figure 11. Real network topology.



(**a**)



(**b**)



(**c**)

Figure 12. Performance comparison under real network topology data. (**a**) comparison of the ratio of the number of duplicate packets; (**b**) comparison of the resolution probability; (**c**) comparison of the resolution delay.

6. Conclusions

In this paper we discuss the availability problem of IPFS. To solve this problem, a network architecture based on combining ICN networks with IPFS is proposed, called the IBIHA architecture. The scheme first proposes the concept of augmented nodes and shows the feasibility and advantages of deploying IPFS applications in an ICN network. Then, a high-impact node screening algorithm based on inter-node data traffic is proposed to deploy the enhanced nodes. We mainly consider metrics such as resolution latency, the duplicate packet ratio of the network, and the resolution success rate as the criteria of network performance. Finally, we compare our designed PwRank algorithm with some existing classical algorithms to fully measure the performance of our proposed method. The results show that our proposed IBIHA architecture and deployment strategy can provide local IPFS networks with lower resolution delay and better results in reducing the redundant packets in the network caused by Bitswap.

Although IPFS is an excellent project, IPFS still has many challenges, and relatively little research has been done on IPFS technology itself. In the future, our research direction will further focus on applying ICN to improve the shortcomings of IPFS networks.

Author Contributions: Conceptualization, R.Z., J.Y. and Y.L.; methodology, R.Z., J.Y. and R.H.; software, R.Z.; investigation, R.Z.; writing—original draft preparation, R.Z.; writing—review and editing, R.Z., J.Y., Y.L. and R.H.; supervision, J.Y.; project administration, Y.L.; funding acquisition, J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Strategic Leadership Project of Chinese Academy of Sciences: SEANET Technology Standardization Research System Development (Project No. XDC02070100).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to express our gratitude to Jinlin Wang, Jiali You, Yang Li, and Rui Han for their meaningful support of this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bieri, C. An Overview into the InterPlanetary File System (IPFS): Use Cases, Advantages, and Drawbacks. In *Communication Systems XIV*; University of Zurich: Zurich, Switzerland, 2021; p. 78.
2. Haßlinger, G.; Hartleb, F. Content delivery and caching from a network provider's perspective. *Comput. Netw.* **2011**, *55*, 3991–4006. [CrossRef]
3. Ascigil, O.; Reñé, S.; Król, M.; Pavlou, G.; Zhang, L.; Hasegawa, T.; Koizumi, Y.; Kita, K. Towards peer-to-peer content retrieval markets: Enhancing IPFS with ICN. In Proceedings of the 6th ACM Conference on Information-Centric Networking 2019, Macao, China, 24–26 September 2019; pp. 78–88.
4. Ghaznavi, M.; Jalalpour, E.; Salahuddin, M.A.; Boutaba, R.; Migault, D.; Preda, S. Content Delivery Network Security: A Survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 2166–2190. [CrossRef]
5. Nakamoto, S.; Bitcoin, A. A Peer-to-Peer Electronic Cash System. Bitcoin. 2008. Available online: https://bitcoin.org/bitcoin.Pdf (accessed on 25 January 2022).
6. Buterin, V. Ethereum white paper. *GitHub Repos.* **2013**, *1*, 22–23.
7. Benet, J. Ipfs-content addressed, versioned, p2p file system. *arXiv* **2014**, arXiv:1407.3561. *preprint*.
8. Alessi, M.; Camillo, A.; Giangreco, E.; Matera, M.; Pino, S.; Storelli, D. Make users own their data: A Decentralized Personal Data Store Prototype Based on Ethereum and Ipfs. In Proceedings of the 2018 3rd International Conference on Smart and Sustainable Technologies (SpliTech), Split, Croatia, 26–29 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–7.
9. Sun, J.; Yao, X.; Wang, S.; Wu, Y. Blockchain-based secure storage and access scheme for electronic medical records in IPFS. *IEEE Access* **2020**, *8*, 59389–59401. [CrossRef]
10. Tenorio-Fornés, A.; Hassan, S.; Pavón, J. Open peer-to-peer systems over blockchain and ipfs: An agent oriented framework. In *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*; Association for Computing Machinery: New York, NY, USA, 2018; pp. 19–24.
11. Ye, H.; Park, S. Reliable vehicle data storage using blockchain and IPFS. *Electronics* **2021**, *10*, 1130. [CrossRef]

12. Ortega, V.; Monserrat, J.F. Semantic Distributed Data for Vehicular Networks Using the Inter-Planetary File System. *Sensors* **2020**, *20*, 6404. [CrossRef] [PubMed]
13. Muralidharan, S.; Ko, H. An InterPlanetary file system (IPFS) based IoT framework. In Proceedings of the 2019 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 11–13 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–2.
14. Pappas, C.; Chatzopoulos, D.; Lalis, S.; Vavalis, M. Ipls: A framework for decentralized federated learning. In Proceedings of the 2021 IFIP Networking Conference (IFIP Networking), Espoo and Helsinki, Finland, 21–24 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6.
15. Confais, B.; Lebre, A.; Parrein, B. An Object Store Service for a Fog/Edge Computing Infrastructure Based on Ipfs and a Scale-Out nas. In Proceedings of the IEEE 1st International Conference on Fog and Edge Computing (ICFEC), Madrid, Spain, 14–15 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 41–50.
16. Shen, J.; Li, Y.; Zhou, Y.; Wang, X. Understanding I/O performance of IPFS storage: A client's perspective. In Proceedings of the 2019 IEEE/ACM 27th International Symposium on Quality of Service (IWQoS), Phoenix, AZ, USA, 24–25 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–10.
17. Abdullah Lajam, O.; Ahmed Helmy, T. Performance Evaluation of IPFS in Private Networks. In Proceedings of the 2021 4th International Conference on Data Storage and Data Engineering, Barcelona, Spain, 18–20 February 2021; pp. 77–84.
18. Henningsen, S.; Florian, M.; Rust, S.; Scheuermann, B. Mapping the interplanetary filesystem. In Proceedings of the 2020 IFIP Networking Conference (Networking), Paris, France, 22–26 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 289–297.
19. On, G.; Schmitt, J.; Steinmetz, R. The Effectiveness of Realistic Replication Strategies on Quality of Availability for Peer-To-Peer Systems. In Proceedings of the Third International Conference on Peer-To-Peer Computing (P2P2003), Linkoping, Sweden, 1–3 September 2003; IEEE: Piscataway, NJ, USA, 2003; pp. 57–64.
20. Spaho, E.; Barolli, A.; Xhafa, F.; Barolli, L. P2P Data Replication: Techniques and Applications. In *Modeling and Processing for Next-Generation Big-Data Technologies*; Springer International Publishing: Cham, Switzerland, 2015.
21. Guidi, B.; Michienzi, A.; Ricci, L. Data persistence in decentralized social applications: The ipfs approach. In Proceedings of the 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–4.
22. IPFS 2022. IPNS. Available online: https://docs.ipfs.io/concepts/ipns/ (accessed on 25 January 2022).
23. IPFS 2022. Merkle-DAG. Available online: https://docs.ipfs.io/concepts/merkle-dag/ (accessed on 25 January 2022).
24. Maymounkov, P.; Mazieres, D. Kademlia: A Peer-To-Peer Information System Based on the Xor Metric. In *International Workshop on Peer-to-Peer Systems*; Springer: Berlin/Heidelberg, Germany, 2002; pp. 53–65.
25. Guidi, B.; Conti, M.; Passarella, A.; Ricci, L. Managing social contents in decentralized online social networks: A survey. *Online Soc. Netw. Media* **2018**, *7*, 12–29. [CrossRef]
26. De la Rocha, A.; Dias, D.; Psaras, Y. Accelerating Content Routing with Bitswap: A Multi-Path File Transfer Protocol in IPFS and Filecoin. 2021. Available online: https://research.protocol.ai/publications/accelerating-content-routing-with-bitswap-a-multi-path-file-transfer-protocol-in-ipfs-and-filecoin/ (accessed on 25 January 2022).
27. Doan, T.V.; Bajpai, V.; Psaras, Y.; Ott, J. Towards Decentralised Cloud Storage with IPFS: Opportunities, Challenges, and Future Directions. *arXiv* **2022**, arXiv:2202.06315. *preprint*.
28. Chen, Y.; Li, H.; Li, K.; Zhang, J. An improved P2P file system scheme based on IPFS and Blockchain. In Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 11–14 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 2652–2657.
29. IPFS 2022. IPFS Cluster. Available online: https://cluster.ipfs.io/ (accessed on 25 January 2022).
30. Shapiro, M.; Preguiça, N.; Baquero, C.; Zawirski, M. Conflict-free replicated data types. In *Symposium on Self-Stabilizing Systems*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 386–400.
31. Ongaro, D.; Ousterhout, J. In Search of an Understandable Consensus Algorithm. In Proceedings of the 2014 USENIX Annual Technical Conference (Usenix ATC 14), Philadelphia, PA, USA, 19–20 June 2014; pp. 305–319.
32. Zhang, L.; Afanasyev, A.; Burke, J.; Jacobson, V.; Claffy, K.C.; Crowley, P.; Papadopoulos, C.; Wang, L.; Zhang, B. Named data networking. *ACM SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 66–73. [CrossRef]
33. Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.F.; Briggs, N.H.; Braynard, R.L. Networking named content. In Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, Rome, Italy, 1–4 December 2009; pp. 1–12.
34. Raychaudhuri, D.; Nagaraja, K.; Venkataramani, A. Mobilityfirst: A robust and trustworthy mobility-centric architecture for the future internet. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* **2012**, *16*, 2–13. [CrossRef]
35. Koponen, T.; Chawla, M.; Chun, B.G.; Ermolinskiy, A.; Kim, K.H.; Shenker, S.; Stoica, I. A data-oriented (and beyond) network architecture. In Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, New York, NY, USA, 27–31 August 2007; pp. 181–192.
36. Wang, J.; Chen, G.; You, J.; Sun, P. SEANet: Architecture and Technologies of an On-site, Elastic, Autonomous Network. *J. Netw. New Media* **2020**, *9*, 1–8.
37. Dannewitz, C.; D'Ambrosio, M.; Vercellone, V. Hierarchical DHT-based name resolution for information-centric networks. *Comput. Commun.* **2013**, *36*, 736–749. [CrossRef]

38. Liao, Y.; Sheng, Y.; Wang, J. A deterministic latency name resolution framework using network partitioning for 5G-ICN integration. *Int. J. Innov. Comput. Inf. Control* **2019**, *15*, 1865–1880.
39. Song, Y.; Ni, H.; Zhu, X. An enhanced replica selection approach based on distance constraint in ICN. *Electronics* **2021**, *10*, 490. [CrossRef]
40. Adamic, L.A.; Huberman, B.A. Zipf's law and the Internet. *Glottometrics* **2002**, *3*, 143–150.
41. Zhang, J.X.; Chen, D.B.; Dong, Q.; Zhao, Z.D. Identifying a set of influential spreaders in complex networks. *Sci. Rep.* **2016**, *6*, 1–10. [CrossRef] [PubMed]
42. Guo, C.; Yang, L.; Chen, X.; Chen, D.; Gao, H.; Ma, J. Influential nodes identification in complex networks via information entropy. *Entropy* **2020**, *22*, 242. [CrossRef] [PubMed]
43. Sun, H.-L.; Chen, D.-B.; He, J.-L.; Ch'Ng, E. A voting approach to uncover multiple influential spreaders on weighted networks. *Phys. A Stat. Mech. Its Appl.* **2019**, *519*, 303–312. [CrossRef]
44. Kitsak, M.; Gallos, L.; Havlin, S.; Liljeros, F.; Muchnik, L.; Stanley, H.E.; Makse, H.A. Identification of influential spreaders in complex networks. *Nat. Phys.* **2010**, *6*, 888–893. [CrossRef]
45. Rossi, R.; Ahmed, N. The network data repository with interactive graph analytics and visualization. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence 2015, Austin, TX, USA, 25–30 January 2015.