IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Intercloud Resource Discovery using Blockchain

MEKHLA SHARMA[1], JAITEG SINGH[2], ANKUR GUPTA[3] SUDEEP TANWAR[4], GULSHAN SHARMA[5], AND I. E. DAVIDSON[6]

[1,3]DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, MIET, JAMMU, INDIA (EMAIL: MEKHLA.CSE@MIETJAMMU.IN)
[2]DEPARTMENT OF COMPUTER SCIENCE, CHITKARA UNIVERSITY, PUNJAB (EMAIL: JAITEG.SINGH@CHITKARA.EDU.IN)
[4]DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, INSTITUTE OF TECHNOLOGY, NIRMA UNIVERSITY, AHMEDABAD, GUJARAT, INDIA 382481, E-MAILS: (SUDEEP.TANWAR@NIRMAUNI.AC.IN)
[5,6]DEPARTMENT OF ELECTRICAL POWER ENGINEERING, DURBAN UNIVERSITY OF TECHNOLOGY, STEVE BIKO CAMPUS, DURBAN 4001, SOUTH AFRICA. (EMAILS: GULSHANS1@DUT.AC.ZA, INNOCENTD@DUT.AC.ZA)

CORRESPONDING AUTHOR: SUDEEP TANWAR (SUDEEP.TANWAR@NIRMAUNI.AC.IN), GULSHAN SHARMA (GULSHANS1@DUT.AC.ZA)

ABSTRACT The intercloud represents a logical evolution of cloud computing that extends its computational scale and geographic footprint by collaborating with disparate cloud service providers (CSPs) for resource sharing. Discovering resources belonging to heterogeneous CSPs is not only the primary but critical operation for the intercloud. However, achieving resource discovery in a deterministic manner within this global distributed environment is non-trivial. The literature has proposed several resource discovery approaches for the federated intercloud based on trusted and centralized third-party entities. Few approaches, however, exist for the non-federated intercloud, which by definition has no central entity to enable the resource discovery process. Some P2P-based resource discovery techniques have been proposed by researchers, industry players and standardization bodies like Global InterCloud Technology Forum (GICTF). However, existing P2P-based approaches in the non-federated intercloud do not adequately address authentication, non-repudiation of resource information, secure storage and management of transactional records, management of trust/reputation and optimal resource selection and provisioning. This research paper presents BIRD, a Blockchain-based Intercloud Resource Discovery framework that involves participating CSPs connected in a P2P network using blockchain to manage resource information and maintain transactional records. The BIRD framework alleviates the requirement of a trusted third party for discovering and managing resources. The main features involved in the BIRD framework are i) latency optimization, ii) fine-grained control mechanism, and iii) Quality-of-Service, Trust and Reputation (QTR) indices. Latency optimization achieves faster resource discovery, fine-grained control mechanism for intercloud resource discovery, and QTR is for quality CSP or resource selection. BIRD uses blockchain to maintain transactions between CSPs securely.

INDEX TERMS Blockchain, intercloud, resource discovery.

1. INTRODUCTION

Many experts have considered the intercloud scenario as the future of cloud computing [1] is also referred to as an interconnected global "Cloud of Clouds". The concept was introduced by Cisco [2, 3] to signify a network of clouds established on unified open standard protocols to provide cloud interoperability. It aims at seamlessly utilizing the storage and computational resources dispersed across private and public clouds. Thereby achieving an inter-planetary footprint, service delivery localization, high reliability, and flexibility while maintaining acceptable quality-of-service (QoS). Fig. 1.1 shows the Cisco view of the intercloud [4] wherein multiple private/public/hybrid models interconnect using Cisco's proprietary framework while interoperating seamlessly with cloud-hosted services and data belonging to public clouds from other CSPs such as Amazon, Google, Microsoft etc. Effectively leveraging resources across several geographically dispersed CSPs requires efficient orchestration enabling resources to be discovered, selected, allocated and de-allocated securely based on user-specified parameters like cost, quality-of-service (QoS), latency, reputation of CSPs or even a custom combination of these parameters.

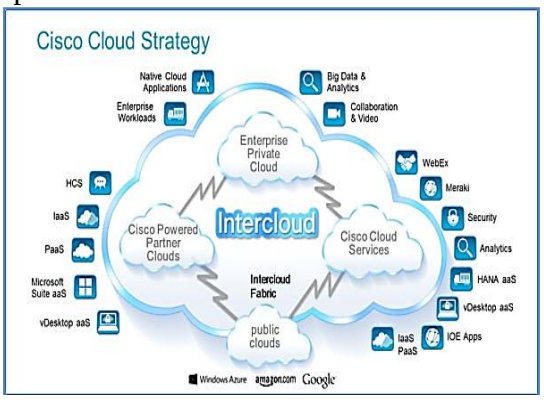

FIGURE1.1. Cisco's view of the unified intercloud offering seamless services over a multitude of cooperating clouds (Source: Cisco)

The resource discovery process within an intercloud environment typically involves operations such as publishing resource information, purging and refreshing resource information, selecting resources, reservation, instantiation and consumption of selected resources, releasing resources after consumption, and financial settlement. Hence, the resource discovery and provisioning process are central to the operation of the intercloud. Given the variety of resources, intercloud scale, and divergent user requirements, the resource discovery process assumes significant complexity, especially for the non-federated intercloud model with no centralized entities and a trusted third party.

Several standardization efforts have been ongoing to help define common operations, metrics, and benchmarks for intercloud resource discovery. Industry bodies such as GICTF [5] and the intercloud Forum [6] have led these efforts. However, these efforts have focused primarily on the federated intercloud model, which offers trusted central entities and services for authentication and brokers providing resource discovery and orchestration services. The non-federated intercloud model, which by definition is decentralized with no trusted third-party entities, have largely remained outside the purview of the standardization efforts. Thus, several open research challenges persist for researchers in this domain. According to Toosi et al. [7] resource discovery is one of the major challenges in heterogeneous intercloud environments. Later, Sharma et al. [8] suggested that achieving efficient and secure resource discovery within an intercloud environment remains non-trivial. Then, Martino et al. [9] presented a position paper providing intercloud research areas and challenges, establishing an initial research roadmap for intercloud computing development areas. Broadly, the challenges in intercloud resource discovery for a non-federated intercloud environment can be summarised as follows:

- Establishing identities of participating cloud service providers and performing authentication are challenging in the absence of a trusted third party.
- The absence of centralized resource registries or directories provides easy lookup services.
- Establishing non-repudiation for stated resource information by participating CSPs.
- Ensuring that all participating CSPs have the same unambiguous view of the resource information at all times.
- Ensuring resource discovery in a deterministic and optimal manner.
- Allowing fine-grained and customized control to CSPs in specifying criteria for resource selection which best meet their requirements.
- Ensuring that malicious or collaborating participants cannot manipulate transactional records and resource information.

Hence, designing an efficient and dependable strategy for resource discovery in the non-federated intercloud environment is non-trivial yet imperative for realizing the vision for the non-federated intercloud. This paper proposed an efficient and secure decentralized mechanism of resource discovery for the non-federated intercloud. The proposed Blockchain-based Intercloud Resource Discovery framework, i.e., BIRD, organizes the intercloud as a latency-optimized P2P network. It also makes innovative use of blockchain [10] to serve as an immutable distributed resource ledger and encrypted store for transactional records, removing the requirement for a trusted third party. BIRD addresses significant gaps in the domain and provides a viable and novel model for non-federated intercloud resource discovery.

1.1. EXISTING WORK

Distributed systems such as Grids, P2P networks and cloud provide a distributed computing infrastructure that can leverage globally available network resources. In these systems, resource discovery is the foremost process, wherein the resource consumer figures out the resource offering that best meets its requirements amongst the available offerings by resource providers. Thus, resource discovery involves searching and locating specific resources across a large distributed space with potentially many options. However, discovering resources among the intercloud across multiple participating CSPs is inherently complex due to its heterogeneous nature, global footprint and dynamic availability of resources, as also stated in [11]. Thus, resource discovery mechanisms formulated for the Grid, P2P networks and even a homogenous cloud environment are not directly applicable to the intercloud, especially for the non-federated model with no centralized entities, requiring specialized mechanisms to be devised. Fig. 1.2 provides a classification for the intercloud models proposed by researchers in [6, 12]. Broadly there are two main classes of intercloud models; federated and non-federated intercloud models. The Federated intercloud model offers trusted central entities and services for authentication and brokers providing resource discovery and orchestration services. Different groups of cloud service providers willingly collaborate and interconnect with one another to share resources. Non-federated cloud is a decentralized environment that is more open and flexible that has no central entity such as resource repositories and brokers to enable the resource discovery process

Resource discovery approaches for the intercloud can be classified as follows:

Broker-based Approach: A majority of the resource discovery approaches existing in the literature pertaining to the broker-based approach. It is widely adopted to discover resources in intercloud. Brokers act as intermediary entities for orchestrating resource requirements between different CSPs. It provides central resource repositories, cache resource information and offers a convenient lookup service for resource discovery. Finally, an encrypted central store for transactional records is used for financial settlements. These centralized elements facilitate resource discovery, matching and selection across CSPs. Lheureux and Plummer [13] emphasize the significance of a brokerage service, which defines three different cloud brokerage types, including arbitrage, aggregation, and intermediation. Similarly, Buyya et al. [14] have acknowledged the significant role of cloud brokers and their multiple

responsibilities ranging from service aggregation to monitoring. Geetha et al. [15] has compared the features of brokers and explained the various brokering frameworks as follows:

- Grozev and Buyya [16] proposed the intercloud framework is an architecture related to a federated intercloud environment consisting of a broker, a cloud exchange, & a cloud coordinator. They have proposed a centralized approach where the central entity can perform resource selection and allocation for requesting users.
- The SLA-based tiered pricing model is proposed by Nair et al. [17], where cloud broker provides the broker service to provide identity management, access management, policy enforcement, and audit capabilities to CSPs in an intercloud

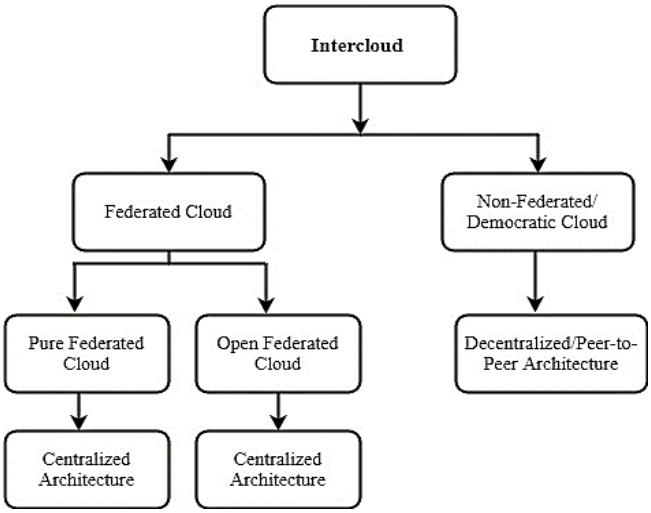

FIGURE1.2. Classification of intercloud models

- CloudRank framework proposed by Zheng et al. [18] focuses on predicting QoS compliance of potential resources. Different values from users of the same cloud service are collected, and rank is predicted based on the perceived ability to meet QoS parameters. This helps CSPs in selecting the best resources.
- Meta-computing scheduling architecture for resources, as proposed by Schwiegelshohn et al. [19], is based on the concept of brokerage and trading, involving a market-based structure connecting various subdomains that discover resources and transact with each other.
- Bessis et al. [20] proposed an algorithmic model for orchestrating job execution in intercloud, especially in the face of flash-crowd scenarios, relying on some central elements to aid in the selection of computing resources to execute the remote jobs

However, the current broker models lack insight into resource level performance for each CSP, their past behaviour and reputation to help prospective collaborating CSPs make informed choices. Centralised brokers also tend to suffer from issues of delivering performance at an extreme scale, besides providing fault-tolerance and reliability as they represent a single-point-of-failure.

P2P-based Approach: The decentralized mechanisms, such as those based on P2P approaches, in non-federated intercloud environments, are naturally applicable to the inter-cloud due to their extreme scalability, resilience and fault-tolerance. P2P networks are naturally fault-tolerant due to the replication of the resource information at multiple nodes in a distributed manner. Gupta et al. [21] proposed a completely decentralized P2P framework that could enable effective resource provisioning over geographically dispersed cloud service providers (CSPs). Yet, the suggested strategy does not provide latency optimization while performing resource matching between providers and consumers. A P2P-based resource discovery scheme based on spatial awareness of the involved cloud data centres attached to different CSPs is proposed [22]. The scheme exploits the locality-based information of data centres and arranges them into the Distributed Hash Table (DHT) [23] for optimal communication. But, the work does not include quality of service criteria (like trust, availability, reputation etc.) that can leverage the selection of quality resources. Sotiriadis et al. [24] presented a decentralized and distributed strategy for discovering resources in a heterogeneous intercloud environment by proposing grouping resources depending upon old service experience encountered. However, the strategy proposed is based on creating clusters of transient resources and gets affected by overheads incurred due to the creation and separation of the clusters each time the resource availability changes, which is quite frequent. Additionally, managing the trail of past service experience of every involved participant incurs its overheads.

Not much work has been carried out regarding resource discovery using the P2P-based approach in the intercloud domain. Even when P2P-based approaches have been recommended by researchers and industry bodies such as GICTF, they have been silent on making the resource discovery process deterministic and secure with acceptable QoS, besides offering a distributed and fool-proof mechanism for storing transactions records and facilitating seamless financial settlements.

Agent-based Approach: Agents work on behalf of an entity and exhibit properties of autonomy, pro-activity, communication and negotiation, enabling the completion of the assigned task. Agents are used for resource discovery and management of resources between CSP-CSP, CSP-SP besides SLA negotiations. Cloudle [25], an agent-based search engine, supports similarity reasoning, compatibility reasoning and numerical reasoning by consulting cloud ontology for regulating the likeness between provider's and consumer's service specifications. Agent-based resource discovery strategy using bloom filter is proposed by Nikbazm R and Ahmadi M [26]. In this strategy, the resource information is stored in a Bloom filter which is then sent to the related broker agent, which matches the requirements against the resource database. Agents seem an appropriate mechanism for automating complex interactions within an intercloud environment and hence play a significant role in automating resource discovery for the intercloud among other functional areas [27].

Ontology-based Approach: Ontology [28] is a common vocabulary that promotes sharing of information in an inter-cloud domain and resulting in meaningful search. The main objective of an intercloud CSP is to provide numerable computing resources and total transparency while providing visibility of the resources simultaneously. It ensures that the resources/services can meet compliance and match the functional, architectural, policies and constraint requirements of other cloud service providers. Current work in this domain focuses on defining declarative semantic model/language that captures both requirements and constraints of computing resources. EDML [29] is defined as a modelling language that specifies data centre computing resources semantics, expressed in XML-based mark-up language. Willner et al. [30] proposed a Federated Infrastructure Discovery and Description Language (FIDDLE) that can be employed and diversified to interchange information happening between federated infrastructures to discover & consume the unused resources/services independently from specific APIs or architectures. However, FIDDLE does not address the non-federated intercloud model. Another approach in [31] addresses organizing resource information across multiple providers by enabling resource discovery and selection procedures based on multilayer ontology. The ontology describes user requirements, resource constraint requirements and cloud resource attributes. Also, a bi-dimensional matching algorithm performs the required attribute matching. However, the scheme does not take into account historical resource attributes and performance. Table I summarises the main contributions and limitations of the work done in intercloud resource discovery by various researchers.

TABLE I
CONTRIBUTIONS AND LIMITATIONS OF EXISTING INTERCLOUD RESOURCE DISCOVERY MODELS

Research Models/ Frame works	**Focus**	**Resource Discovery Approach**	**Architectural Model**	**Limitations**
Market-based negotiation model [6, 16]	Federation of clouds for performing resource allocation through central entity for providing guaranteed quality-of-service.	Broker-based Approach	Centralized Architecture	Suffers from single-point-of-failure, unshielded security and issues with non-adaptability in non-federated model. Requires a trusted third-party for centralised orchestration of resources.
Meta-computing scheduling architecture [19]	A meta-computing architecture linking independent resources and providing information about them to enable efficient discovery and orchestration.	Broker-based Approach	Centralized Architecture	Uses a centralised broker susceptible to single-point-of-failure and frequent resource information updation requests from diverse sources making orchestration sub-optimal.
Resource clustering architecture [24]	Resource discovery strategy based on previous resource requests and past results.	Broker-based Approach	Centralized Architecture	Scheme incurs overheads in creation of resource clusters and storing large amounts of resource discovery interactions to keep track of past requests, experiences and taking intelligent decisions based on that data. Updating the past experiences with current experiences takes time.
C2C-framework [21]	A P2P based non-federated model connecting CSPs together and enabling cloud-to-cloud (C2C) resource discovery creating a shared ecosystem of pooled compute resources.	P2P-based Approach	Decentralized Architecture	Not latency optimized. For large networks time taken for the network to converge increases. Moreover, resource information propagation takes place one hop at a time, which is less then optimal.
P2P-based distributed resource discovery [22]	Resource discovery mechanism based upon the spatial awareness of various cloud data centres that belong to diverse and disparate CSPs providing minimal response time	P2P-based Approach	Decentralized Architecture	Work does not include quality of service criterion (like trust, availability, reputation etc.) that can be leveraged for selection of quality resources.

Direction-aware resource discovery architecture [32]	Developed a direction-aware strategy for resource discovery allowing shaping of network traffic and minimizing latency in a distributed environment.	P2P-based Approach	Decentralized Architecture	Incapable of rapidly responding to changes in network topology or handling flash-crowd scenarios.
FIDDLE: Semantic information model [30]	Formally explains the federation between the cloud providers encompassing infrastructure and life cycle of various resources/services offered	Ontology-based Approach	Centralized Architecture	Failed to address resource discovery in the non-federated intercloud model
Multi-layers ontology scheme [31]	Multi-layer ontology is proposed from the view of software developer's requirements for resource discovery and selection procedure	Ontology-based Approach	Decentralized Architecture	Failed to consider historical resource attributes and performance.
Cloudle [25]	Agent based search engine proposed for bolstering cloud service discovery, service negotiation and service composition.	Agent-based Approach	Centralized Architecture	Uses set of centralized broker agents susceptible to single-point-of-failure

1.2 CONTRIBUTION OF THIS WORK

Compared to centralized resource discovery architectures for the intercloud, the decentralized models seem to be more scalable. Still, the operations involved are complex and require more coordination among the participating CSPs. Other issues involved, especially in non-federated decentralized models, include managing up-to-date resource information, providing fine-grained control over resource selection to CSPs, ensuring un-ambiguity of resource information, and orchestration at scale across different CSPs finally ensuring non-repudiation of transactional records. Most of the work done by researchers and standardization bodies has focused primarily on the federated intercloud model, which includes trusted central entities performing authentication and resource discovery & orchestration. The non-federated intercloud model does not include trusted third-party services, thus necessitating a dependable strategy for resource discovery. It can be seen from the review of existing work that a comprehensive scheme tailored to meet the resource discovery requirements of the non-federated intercloud. It provides fine-grained control over resource discovery, enabling high-quality resource selection, delivering acceptable performance, providing fool-proof security and non-repudiable transactional records in the absence of a trusted third party. It remains a major gap in the domain. This research paper is an extension of the work initially presented in [33,34]. It, therefore, proposes the BIRD framework, which makes novel use of blockchain to overcome the need for a trusted third party for enabling seamless resource discovery for the non-federated intercloud. A latency optimized P2P network of participating CSPs reduces the communication overheads while the shared ledger maintained by each CSP provides an unambiguous view of resources across the intercloud. Smart contracts implemented by each CSP consider cost, quality-of-service, trust and reputation of participating CSPs to enable customized and high-quality resource selection. Table II presents a feature comparison between the proposed BIRD framework and the two main resource discovery frameworks, GICTF [5] and the Intercloud Framework [6].

TABLE II
COMPARATIVE FEATURE ANALYSIS OF BIRD WITH EXISTING FRAMEWORKS

Feature/Offering	**GICTF**	**Intercloud Framework**	**BIRD**
Resource Discovery Approach (P2P-based or Broker-based)	P2P-based Approach	Broker-based	P2P-based Approach
Vendor lock-in Situation	Partial	Yes	No
Third-Party Reliance	Yes	Yes	No
Trust and Reputation	No	No	Yes
Fine Grained Resource Selection	No	No	Yes
Cloud Provider's SLA	Yes	Partial	No
Decentralization	Partial	No	Yes
Real-Time Resource Performance Monitoring	No	No	Yes
Transactional Records	Does not Address	Central Repository	Blockchain (Distributed)

The BIRD framework makes innovative use of the blockchain [10] concept to secure CSP-to-CSP transactions without requiring a trusted third party. It encompasses a fully decentralized, latency-optimized P2P network of participating CSPs, associated protocols for facilitating efficient resource discovery and a trusted blockchain ledger for maintaining and handling transactional records. Thus, the requirement of a centralized authority present in existing non-federated intercloud resource discovery frameworks is

alleviated. Multiple CSPs jointly arrive at a consensus recorded and entered in the ledger for future reference, ensuring immutability and transparency. Time-stamped transactions enable CSPs to maintain traces of resources advertised by other CSPs. Once a CSP negotiates a "contract" with another CSP, both the individual CSPs begin their transactions in a completely decentralized manner. Hence, the framework facilitates optimal resource discovery/selection and builds trusted relationships leading to a more secured resource discovery and sharing environment. Table III summarises the implications of using blockchain as an integral part of the BIRD framework and the features that it enables.

Table III
Benefits of using blockchain in BIRD framework

BIRD Features	Implication of using blockchain concept
Immutability and non-repudiation	• Blockchain provides a shared ledger storing all resource information in an encrypted manner. • Proof-of-stake, using encrypted real-time performance information and computed trust value for each CSP, is used to arrive at a consensus. Malicious CSPs are therefore unable to manipulate this information. • Incorporates authentication, verification and authorization of CSPs.
Resource selection	• Maintains the underlying details of the CSPs and resource availability in an encrypted form. • Helps in selecting and consuming resources through a standardized and decentralized seamless interface • Provides "Smart Contracts" feature allowing automatic execution of contracts when specified conditions are met.
Handling failures	• Involves a decentralized P2P architecture with the blockchain shared ledger such that even if a CSP fails, the resource discovery process will not halt as the same resource information is available through exact copies of the shared ledger maintained by all participating CSPs.
Quality of resource selection	• Ensures that deployed resource instances are from trusted CSPs that are continuously monitored and their performance data updated and stored in an encrypted format. CSPs cannot access or manipulate their historical performance data. • Trust, reputation and quality-of-service information is stored in the blockchain in the form of moving averages which are readily accessible to the smart contracts. This enables CSP specified weights to be applied for customised and secure resource selection.
Financial Settlement	• All contract information stored securely in the blockchain. • Resource consumption too recorded in the blockchain, enabling seamless final financial settlement among CSPs.

Following are the main contributions of the proposed BIRD framework:

- Latency optimization of the P2P network overlay topology confirming clustering of physically close CSPs leading to speeding-up of the resource discovery process.
- Ensuring multi-factor optimization and fine-grained control mechanisms for effective resource discovery and provisioning.
- Computation of cost, quality-of-service, trust and reputation indices for each CSP-to-CSP interaction, deriving cumulative metrics and further utilizing these metrics to improve quality of resource selection and negotiation decisions.
- The innovative use of blockchain to connect CSPs participating to record all real-time resource availability, negotiations, transactions, and usage information ensuring transparency, immutability and non-repudiation.
- Ability to operate securely and efficiently in a non-federated open/public intercloud.

1.3 ORGANIZATION AND READING MAP

The structure of the research paper is shown in Fig. 1.3. The rest of the paper is organized as follows: Section 2 provides the details of blockchain preliminaries used in the paper. Section 3 covers the methodology employed and the taxonomy of the research paper. Section 4 details the system model of the Blockchain-based Intercloud Resource Discovery (BIRD) Framework. Section 5 describes the sequence of operations in detail and the major protocols/algorithms involved. Section 6 describes the experimental results obtained and establishes the effectiveness of the proposed framework. Finally, section 7 concludes the paper and elaborates the future work that can be carried out to expand the proposed resource discovery framework.

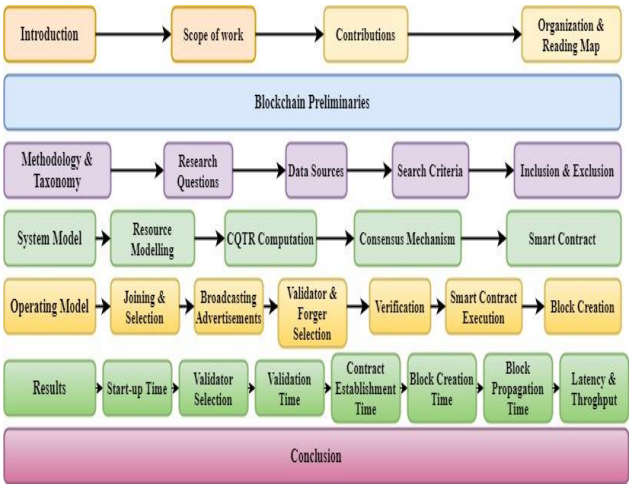

FIGURE1.3. Structure of the research paper

Table IV lists all the acronyms used in the paper.

TABLE IV
THE LIST OF ABBREVIATIONS

Acronym	Description	Acronym	Description
AHP	Analytic Hierarchy process	QoS	Quality-of-Service
BIRD	Blockchain-based Intercloud Resource Discovery	RA	Resource Advertisement

CSP	Cloud Service Provider	RR	Resource Request
DC	Data Centre	SLA	Service Level Agreement
SP	Service Provider	SSO	Single-Sign-On
DHT	Distributed Hash Table	WMA	Weighted Moving Average
VM	Virtual Machine	PMS	Performance Measurement Service
FIDDLE	Federated Infrastructure Discovery and Description Language	PoS	Proof-of-Stake
GICTF	Global Intercloud Technology Forum	PoW	Proof-of-Work
GUID	Globally Unique ID		

2. BLOCKCHAIN PRELIMINARIES

Blockchain is a distributed ledger technology providing a digital registry of transactions and finds wide applicability across industries which require maintaining records for financial transactions in a secure manner [35]. The prominent features of the blockchain technology include:

- ***Decentralization***: involves P2P architecture with participating nodes
- ***Distributed Shared Ledger***: acts as a trusted and authentic source for participating nodes.
- ***Consensus***: ensures that the exact copy of shared ledger exists at each participating node and thus lowers the possibility of fraudulent transactions since some participating nodes can tamper with data.
- ***Privacy & Confidentiality***: generates public & private key for record sealing which are protected via digital signature.
- ***Time-stamping***: defines the time when transactions were actually chained to the current block.

Some of the common blockchain terms used in this paper are explained below:

- ***Block***: Block is a container data structure that groups various transactions and includes them in the blockchain ledger. Each block contains a header; that consists of the metadata, accompanied by a group of transactions. A block is recognized by its hash value created using SHA256 [36] cryptographic hash algorithm stored within the header field of each block. Every block created refers to the previous generated block included in the blockchain.
- ***Hashing***: Hashing [37] is a mathematical function used for mapping data of variable size to a string of bits having a fixed size. In cryptocurrencies like Bitcoin [38], the transactions are used as input and hashing algorithm is executed to get a fixed-length output. Hashing enables security during the propagation of message transmission when intended for a particular recipient over the network..
- ***Consensus Algorithm***: It is a mechanism used to achieve agreement on a single data value among the distributed systems and designed to ensure reliability in the network.
- ***Proof-of-Work (PoW)***: Proof-of-work [39] is a commonly used consensus algorithm in cryptocurrency networks like Bitcoin. It is a mechanism in which each network participant, also known as 'miner', is required to prove that the work done and proposed qualifies them for adding a new block of transactions in the blockchain. The whole process is termed as 'mining' in Bitcoin and takes a longer processing time and high energy consumption.
- ***Proof-of-Stake (PoS)***: Proof-of-Stake [40] is another widely used consensus algorithm that works differently from the proof-of-work algorithm. In this mechanism, individual participants are chosen, also known as validators and forger, depending upon a certain set of criteria or stakes to validate transactions and generate blocks in the blockchain. Proof-of-Stake is considered as being superior block validating & generating algorithm over proof-of-work due to low computational and energy requirements.
- ***Validator***: Entity in Proof-of-Stake algorithm is responsible for verifying the transactions that happen between the participants within the blockchain.
- ***Forger***: Entity in Proof-of-Stake algorithm is responsible for writing blocks in the blockchain.
- ***Blockchain Transaction***: Blockchain transaction is a new exchange record of some value/data/agreement between two different participants. Transactions can happen in a node and take time to verify when a new block containing those transactions is created. The validator is responsible for verifying the transactions stored in blocks, and the blocks form a chain to create a blockchain. Hash is generated for each transaction which gets linked to the previous block created within the blockchain.
- ***Validation Process***: The mechanism of checking and verifying the transactions against some validation rules.
- ***Smart Contract***: It is defined as a computer code that runs over the blockchain network. It contains a certain set of rules/criteria specified by individual participants, which result in the contract getting executed by both parties when met. Smart contracts enable automated negotiation, contract establishment and transaction execution without requiring any trusted third-party.

3. METHODOLOGY AND TAXONOMY

In this section the methodology employed for the present research is outlined. Moreover, a detailed taxonomy based on literature review is presented.

3.1 PROBLEM STATEMENT

The intercloud is a group of interoperating clouds facilitating connectivity and collaboration. Although the

intercloud environment offers various benefits over traditional cloud deployment models, the investigation started with a fundamental question: what would it take for disparate cloud service providers within the intercloud environment to efficiently and effectively share resources? For disparate cloud service providers to flawlessly work together, reliable protocols are needed. Widely accepted standards currently do not exist for a genuinely open and non-federated intercloud, thus necessitating the formulation of a comprehensive framework allowing CSPs to collaborate without the need for a trusted third-party as required in the federated model.

The problem addressed by the present research, therefore, is *"To enable different cloud service providers to collaborate seamlessly by leveraging each other's resources, to achieve their mutual objectives in a performant, secure and flexible manner without conforming to the traditional federated model"* By associating with different clouds, cloud providers can leverage a shared pool of resources to provide enhanced service levels to their customers while gaining fiscal advantages. It also helps CSPs to amplify their geographic footprint without installing their own computing resources globally, saving significant capital investment.

3.2 RESEARCH QUESTIONS

The existing literature available on resource discovery within the intercloud environment was examined, we identified various gaps.

Table V summarizes the identified research questions and motivation for research.

TABLE V
RESEARCH QUESTIONS AND MOTIVATION

Research Questions	Motivation
How can geographically dispersed CSPs connect with each other optimally in a non-federated fashion?	Topology construction and optimization for the intercloud is a gap in the domain. Hence, there is a need to study suitable mechanisms to connect CSPs efficiently in an optimal manner to reduce communication overheads.
How can resource information be determined to be authentic, stored immutability, is non-repudiable and made available to all participating CSPs in an unambiguous manner in the absence of a centralised resource repository?	For the non-federated intercloud, viable resource discovery and management schemes need to address these critical challenges. Existing researches have not adequately addressed these issues. GICTF, the industry body, while recommending the use of P2P networks for designing non-federated intercloud, is silent on addressing these questions.
How can transactional records be managed securely without the need for a trusted third-party?	For the non-federated intercloud, existing research suggests using some centralised elements to manage transactions between participating CSPs. To realise the vision of a truly decentralised non-federated intercloud, it is imperative that a viable alternative alleviating the need for the trusted third-party is devised.
How can resource discovery in the non-federated intercloud be improved qualitatively?	It is important for participating CSPs to trust available resource information and have some visibility around the past performance of prospective partner CSPs. Further, meaningful trust and reputation metrics that CSPs cannot manipulate need to be devised. This will help in significantly enhancing the quality of resource selection leading to higher QoS. Existing research has been limited to defining weighted formulae for customised resource selection.

3.3 DATA SOURCES

A comprehensive review of related research papers with significant citations about intercloud resource discovery was taken into consideration, although the work in this domain is still at its early stage. Standard peer-reviewed journal databases including IEEE Xplore, ACM Digital Library, SpringerLink, Google Scholar were used to search for the existing work done. Besides research articles, few white papers related to efforts done by industry standardization bodies in the domain of intercloud resource discovery were also studied.

3.4 SEARCH CRITERIA

Search using keywords like "Intercloud Resource Discovery", "Resource Management in the intercloud", "Resource Exchange in Distributed Systems", "Cloud Resource Orchestration", "Security Issues in Distributed Resource Discovery" and other related keywords as shown in Fig.3.1 were used.

Possible Search Strings
Keyword ={ "Intercloud Resource Discovery", "Resource Management in the intercloud", "Resource Exchange in Distributed Systems", "Cloud Resource Orchestration", "Security Issues in Resource Discovery", "Resource Selection and Provisioning", "Automated Resource Discovery", "Authentication Challenges in the Intercloud", "Resource Discovery Models", "Resource Modeling", "Accounting Management in the Intercloud", "Reputation and Trust Management in the Cloud", "Resource Management in Grids/Mobile/Ad hoc Networks", "Optimization in Resource Discovery", "Multi-factor Optimization Strategies"}

FIGURE3.1. Used search strings

3.5 CRITERIA OF INCLUSION AND EXCLUSION

A total of 215 articles were excluded based on low relevance to the intercloud domain and, more specifically, to resource discovery. Most of the research papers were about different aspects of resource management in Grid Computing, P2P Networks, Cloud Computing and Ad hoc Networks. Finally, research articles were shortlisted related to the resource management in the intercloud domain, out of which few papers were directly related to the intercloud

resource discovery process. Few white papers pertaining to efforts done by Industry Standardization Bodies in the domain of intercloud resource discovery provided a strong motivation for the work done in this paper.

4. SYSTEM MODEL

Fig. 4.1 presents a detailed schematic of the BIRD framework which encompasses a dispersed cloud service providers (CSPs) network connected in a P2P fashion. All CSPs maintains a copy of the decentralized and distributed intercloud ledger (blockchain) that facilitates resource discovery and agreements that materialize with other CSPs. Smart contracts [41] which are code segments embedded in the intercloud ledger allow CSPs to enter into contracts when their specified requirements, both qualitative and quantitative, are met. These contracts contain the rules for negotiating the terms of the agreement, automatically verify fulfilment, and then execute the agreed terms between CSPs. CSPs also provides independent standard virtual machine (VM) to remotely install the BIRD performance measurement services (PMS). PMS monitors the real-time performance metrics that includes response time, reliability, availability etc. of CSPs by implementing synthetic workloads over a sustained period of time. Metrics are then used to calculate the compound metrics like QoS, Trust and Reputation (QTR). These along with cost comprise the CQTR metrics of a CSP for ensuring the resource discovery and selection process within the intercloud. This helps in building the historical performance profile of each CSP and its performance patterns for future deals and contract settlements. The CSPs in the non-federated intercloud share and maintain same view of the shared ledger which records time-stamped and encrypted transactions, CQTR metrics and contract information of the CSPs. Resource information is encapsulated in resource advertisements. CSPs participating can safely track the resources advertised through resource advertisements (RA) and resource requests (RR) by other CSPs which can help in optimal resource discovery based on multiple parameters and customised selection criteria.

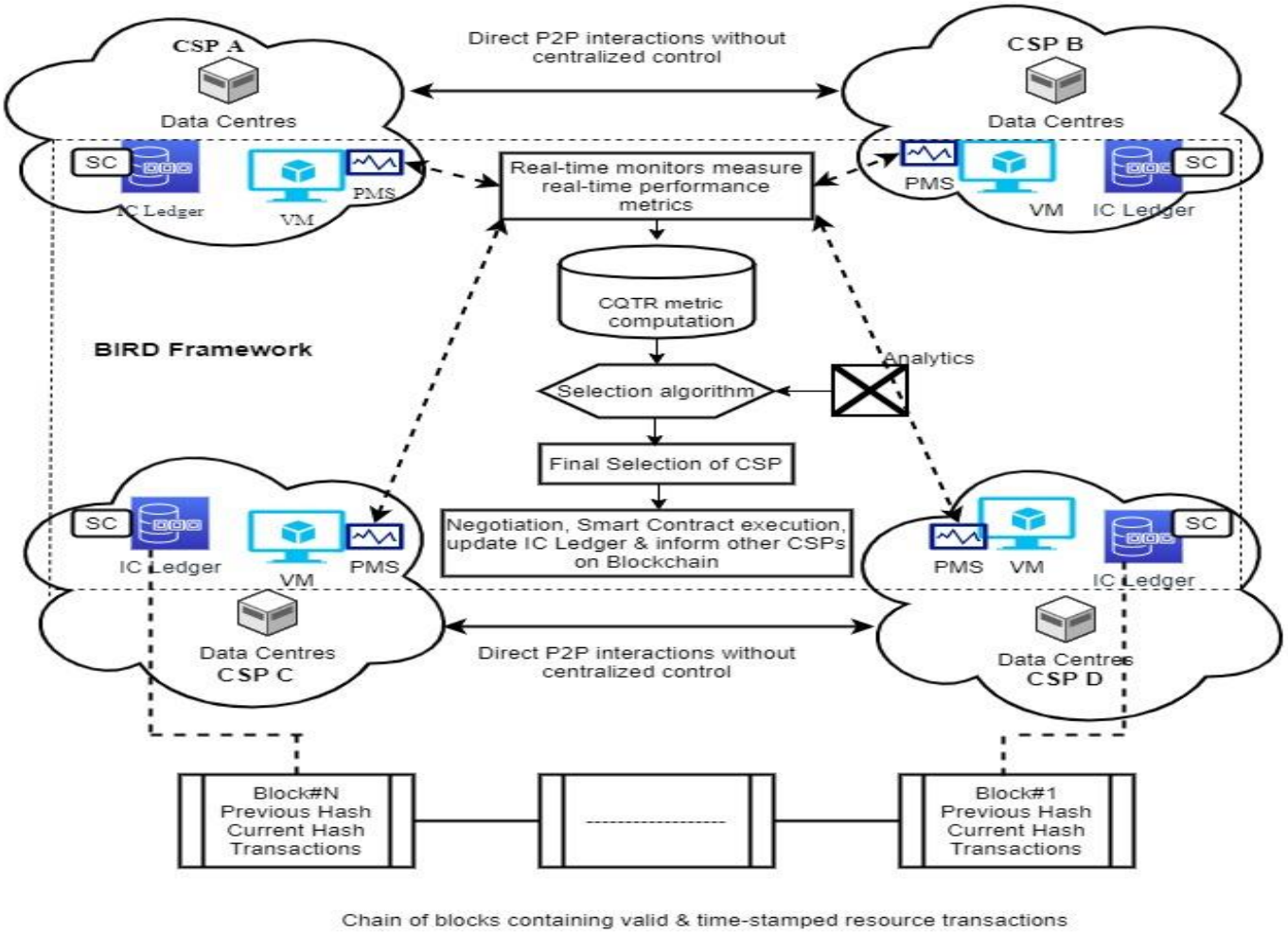

FIGURE 4.1. Conceptual representation of BIRD framework

The use of blockchain as a shared ledger resolves many security related issues including verification & validation, ensuring non-repudiation, data immutability and mitigating the impact of malicious CSPs in the intercloud. The BIRD framework further computes and utilizes cumulative QoS, Trust and Reputation metrics for each CSP to make

informed resource selection decisions. Other CSPs can fetch historical information both qualitative and quantitative in one lookup and hence there is no need to traverse the entire blockchain to check previous transaction records and historical performance information.

Mainly the CSPs in BIRD framework are responsible for the following:

- Organizing into latency optimized P2P network which is based on Random Landmarking [42, 43] to speed-up the resource discovery process.
- Designating weights in order to customize the resource selection process based on desired parameters like cost, quality-of-service parameters, trust and reputation from prospective partner CSPs.
- Forwarding Resource Requests (RR) and Resource Advertisements (RA) such that each CSP over the network is responsible for advertising the resource information or requests to other neighbouring CSPs.
- Verifying and validating the resource requests or advertisements that are received from other CSPs which prevents malicious behaviour in terms of misinformation being propagated through the intercloud.
- Negotiation of resource contracts (also termed as smart contracts, digital contracts or blockchain contracts) with other CSPs that are based on the resources provided and resources expected along with the associated performance, quality and cost constraints. Once the contracts are finalized, the CSPs interact independently and record the final transaction on the IC Ledger (intercloud ledger) which gets propagated to other CSPs subsequently so that the entire network can converge.
- Consensus management by ensuring that ledger transactions are synchronized across the P2P network and hence the ledger is updated only when the final contracts are established and verified by the appropriate CSPs. If any malicious activity or misreporting is detected, the contracts are immediately terminated and CSPs blacklisted.
- Settlement with contracted CSPs post conclusion of contract and presentation of bill of work by the resource providing CSP to the resource availing CSP.
- Calculation of CQTR (Cost, Quality-of-Service, Trust, and Reputation) metric post conclusion of the contract, updating cumulative values and recording them in the ledger by both participating CSPs.

CSPs in the BIRD framework can perform two different roles; behaving as a non-validator CSP or validator/forger CSP each performing certain set of operations as shown in Table VI.

TABLE VI
ROLE OF CSP IN BIRD FRAMEWORK

Non-Validator CSP	Validator/Forger CSP
• Self-organize into optimized P2P network	• Verify & validate Resource Advertisements (RA) or Resource Requests (RR)
• Broadcast Resource Advertisements (RA) or Resource Requests (RR)	• Serve as Escrow Account for holding the stake
• Execute smart contracts to finalize agreements, financial settlements & presentation of bill of work	• Consensus management for IC Ledger synchronization

4.1 RESOURCE MODELLING

Let CSP_i (i= 1, 2, 3... N) be the set of N cloud service providers comprising a non-federated intercloud environment. Each CSP further comprises up to M data centres (DC) such that;

$$CSP_i = DC_{i1}, DC_{i2}, DC_{i3} \ldots DC_{iM}$$

The promotion of resource availability by CSPs or search for resources from other CSPs is done through the following constructs:

1. ***Resource Advertisement (RA)*** broadcast by the CSPs desirous of sharing resources with other CSPs.
2. ***Resource Requests (RR)*** transmission by CSPs for making use of resources from the other CSPs.

Let RA_i be the resources advertised or broadcasted by CSP_i and RR_i are the resources needed by CSP_i within the framework. We can express resource advertisements and requests as 5-tuple as shown below;

$$RA_i = \{RO_i, P_i, Qa_i, Ta_i, Ra_i\}$$

$$RR_j = \{RN_j, C_j, Qr_j, Tr_j, Rr_j\}$$

where,

RO_i = Resources offered by the CSP_i

P_i = Price of resources

Qa_i = Quality-of-Service index of CSP_i

Ta_i = Trust index of CSP_i

Ra_i = Aggregated Reputation-Index of CSP_i

RN_j= Resources needed by the CSP_i

Cj = Expected Price of resources

Qr_j = Expected Quality-of-Service index of CSP_i

Tr_j = Expected Trust index of CSP_i

Rrj = Expected Reputation-Index of CSP_i

While modelling resource exchange in intercloud on the IaaS model with VMs being the main resource for exchange, the proposed framework is equally applicable to the PaaS and SaaS model as well. If we consider resources in terms of raw VMs (virtual machines) then RO_i can be represented as shown below;

$$RO_i = {}^{x}\sum_{y=1} VM_y$$

where,

x is the total amount of VMs offered by the CSP_i at specific time which can vary according to the internal resource demand and availability.

Further,

- Trust index measures the trustworthiness of a CSP by measuring its availability and past record of transactions or Transactional Success Ratio (TSR). The higher the availability and more the number of concluded transactions with a diverse set of peers, the higher is the trust score.
- Quality-of-Service Index is the average of QoS conformity for a particular CSP. As illustration, considering a QoS index of 0.95 would specify that the CSP achieved 95% total QoS compliance on parameters such as response time, latency etc.
- The Reputation index for a particular CSP is a long-term metric which is calculated only after certain numbers of transactions are performed by a CSP. Hence, reputation of a CSP considers both the QoS provided by CSP and its Trust-Index score over a pre-defined time period.

In a similar way, Resource Request advertisements are published by various CSPs that require resources with required specifications, including Cost and QoS requirements, where:

RR_j = Resources required by CSP_j

RR_j can also be further indicated in terms of raw VMs (virtual machines) as;

$$RR_j = {}^{k}\sum_{j=1} VM_j$$

where,

k is the sum total of number of VM's required by a CSP at a particular time.

The main aim of CSP requesting resources is to find another provider CSP that meets the requirements subject to the fulfilment of certain constraints as specified in smart contracts. Smart contracts remove the reliance on third-party entities allowing the CSPs to transact independently with each other. The resource requests which the CSP does not service are flooded in a controlled manner over the P2P network of CSPs. Out of the CSPs that respond to the resource advertisement, the one that can fulfil the required constraints per the weighted formula is selected. For each negotiation between CSPs, multiple commits are done on the shared ledger. First, when the contract is established, second, when the resources are consumed by CSPs and finally, once they are released. Along with the resource information, the CQTR value of each CSP is also a part of the resource advertisements.

For each CSP, the selection criteria as specified in smart contract can be expressed as;

$$CSP_{List} = \{W_1 * (C) + W_2 * (QI) + W_3 * (TI) + W_4 * (RI)\}$$

where,

C = Cost-value

QI = Quality-of-Service Index

TI = Trust Index

RI = Reputation Index

W_1, W_2, W_3 and W_4= Weights whose value range between 0 and 1

As can be seen from above, the individual weights assigned allow CSP's to create a customised selection process with fine-grained control for each selection criteria. This greatly impacts performance, quality-of-service, and overall cost and helps in selecting a particular CSP and its associated resources optimally. Thus, the quality of the CSP selection process is a key to achieving the desired cost to performance ratio. The proposed framework incorporates cost, quality-of-service index, trust and reputation as key parameters evaluating the quality of CSP resource offerings.

4.2 CQTR METRIC COMPUTATION

The optimized resource selection methodology in BIRD is based on a weighted formula of CQTR (Cost, Quality-of-Service, and Trust & Reputation) metrics monitored by PMS installed on each CSP, enabling customized resource selection. In a non-federated or public intercloud environment, CSPs often interact with each other without having assurances about their genuineness or the quality of resources on offer. There is also insufficient information for deciding which resources to select. Besides data on past performance, detailed interaction history leading to computation of trust and reputation metrics are needed to support informed decision making. The open and dynamic nature of the intercloud and the independent capacity planning and provisioning of resources within each CSP make resource discovery and sharing in the intercloud environment a challenging task. Finally, the dynamic resource requirements of the resource selecting CSPs require that CSPs need fine-grained control over the resource selection process. BIRD ensures customized resource selection by using the concept of Weighted Moving Average (WMA) [44] of CQTR metrics. The weighted moving average is a popular trend analysis indicator that can ensure that the oldest data points are dropped and the recent data points are factored in so that optimal resource selection can be made based on the latest performance trends. Cost is the amount paid or spent by CSPs to buy or obtain the resources required. The discussion about other QTR metrics is given in detail below:

a) QUALITY-OF-SERVICE (QOS)

Finding the best-suited CSP requires a trade-off among many parameters, including latency, quality of service warranties, cost, and past behaviour of CSPs etc. However, QoS-compliant resource discovery models for the intercloud find limited reference in literature. The RESERVOIR model for open federated cloud computing, as suggested by Rochwerger et al. [45], only concentrates on managing the server virtualization at the expense of other equally essential performance parameters. Business-oriented federation

model for real-time applications proposed by Yang et al. [46] considers only the critical requirements for real-time applications. This requires building federation mechanisms for provisioning resources across cloud service providers to deliver on-demand, cost-effective and QoS-oriented services.

Multi-parameter model in BIRD for QoS calculation is represented as,

Q = {R, P, L}

where,

R= Reliability,

P = Processing Time

L= Latency

- ***Reliability (R)*** includes the CSP's ability to perform under defined conditions for a certain period of time without failure. It can be represented as;

  $R = 1- n/M*t$

  where,

  t =Time for which resources or services are provided by CSP to other CSPs

  n = Time that includes CSP's unavialbility

  M = Time of operation or observation

- ***Processing Time (P)*** measures the time a CSP takes to execute the resource requests. It can be represented as;

  $P = {}^{M}\sum_{j=1} (\beta/\lambda) / M$

  where,

  β = Minimum processing time discovered during observation interval

  λ = Average processing time observed

  M = Number of requests handled by a CSP in a time slot

- ***Latency (L)*** is a complex metric in non-federated open public intercloud environment mainly because of unpredictability and huge geographically dispersed locations which calls for its continous measurement.

These parameters are continuously measured and monitored by the BIRD Performance Management Service (PMS) installed on a sample VM assigned for each CSP in order to calculate the QoS value for a CSP in an automated manner.

b) TRUST

Trust is an essential criterion in a non-federated intercloud environment since the CSPs need to know whether the resources they are accessing are genuine and provided by a trustworthy CSP. Existing mechanisms in the intercloud rate a CSP based on the cumulative past experiences of transacting cloud service providers. Such mechanisms can be circumvented by colluding CSPs. A multi-faceted trust management system, as explained by Habib et al. [47] helps to differentiate between a trusted and un-trusted cloud service provider. Its main contribution is that end-users can select the attributes using which trust ratings can be calculated. A framework for Trust-as-a-Service is proposed by Noor et al. [48], which explains an adaptive credibility model used to assess trustworthiness and to distinguish between credible and malicious feedback. The majority consensus feedback provided by the consumers is used to calculate the trust of a cloud service. However, such schemes are unable to eliminate user-bias ratings. BambooTrust is a scalable and distributed trust management system described by Kotsovinos et al. [49] based on an existing model named XenoTrust and the Bamboo distributed hash table. The framework defines a set of rules to decide how to evaluate the reputation information collected from different parties. Abawajy [50] suggest a reputation-based trust management system for an intercloud environment where a resource manager is responsible for provisioning and allocating resources and maintaining trust information for all the clouds. However, such a scheme is not feasible for a non-federated intercloud environment with no central authority. Filali et al. [51] presented a trust model based on the QoS and CertainTrust model. The trust value of a CSP is constituted by using two factors, i.e., Trust and Performance value. The user first sends the request to a cloud provider and then the calculation of the initial global trust value is done. The transaction is endorsed if the value is at a higher level above the threshold value. Trust models for distributed systems are typically policy, recommendation and feedback based.

Many limitations exist in the proposed trust models, especially regarding their applicability to the non-federated intercloud model and specific requirements of the BIRD framework. In most proposed models, there is reliance on a trusted third party like a broker or a cloud trust authority to facilitate trust management. Trust models based upon reputation and feedback, biased ratings and collusion among malicious peers can lead to wrong trust perceptions. In many approaches, subjective techniques allocate weights to trust factors like an expert opinion, user experience etc. The subjective weight assignment models do not always accurately calculate trustworthiness since they fail to address the complexity and dynamic adaptability involved in calculating trust.

BIRD addresses these challenges by completely automating trust calculation rather than focussing on user ratings or feedback and working on quantifiable continuous measured metrics. Based on the trust calculation, the long-term reputation of individual CSPs is determined. The computed trust index is further stored in the blockchain preventing collusion to manipulate the trust index artificially.

CSP trust calculation is based upon the Availability (A) and Transactional Success Ratio (TSR) that includes the record of transactions performed by a CSP. This is explained below as:

- ***Availability (A)*** of a CSP can be calculated by measuring the time duration for which the CSP remains down relative to the total operational time of CSP. It can be represented as;

  $A = 1- (t_{down} / t_{up} + t_{down}) \leq 1$

where,

t_{up} = CSP's up-time for a particular period of time

t_{down} = CSP's down-time for a particular period of time

- ***Transactional Success Ratio (TSR)*** involves the correlation between the number of successful resource transactions carried out by a CSP to the total number of resource transaction requests received. It can be represented as;

TSR = Number of successful transactions /Total number of transactions

Hence, BIRD calculates trust-index of a CSP as;

$$T_{CSP} = A * TSR$$

c) REPUTATION

Trust computation within the BIRD framework depends upon the moving averages of observed parameters, whereas reputation calculation is seen as a measure of long-term trust. Hence, the reputation index is assigned to those CSPs who have a track record of successful transactional history. So, long-term trust indices plus the transactional record of a CSP comprise its reputation index. The reputation index is also utilized to aid in the qualitative selection of CSPs, especially when the long-term trustworthiness of a CSP is required for critical resource requirements.

The reputation index of a CSP in the BIRD framework can be calculated by taking the product of the cumulative value of the trust index and the value of the QoS index such that,

$$R_I = {}^{n}\sum_{t=0} Q_I * T_I$$

where,

Q_I = Quality-of-Service Index

T_I = Trust Index

4.3 BIRD CONSENSUS MECHANISM

The BIRD framework employs a consensus algorithm [52] to confirm the CSP-to-CSP transactions and ensure that all participating CSPs have the same view of the contract information and real-time resource information, besides performance information. Excluding the need of the intermediaries and enabling a genuinely decentralised non-federated intercloud model is the key contribution of the BIRD framework. Proof-of-Work (PoW) [39] is a distributed consensus mechanism used in blockchain-based cryptocurrencies to earn the privilege to confirm transactions, generate new currency and create new blocks added to the chain. But the significant drawbacks of the PoW consensus mechanism which render it infeasible for use in the BIRD framework include:

a) Energy Consumption: due to the bulk of computational power needed to test millions of transactions per second by 'miner', PoW is highly costly and energy intensive.

b) Vulnerability: due to 51% attack i.e. malicious miners can take over 51 percent of the network's computing power and can dominate the network leading to manipulation in the blockchain.

Hence, the BIRD framework innovatively utilizes the Proof-of-Stake (PoS) [40] consensus algorithm, which benefits increased efficiency. Each CSP provides its proof-of-stake in the form of its CQTR metrics, which accompany the resource advertisements. The higher the CQTR metric, the more stake a CSP exerts in the network for validating transactions and serving as a forger node, having the right to create new blocks and making modifications to the shared ledger. CSPs with lower stakes give way to those with higher stakes to enjoy privileges to act as validator nodes. This incentivizes CSPs offering a better quality of services and enjoying higher trust ratings.

4.4 BIRD Smart Contracts

The automated contract execution aims at selecting the most suited CSP among the available options. This encompasses multi-factor selection involving the CQTR (cost, quality-of-service, trust and reputation) compound metric calculation using Analytic Hierarchy Process (AHP) [53] algorithm providing high-quality CSP/resource selection. AHP evaluates the weighted scores based on CQTR metrics of all corresponding CSPs. The weights assigned to each criterion enable customized selection of resources depending upon the changing needs and priorities of a CSP. Values for these weights vary between 0 and 1 to indicate the relative importance of each parameter to a CSP allowing it to rank prospective CSPs and select the highest rated one. Smart contracts get automatically executed and the top-ranking CSP is contacted for establishing the contract.

4.5 BIRD Security

Security concerns are one of the significant issues for the intercloud resource discovery. The various existing security provisions include authentication, identity management, data security etc. Specifically, from a resource discovery perspective, distributed authentication, insecure communication channels leading to man-in-the-middle attacks, malicious users and non-fulfilment of contracts remain major challenges. BIRD significantly addresses the security challenges in resource discovery within the intercloud environment. The summary of the security features provided by BIRD is provided below:

- Computation of CQTR score is entirely automated, based on real-time measurements obtained by the performance measurement services (PMS) installed at each CSP and stored in an encrypted form, ensuring non-repudiation. Further, the calculation of the reputation score of a CSP does not depend upon assigned ratings by other CSPs. Instead, it is calculated automatically only after a certain number of successful transactions concludes. This prevents malicious users from acquiring reputations without being a part of genuine transactions.
- Automating the reputation calculation also ensures that participating CSPs cannot collude to create artificially high reputation scores. Finally, the proposed scheme removes the possibility of any bias to creep into the reputation scores.

- CSPs which fail to fulfil contractual obligations face termination of the contract, blacklisting by prospective partner CSP.
- The use of blockchain is novel and significant as it records all the information in the shared ledger maintained at each CSP and hence ensures authentication, immutability, non-repudiation, transparency and removal of dependence on centralized third-party for financial settlements and dispute resolution.

5. BIRD OPERATING MODEL

The BIRD framework encompasses a set of protocols and associated algorithms facilitating the CSPs to join the intercloud, stake claims to serve as validator nodes, send and receive resource advertisements, specify selection criteria, execute smart contracts and finally enter into the contract with another CSP. The details of major operations and their sequence within the BIRD framework are described below:

a. JOINING & SELECTION OF CSPs: Creating a P2P network of CSPs for direct communication without requiring a pervasive third-party need to solve two significant problems: CSP identification and CSP neighbour selection. Identification enables distinction between CSPs and is made by assigning each CSP a unique Globally Unique ID (GUID) [54] when joining the P2P network. Locating the ideal CSP to connect with means finding peers by referring to a list maintained by the BIRD framework that contains information about the already joined CSPs in the intercloud. The selection benchmark for a prospective neighbour, CSPs in BIRD, includes minimum latency by possibly selecting the nearest CSP. Usually, geographical proximity, indicative of lower communication latency, is the most apparent choice while selecting the neighbours from the list of CSPs. Random Landmarking used for static networks [42, 43] and Mobile Adhoc networks [55] are popular techniques for building a latency-optimized P2P network. BIRD makes use of this Random Landmarking strategy, using well-known landmark nodes to structure themselves into the group of peers that are in close physical vicinity to one another, hence overcoming latency issues.

The BIRD framework performs the role of the boot peer, also known as bootstrapping node [56], for kick-starting the formation of the P2P network. Subsequent CSPs joining the P2P network use the BIRD framework to obtain information about other CSPs to select the best neighbour to join. This is done to construct a P2P overlay network connecting CSPs that are latency optimized, i.e., the physically close CSPs clustered together. The joining CSP first needs to register with the BIRD framework. After that, it gathers important information regarding its neighbourhood (against some landmark nodes) by sending "Join Request". The BIRD framework acknowledges the request by sending the joining peer a GUID (for identification) and a list of landmarked nodes. The joining CSP then uses this information to measure the distances to the temporary landmark node by sending ping requests to determine which landmarking node is it closer to. The landmark node provides it with a reference of one CSP already registered to connect to physically. The sequence diagram explaining the various steps of the CSP joining phase are shown in Fig. 5.1.

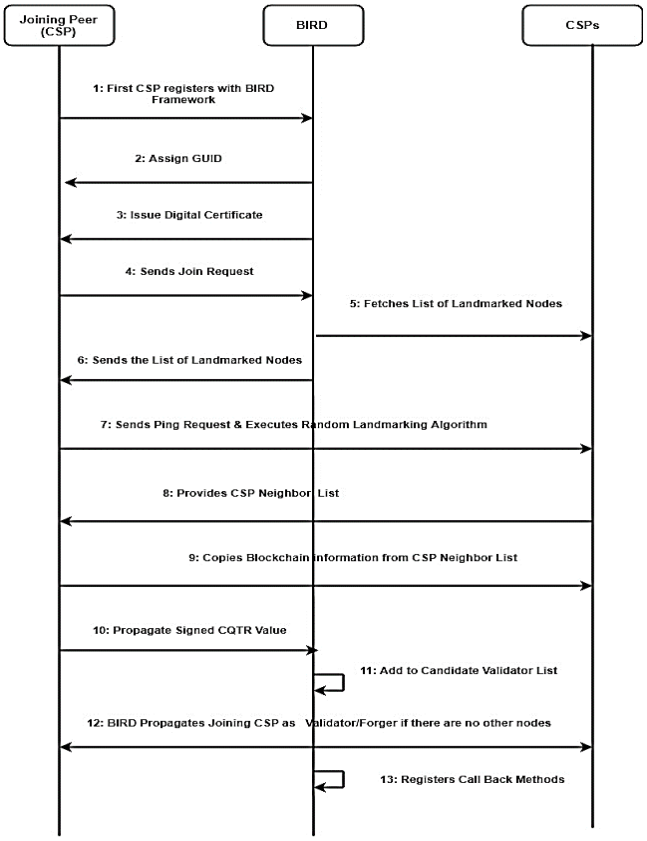

FIGURE 5.1. CSP joining phase

Consequently, the physically close CSPs are clustered together in the network using topology optimization in a non-federated ecosystem of CSPs. It is seen that, on average, the Random Landmarking algorithm provides an overlay to physical hop distance ratio of 1:1.6 for networks for up to 10,000 peers. As soon as a new node joins the network, it copies every block history (containing valid transactions) along with smart contract codes and contracts history maintained in the ledger of the connecting CSP. For real-time monitoring of CSP, BIRD performance measurement services (PMS) is installed on each CSP. This is an offline process.

The newly entered CSP can now receive many Resource Advertisements (RA) and at the same time can send Resource Requests (RR) from/to other CSPs. A trusted interaction model is supported by BIRD, under which all the CSPs acquire a digitally signed certificate [57] (only one time) from the BIRD Framework to confirm its identity & credentials.

b. BROADCASTING ADVERTISEMENTS (RR/RA): Using flooding protocol, CSPs propagate resource advertisements/requests to other participating CSPs

within the BIRD framework. PMS measures and records signed CQTR score for each CSP which is included in the RR or RA. Propagation of RR/RA happens within the specified time frame and in a supervised manner. Once the expiration of the time window happens, old RR/RA are discarded by CSPs to make sure that stale copies are not circulated within the network. Hence, only validated RR is added to the response list, which is within the specified time window and based upon them, the smart contract is invoked. Also, they are added to the response list when the advertising/claimant CSP's validated RA contains more available resources than the other CSP requested and met the desired criteria. The sequence diagram for the steps involved in processing the advertisements and resource requests is shown in Fig. 5.2.

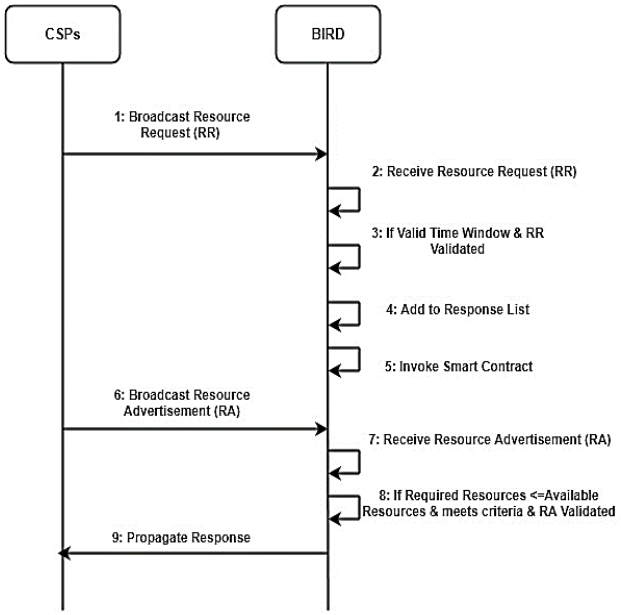

FIGURE 5.2. Processing of RR/RA

c. VALIDATOR & FORGER SELECTION: PoS algorithm is used in BIRD that achieves consensus by requiring each participating CSP to stake signed QTR metrics along with cost value to get a chance of being selected as a validator for validating RR/RA, acting as a forger to write blocks in the blockchain and finally get incentives in terms of higher reputation score. The incentive mechanism encourages CSPs to act as validators and earn higher scores to increase the chances of successfully entering into contracts with other CSPs by meeting their selection criteria. Validator selection within the BIRD framework necessitates examining the CQTR score of the claimant CSP and inspecting the global selection cycle of validators. Weights are adjusted for each selection cycle to randomize validator selection. Each validator-CSP is selected by comparing the QTR values and cost metric amongst the various CSP claimants and finally adding the selected validators to the validator list maintained by the BIRD framework. The newly added validator in the list is propagated to all the other validators so that all CSPs have the same information.

Fig.5.3 presents the sequence of steps involved in validator & forger selection.

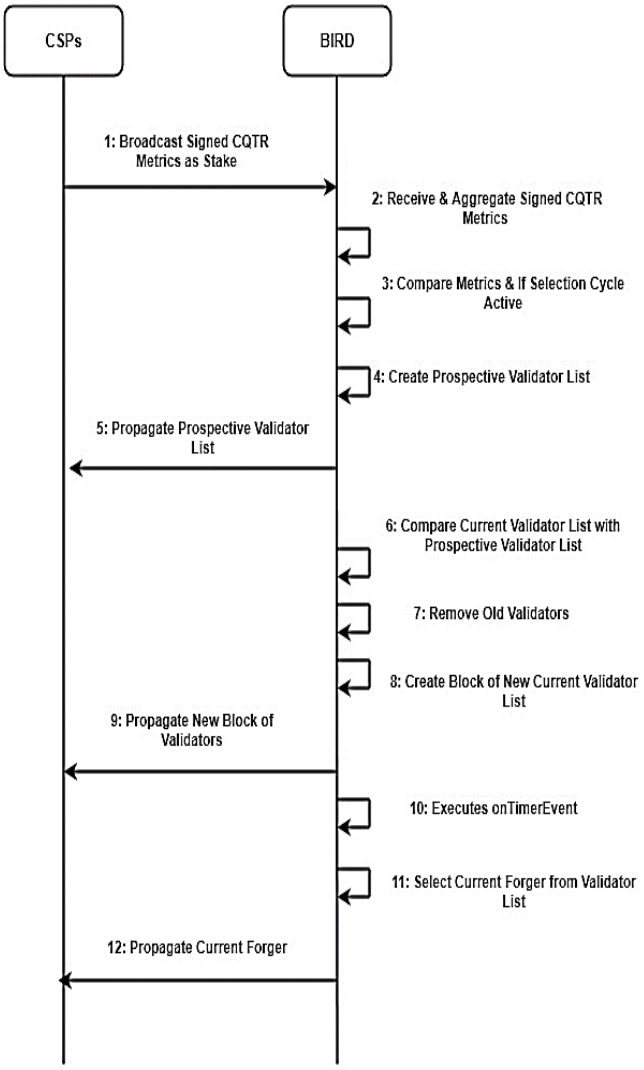

FIGURE 5.3. Validator & Forger selection

BIRD also provides a unique feature of randomizing validator selection to remove duplicates by merging the validator lists created and ensuring that about 75% of the old validators are removed from the current validator list and only the new one or top-ranked ones are added to the validator list at any time. Randomizing of weights is done within each selection cycle for selecting validators, making it non-deterministic. In contrast, the deterministic approach forger selection uses a round-robin mechanism within the validator chosen list. After a pre-determined time, the forger selection cycle is executed, which selects a single forger from the validator list to ensure fairness, letting each CSP be selected deterministically from within the validator list.

d. VERIFICATION: BIRD performs verification of advertisements (RR/RA) which are required for future contract settlements between CSPs. The verification of advertisements broadcasted over the network are required to be verified by validator-CSPs and are done as follows:

a) The validity of the digital certificate and GUID for each CSP is ascertained by using the verification service provided by BIRD to ensure their credibility, preventing malicious peers from participating in BIRD framework.

b) To check whether the resource advertisement (RA) issuing CSP has the required number of resources as stated in the advertisement. This is done by checking the last valid resource transaction involving that particular CSP by performing a ledger lookup and retrieving the current resource balance for that CSP. If there exists a resource mismatch, the advertising CSP is placed on the blacklist to prevent it from participating in future contracts.

e. SMART CONTRACT EXECUTION: Once the validator-CSPs verify RA/RRs issued by CSPs, they are then propagated to other validators and subsequently to the rest of the intercloud network. Multi-factor selection or decision-making is provided by BIRD that involves the CQTR compound metric calculation using Analytic Hierarchy Process (AHP) algorithm. AHP is a leading approach in multi-criteria decision making that divides the objective or goal (high-quality CSP selection), available criteria (CQTR metrics) and available alternatives (CSP1.....CSPN) into a hierarchical structure as shown in Fig. 5.4. It is a useful decision support tool that provides a comprehensive and rational framework for structuring a decision-making problem.

Prospective CSPs receive the validated RA/RRs from various CSPs. Once the highest-ranking CSP is selected by using the AHP algorithm involving CQTR metrics, the "Request for Contract" from the requesting CSP and "Expression of Interest" from the responding CSP gets implemented automatically, which is done through the execution of their specific smart contracts.

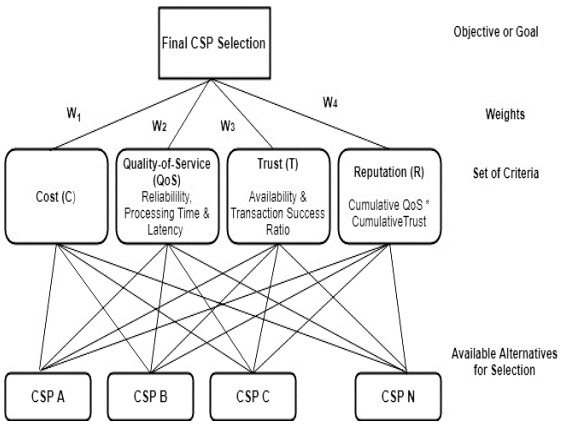

FIGURE 5.4. AHP hierarchy for high quality CSP selection in BIRD

Contracts are triggered between CSPs and help in exchanging resources. After the contract has been established between the contracting CSP and the contracted CSP, the contract information is broadcast over the network to inform all the validators. Fig. 5.5 explains the algorithm used for smart contract execution using AHP that involves mainly the following steps:

- Evaluate all the responses received from various CSPs and determine the relative importance of alternatives concerning criteria for CQTR values by applying weights
- Generate a pair-wise comparison matrix for each criterion by developing a matrix containing pair-wise comparisons of alternatives and goals on each criterion
- Normalizing the matrix by dividing each entry by the sum of the column to get the appropriate weights and corresponding ranking
- Calculate the weighted average ranking for each alternative and finally selecting the one with the highest rank

smartContract (responseList) //while responses RR received from CSPs have been validated

1: while (isvalidated(responseList.getRR()))

2: addToList(RR, validatedResponseList)

3: FOR each RR in validatedResponseList

//Determine relative importance of alternatives with respect to defined criteria using scale of relative importance of weights ranging between 0 and1for the CQTR values contained in each RR

4: RR = validatedResponseList.getRR()

5: weightedRR = RR.applyWeights(mySelectionWeightsList)

6: addToWeightedRRList (weightedRR, weightedRRList)

7: END FOR

8: compMatrix = generatePairWiseComparisonMatrix(weightedRRList)

// calculate sum and normalized priority both for alternatives mapped with criteria and criteria mapped with goal

9: compMatrix.computeColumnSum()

10: compMatrix.performNormalization()

// calculate final priority by multiplying the weights score evaluated for criteria with all alternatives and summing the total score for each alternative

11: compMatrix.computeAlternatives(mySelection WeightsList, altWeightsList) //Rank the resource responses RRs from different CSPs after applying multi-factor optimization

12: rankList = compMatrix.rank()

//Initiate contract establishment with top ranked CSP

13: while (SUCCESS != TRUE)

14: topCSP = rankList.getCSP()

15: SUCCESS = invokeSmartContract(topCSP, contractTerms);

16: contracting CSP = this.CSP

17: contractedCSP = topCSP

18: END while

//propagate the contract information to all validators

19: Bird.informValidators(contractingCSP, contractedCSP, contractTerms);

20: END while

FIGURE 5.5. Smart contract execution using AHP algorithm

5.6 BLOCK CREATION: Forgers are finally accountable for creating a valid hashed block that records the contract details and writing it to the blockchain-based shared ledger. Two blocks may be created at the same time by the validators that contain similar recorded transactions. Still, only one block is added to the blockchain as they are routed to the forger, responsible for managing duplicates. Each block generated by the Forger-CSP is added to the blockchain-based shared ledger and propagated to the other CSPs. Thus, each CSP now shares a common view of the CSP-to-CSP transactions through the shared ledger. Information regarding successful completion and partial completion of contracts (due to technical or other challenges on the CSP side) is also recorded in the blocks and added to the blockchain shared amongst the CSPs. This information forms the basis for future resource selection and contract finalisation.

Fig.5.6 explains the sequence of steps for the final block creation and addition in BIRD. Searching for a specific block (containing the valid contract information) required by a Non-Forger CSP involves traversing all the blocks in the blockchain until a specific block is reached and hence the total time complexity involved is O (n^2), where n is the number of blocks with each block containing n transactions. It thus involves both block search and then transaction search containing contract information.

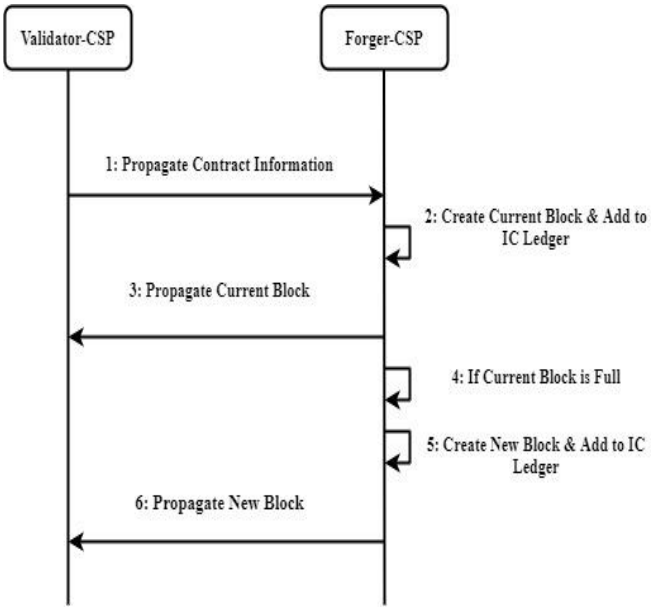

FIGURE 5.6. Addition of new block in shared ledger in BIRD Framework

6. RESULTS

The feasibility of the BIRD framework has been validated using a custom simulator. There are very few blockchain simulators that are available such as SimBlock [58], Bitcore [59], Tierion [60] etc., but these simulate crypto-currency based applications primarily. There are no generic blockchain simulators available that allow simulation of blockchain-as-a-service, necessitating the use of a custom simulator that would enable the modelling of CSP-CSP transactions and underlying blockchain protocols. The simulation was performed on a Dell Workstation, which has the following configuration parameters as listed in Table VII.

TABLE VII
CONFIGURATION PARAMETERS OF DELL WORKSTATION

Name	**Configuration**
Model	Dell T5500 Workstation
Processor	Intel Xeon Processor
Operating System	Red Hat Enterprise Linux WS v.5
GPU	`NVIDIA Tesla C1060
Hard Drive	SATA 3.0GB/s 7200 RPM with 16MB DataBurst Cache™ 1.5TB
Memory	16 GB
Graphics	NVIDIA Quadro 5000
Chipset	Intel 5520

The simulation parameters are described in Table VIII below:

TABLE VIII
SIMULATION PARAMETERS

Parameters	**Value/Range**
Number of CSP Nodes	100-500
Number of Validators	10-50
Number of Transactions	1000-5000

Validation Time Window	30 seconds
Validator Invalidation Interval	1 hour
Block Size	1MB
Validator Selection Window	1 minute
Consensus Algorithm	Proof-of-Stake

The class diagram for the BIRD simulator is shown in Fig. 6.1.

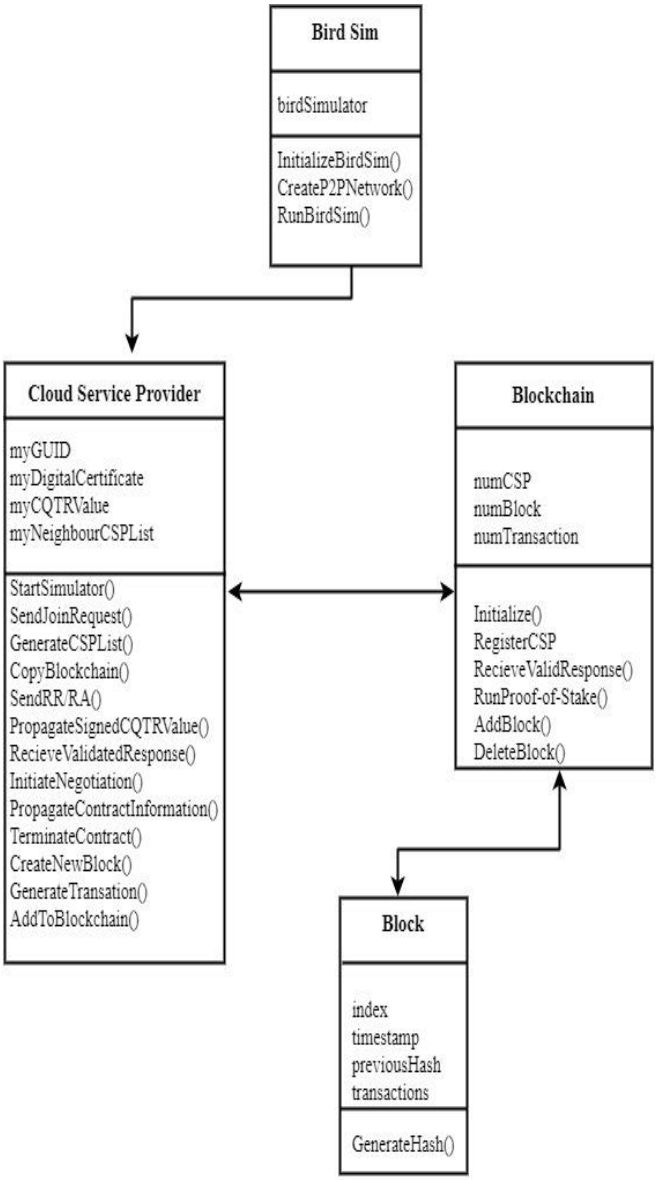

FIGURE 6.1. BIRD simulator classes

6.1 START-UP TIME

The start-up performance of the BIRD framework depends upon the number of participating CSPs. This experiment measures the start-up time for the entire BIRD framework comprising up to 500 CSPs. Start-up time includes the bootstrapping time of individual CSPs, using Random Landmarking scheme to self-organise, join time of each CSP and the initial blockchain set-up time (that includes establishing smart contract codes, cryptographic keys to be distributed among various participants) etc. With the increasing number of CSPs, there is a growth in the start-up time which starts out linearly and later flattens out for new peers as they are able to discover peers faster and in their vicinity. As can be seen from Fig.6.2, with the increasing number of participating CSPs the average start-up time per CSP reduces from 1.8 seconds (100 CSPs) to 0.84 seconds (500 CSPs). Total start-up time varies from 180 seconds (100 CSPs) to 420 seconds (500 CSPs)

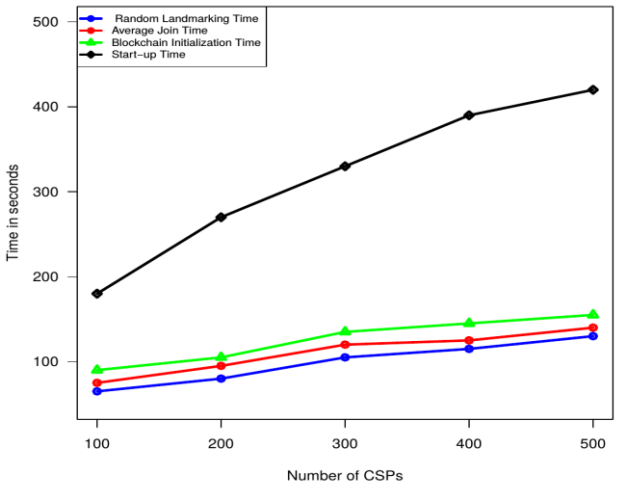

FIGURE 6.2. BIRD Start-up time

6.2 VALIDATOR SELECTION

Selection of validators required for validating the transactions, including RAs/RRs, involves the comparison of CQTR values of prospective CSPs through a network-wide contention process. Each CSP can potentially put forth its stake by sharing its CQTR value. The CSP with the lower CQTR value drops out of contention after comparison, while the victor CSP's CQTR value is forwarded for further comparison. The comparisons continue until the required number of validators remains. In this experiment, the validator selection time is measured for a fixed and varying number of CSPs. As shown from Fig.6.3(a), the time required to select ten validators from 500 CSPs takes 300 ms, while selecting 50 validators takes 190 ms. Results clearly show that the selection time for selecting more validators is less as the algorithm converges after fewer comparisons. When selecting fewer numbers of validators, more CQTR comparisons and packet transmissions take place to arrive at a global consensus.

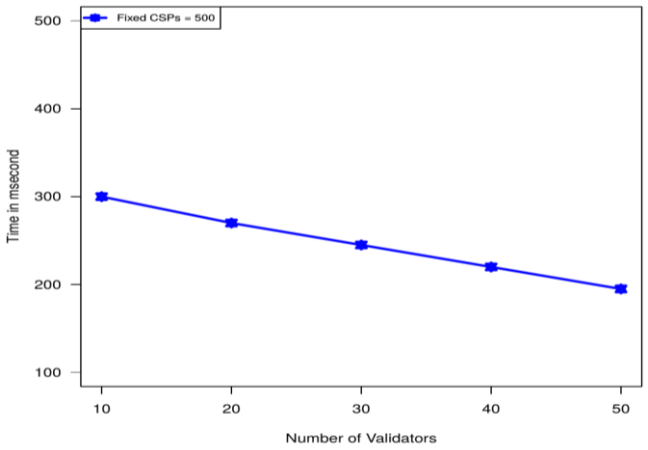

FIGURE 6.3(a). Validator Selection Time (VST) with fixed CSPs & varying validators

Fig. 6.3 (b) depicts the measurements taken for varying numbers of CSPs ranging from 100-500, and the selection time for selecting 10-50 validators was calculated. For instance, the time required to select ten validators from 100 CSP is 220 ms which is more than the 110 ms for selecting 50 validators. Similarly, selecting ten validators from 500 CSPs requires more comparisons than selecting 50 validators resulting in a time increase from 170msec (50 validators) to 290msec (10 validators).

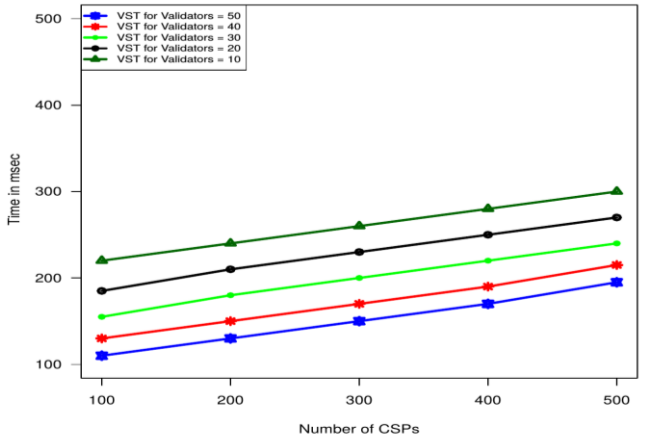

FIGURE 6.3(b). Validator Selection Time (VST) for varying validators with varying CSPs

6.3 VALIDATION TIME

Validation involves verification of resource advertisements and resource requests required for future contract establishments between CSPs. Hence, it is important to measure the number of validators concerning the number of CSPs on the validation time for the varying number of RAs/RRs. The following sets of operations are involved in calculating the validation time:

1. Verification of the digitally signed self-certificate and GUID generated by individual CSP
2. Checking whether the CSP which is issuing resource advertisement (RA) has the required number of resources
3. Searching within the blockchain by traversing the blocks and looking up for the transactions already added to check for CSPs trust and reputation.

In Fig. 6.4 (a) the number of CSPs was kept fixed (500) and the number of validators varied from 10-50, for which the validation time was calculated. By looking at the results, we can say that when the number of validators is increased with the fixed number of CSPs, the average RA/RR validation time decreases from 37 seconds with ten validators for 3000 RR/RA's to 20 seconds with 50 validators for 3000 RR/RA's. This is responsible for increasing the throughput of the system as a huge number of transactions can be validated faster, resulting in higher system throughput.

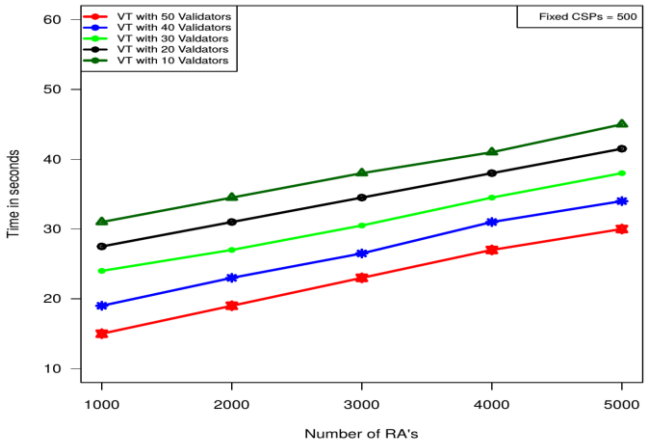

FIGURE 6.4(a). Validation Time (VT) for fixed CSPs (500) & varying validators

In the results from the second experiment, as shown in Fig.6.4 (b), the validation time was calculated for varying numbers of CSPs while the validators were kept fixed. As the population of CSPs increases while keeping the number of validators fixed, RA/RR validation time increases (14seconds with 100 CSPs for 3000 RR/RA's to 39seconds with 300 CSPs for 3000 RR/RA's).

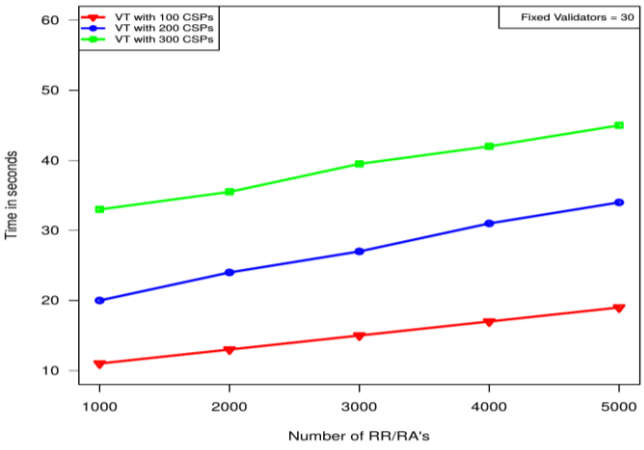

FIGURE 6.4(b). Validation Time (VT) with varying CSPs & fixed validators (30)

6.4 AVERAGE CONTRACT ESTABLISHMENT TIME

The contract establishment between CSPs involves the following steps:

1. Sending of RA/RR over the BIRD network
2. Receiving responses from multiple CSPs
3. Validation time required to validate the received responses
4. Selecting the best-fit CSP based on AHP algorithm
5. Smart contract execution

The experiment was carried out for a total of 5000 transactions flooded by 500 CSPs with 50 validators. As shown in Fig.6.5, while the number of transactions increases from 1000 to 5000, there is a significant increase in the average contract establishment time. It is because of the larger number of transactions that need to be validated and the time required by the validators to access the blockchain and verify the resources as claimed by CSPs goes up. The longer the blockchain, the lookup time increases proportionately, although the use of compound CQTR values with moving averages ensures that only one last transaction of the CSP needs to be looked up to determine the compound CQTR score and associated resource balance for a CSP.

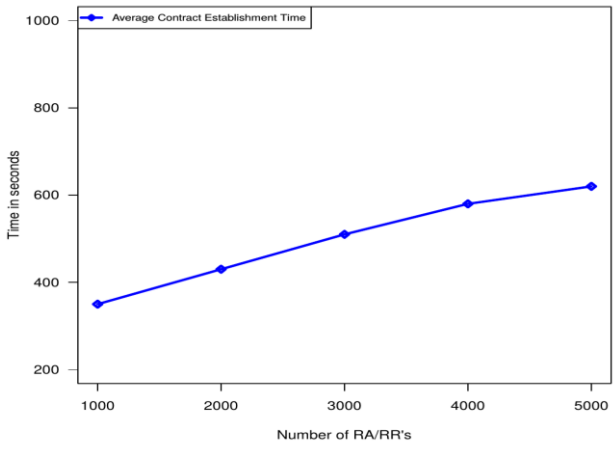

FIGURE 6.5. Average contract establishment for varying number of RR/RA & fixed number of CSPs and validators

6.5 BLOCK CREATION TIME

After the contract is finalized between the contracting and contracted CSP, the contract information is forwarded by any validator from the list to the forger and inform all the other set of validators. The forger writes the block of transactions to a previously created current block (if space is available) or creates a new block of transactions. The newly added block or the updated block is propagated until all validators and CSPs have the same copy. Fig.6.6 shows the new block creation time for an inter-cloud of 500 CSPs and a varying number of validators. The block creation time with 10 validators is 1.8 seconds, whereas with 50 validators, it is 5.5 seconds.

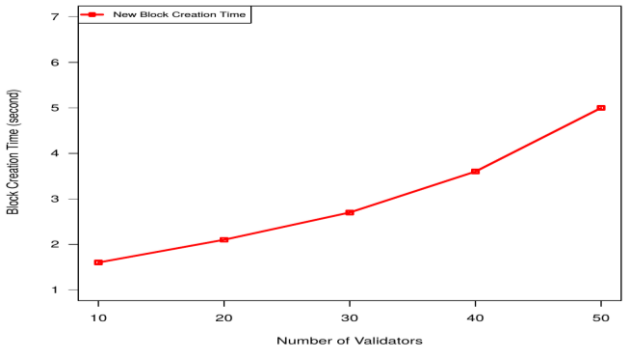

FIGURE 6.6. Block creation time

6.6 BLOCK PROPAGATION TIME

The propagation of blocks over the network happens in a P2P fashion, wherein each node propagates the block to the other set of nodes. The time needed to propagate the blocks depends upon the size of the inter-cloud, the number of validators and to a small extent on the size of the block itself. Once the block is created and written by the forger, it must be propagated to the other CSP nodes in the network. Fig.6.7 summarizes the result for the block propagation time for 500 CSPs and 50 validators. For different block combinations, the block propagation time varies. As seen for 200 CSPs, the block propagation time for three different block sizes ranges from 13 to 20 seconds. Hence, can be concluded that smaller block gets propagated faster than larger blocks.

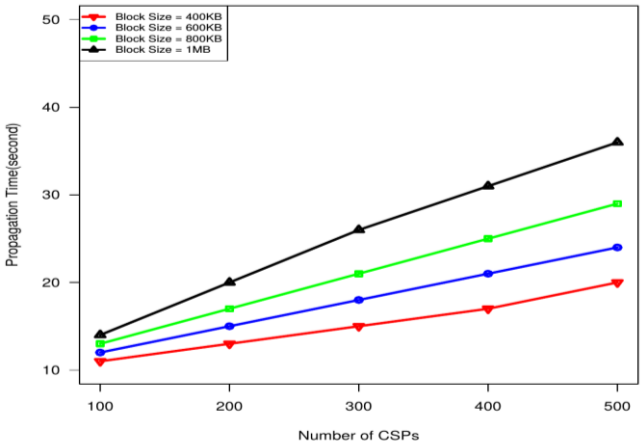

FIGURE 6.7. Block propagation time

6.7 LATENCY AND THROUGHPUT

The performance of the BIRD framework is measured in terms of latency and throughput (for RA/RR), considering the fixed number of CSPs over a period of 1 hour. Latency involves the time required to discover the validators and receive the resource responses. In contrast, throughput involves the number of transactions validated by validators and written to the block by the forger. In the first scenario, 500 CSPs were considered and the number of validators varied. A total of 1000 transactions were initiated. BIRD achieves the best latency and throughput figures when the number of validators selected is 50 as can be seen in Fig6.8 (a) and Fig.6.8 (b) for a fixed number of CSPs.

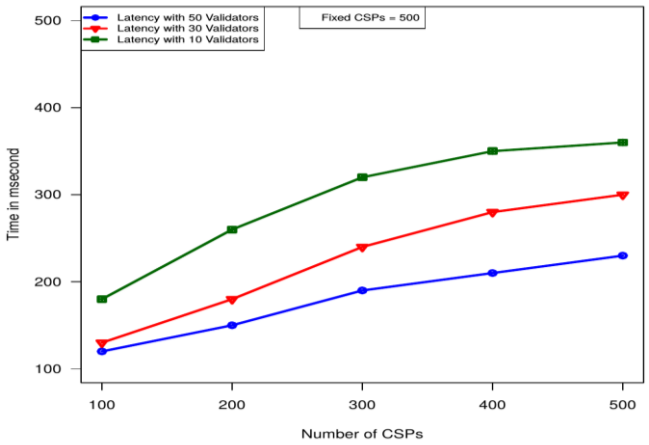

FIGURE 6.8(a). BIRD average latency for fixed CSPs

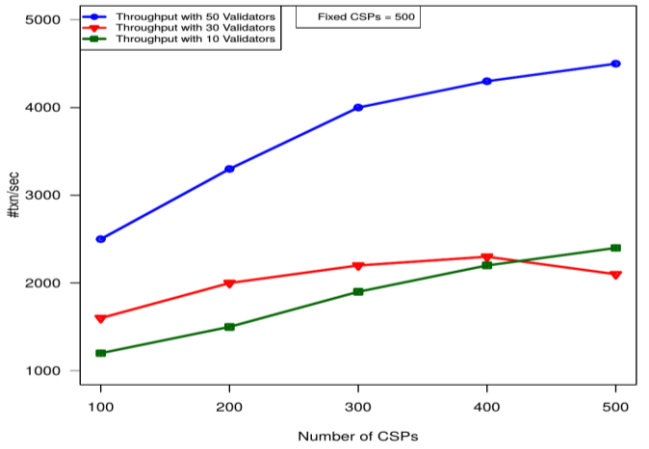

FIGURE 6.8(b). BIRD throughput for fixed CSPs

7. CONCLUSIONS AND FUTURE WORK

This paper proposes a resource discovery framework based on the P2P approach for the non-federated intercloud environment, which utilizes blockchain to enable CSP-to-CSP interactions facilitating collaboration and resource utilization without relying on a trusted third party. The framework addresses an important gap in the intercloud domain as specified by the GICTF[5]. The use of blockchain to record transactional details emanating from CSP-CSP interactions and smart contracts for automating contract establishment among CSPs has not been proposed in the existing literature. Hence, the proposed framework is novel. The major contributions of this research paper using the BIRD framework are summarized below:

- Overlay topology optimization of P2P network comprising CSPs to reduce mean latency for the intercloud.
- Qualitative selection among prospective partner CSPs using multi-factor optimization and fine-grained control mechanisms
- Automated computation of CQTR metrics for each CSP removing chances of misinformation, biases in computing trust and reputation and preventing collusion among malicious CSPs.
- Application of blockchain concept for securely tracking resource related information, recording transactions, providing immutability, transparency and overall security.

A detailed design of the framework has been proposed, and simulation results establish the viability of the proposed framework. Based on early results, it can be concluded that the blockchain concept applies to the non-federated intercloud model with high relevance and success. The present work details all aspects of resource discovery within the intercloud environment with optimal strategies and mechanisms to discover the resources effectively and efficiently.

Future work shall involve:

- Devising a comprehensive quality-of-service framework that enables performant resource provisioning in the face of faults and network reconfigurations.
- Predictive analytics is an excellent fit for the non-federated intercloud resource discovery model leading to just-in-time resource provisioning. CSPs can use historical resource requests and deep learning-based AI techniques to predict resource requirements and periods when their resources fetch a premium in the marketplace, leading to increased revenue generation. Building intelligence into the blockchain itself can make the overall system more performant. The blockchain can then optimize itself to deliver higher levels of service guarantees to the overlying CSPs. For instance, an elastic blockchain might optimize the number of validators and forgers required based on the transactional load on the system and the total number of CSPs. It can also maintain separate blocks containing references to CSPs engaged in frequent transactions to dramatically improve the performance of blockchain lookups and significantly improve validation time. Thus, a customized blockchain delivering a comprehensive resource-discovery-as-a-service for the intercloud domain is entirely feasible in the future. The next-generation of intercloud resource discovery models can be expected to be characterized by increased intelligence and be autonomous to a large extent.
- Real-world deployment and performance benchmarking to drive further optimization and analyzing the variant operations incorporated in the framework.

REFERENCES

[1] J. Weinman, "What's Next for the Cloud? The Intercloud," 2013, accessed: 2015-01-20. [Online]. Available: https://www.forbes.com/sites/joeweinman/2013/10/08/whats-next-for-the-cloud-the-intercloud-2/#3982dc62c7db

[2] D. Bernstein, E. Ludvigson, K. Sankar, M. Morrow and S. Diamond, "Blueprint for the InterCloud - Protocols and Formats for Cloud Computing Interoperability," *4th International Conference on Internet and Web Applications and Services*, IEEE, jun 2009, pp. 328-336. [Online]. Available: https://ieeexplore.ieee.org/document/5072540

[3] Intercloud, accessed: 2015-03-05. [Online]. Available: https://www.techopedia.com/definition/7756/ intercloud.

[4] SDxcentral, "What is the Definition of IEEE Intercloud? ", accessed: 2015-08-12. [Online]. Available: https:// www.sdxcentral.com/cloud/definitions/what-is-intercloud/

[5] GICTF, "Use Cases and Functional Requirements for Inter-Cloud Computing," accessed: 2016-05-13. [Online]. Available: http://www.Inter cloudtestbed. org/uploads/ 2/1/3/9/21396364/gictf_whitepaper_20100809.pdf,White Paper, 2010

[6] N. Grozev and R. Buyya, "Inter-Cloud architectures and application brokering: taxonomy and survey," *Software: Practice and Experience*, vol. 44, pp. 369–390, dec 2014. [Online].Available: http://www.cloudbus.org/papers/InterCloud-Brokering-Taxonomy.pdf

[7] AN. Toosi, RN. Calheiros and R. Buyya, "Interconnected cloud computing environments: Challenges, taxonomy, and survey," *ACM Computing Surveys*, vol. 47, pp.1–47, may 2014. [Online]. Available: https://dl.acm.org/doi/10.1145/2593512

[8] M. Sharma, A. Gupta and J. Singh, "Resource discovery in inter-cloud environment: A Review," *International Journal of Advanced Intelligence Paradigm (IJAIP)*, In Press, 2017. DOI: 10.1504/IJAIP.2018.10023054.

[9] BD. Martino, M. Biancani, J. Aznar, D. Gallico, AJ. Ferrer, K. Djemame, ED. Nitto, G. Kecskemeti, Z. Zhao, FD. Andria, R. Woitsch, K. Kritikos, P. Deussen, P. Massonet, M. Villari, J. Costa, E. Kowalczyk, and JP. Morrison, "Inter-cloud Challenges, Expectations and Issues Cluster Position Paper," *Initial Research Roadmap and Project's Classification*, dec 2015.[Online].Available:https://eucloudclusters.files.wordpress.com/2015/05/inter-cloud-pp_dec-2015.pdf

[10] Blockchain, accessed: 2017-11-06. [Online]. Available: https://en.wikipedia.org/wiki/Blockchain.

[11] G. Tricomi, G. Merlino, A. Panarello and A. Puliafito, "Optimal Selection Techniques for Cloud Service Providers", *IEEE Access*, vol. 8, pp. 20391-203618, nov. 2020. DOI: 10.1109/ACCESS.2020.3035816

[12] L. Kapoor, S. Bawa and A. Gupta, "Intercloud: A Hype or Reality?," *International Journal of Next-Generation Computing*, vol. 9, 2018. [Online]. Available: http://perpetualinnovation.net/ojs/index.php/ijngc/article/view/466

[13] B. Lheureux and D. Plummer, "Cloud Service Brokerages: The Dawn of the Next Intermediation Age," accessed. 2010-11-08. [Online].Available:https://www.gartner.com/doc/1463714/cloud-servicesbrokerages-dawn-intermediation

[14] R. Buyya, C. Yeo, S. Venugopal, J. Broberg and I. Brandic, "Cloud computing and emerging IT platforms, vision, hype and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, pp.599–616, jun 2009. [Online].Available:https://www.sciencedirect.com/science/article/abs/pii/S0167739X08001957

[15] V. Geetha, R. Hayat and M. Thamizharasan, "A survey on needs and issues of cloud broker for cloud environment," *International Journal of Development Research*, vol. 4, pp.1035–1040, may 2014.[Online].Available:https://www.researchgate.net/publication/280323967_A_Survey_on_the_needs_and_issues_of_cloud_broker_in_cloud_environment

[16] R. Buyya,R. Ranjan and RN. Calheiros, "InterCloud: Scaling of Applications across multiple Cloud Computing Environments," *10th International Conference on Algorithms and Architectures for Parallel Processing*, jan 2010, pp. 13-31, 2010. [Online]. Available:https://www.researchgate.net/publication/285895100_Intercloud_Scaling_of_applications_across_multiple_cloud_computing_environments

[17] S. Nair, S. Porwal, T. Dimitrakos, A. Ferrer, J. Tordsson T. Sharif, C. Sheridan, M, Rajarajan and AU Khan, "Towards secure cloud bursting, brokerage and aggregation," *IEEE 8th European Conference Web Services (ECOWS)*, dec 2010. pp.189–196. [Online]. Available: https://ieeexplore.ieee.org/document/5693261/authors#authors

[18] Z. Zheng, X. Wu, Y. Zhang, M. Lyu and J. Wang, "QoS ranking prediction for cloud services," *IEEE Transactions on Parallel and Distributed Systems*, jun 2013, vol. 24, pp.1213–1222. [Online].Available:https://ieeexplore.ieee.org/document/6320550

[19] U. Schwiegelshohn and R. Yahyapour, "Resource Allocation and Scheduling in Metasystem," *Proceedings of 7th International Conference on High-Performance Computing and Networking*, apr 1999, pp. 851-860. [Online]. Available: https://dl.acm.org/doi/10.5555/645563.660323

[20] N. Bessis, S. Sotiriadis, N. Antonopoulos and R. Hill, "Meta-scheduling algorithms for managing inter-cloud interoperability," *International Journal of High Performance Computing and Networking*, vol. 7, pp. 156–172, sept 2013. [Online].Available:https://www.researchgate.net/publication/262402470_Meta-scheduling_algorithms_for_managing_inter-cloud_interoperability

[21] A. Gupta, L. Kapoor and M. Wattal, "Cloud-to-Cloud (C2C): An Ecosystem of Cloud Service Providers for Dynamic Resource Provisioning," *Advances in Computing and Communications*, 2011, pp.501-510. [Online]. Available: https://link.springer.com/chapter/10.1007%2F978-3-642-22709-7_49

[22] L. Kapoor, S. Bawa and A. Gupta, "Peer Clouds: A P2P-Based Resource Discovery Mechanism for the Inter-cloud," *International Journal of Next-Generation Computing*, 2015, vol. 6.[Online].Available:https://www.researchgate.net/publication/285482785_Peer_Clouds_A_P2P-Based_Resource_Discovery_Mechanism_for_the_Intercloud

[23] Distributed Hash Table, accessed: 2015-12-14. [Online]. Available: https://en.wikipedia.org/wiki/Distributed_hash_table.

[24] S. Sotiriadis, N. Bessis and P. Kuonen, "Advancing Inter-cloud Resource Discovery Based on Past Service Experiences of Transient Resource Clustering," *Third International Conference on Emerging Intelligent Data and Web Technologies (EIDWT)*, sep 2012, pp. 38-45. [Online]. Available: https://ieeexplore.ieee.org/document/6354719

[25] K. Sim, "Agent-based cloud computing," *IEEE Transactions on Service Computing*, 2012, vol. 5, pp.564–57. [Online]. Available:https://www.computer.org/csdl/journal/sc/2012/04/tsc2012040564/13rRUxYIMSg

[26] R. Nikbazm and M. Ahmadi, "Agent-based resource discovery in cloud computing using bloom filters," *4th IEEE International Conference in Computer and Knowledge Engineering (ICCKE)*, oct 2014, pp. 352–357. [Online]. Available: https://ieeexplore.ieee.org/document/6993399

[27] KM. Sim, "Cloud intelligence: agents within an InterCloud," *Published in Awareness Magazine*, 2013.DOI:10.2417/3201311.005153,2013.

[28] Ontology. [Online]. Available: https://en.wikipedia.org/wiki/Ontology_(information _science).

[29] Extended Data-management Markup Language (EDML), accessed: 2015-12-26. [Online]. Available: https://helpx.adobe.com/dreamweaver/extend/edml-files.html.

[30] A. WILLNER, R. LOUGHNANE AND T. MAGEDANZ, "FIDDLE: FEDERATED INFRASTRUCTURE DISCOVERY AND DESCRIPTION LANGUAGE," *IEEE INTERNATIONAL CONFERENCE ON CLOUD ENGINEERING*, MAR 2015, PP. 465-471. [ONLINE]. AVAILABLE: HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/7092962

[31] Z. Hong-lie, L. Xin, L. Yan-ju and L. Cheng, "Research on Cloud Resource Selection Method for the Multi-layer Ontology," *International Journal of Grid and Distributed Computing*, jan 2016, vol. 9, pp.193-200. [Online]. Available: https://www.researchgate.net/publication/300003049_Research_on_Cloud_Resource_Section_Method_for_the_Multi-layer_Ontology

[32] WC. Chung, CJ. Hsu, KC. Lai, KC. Li and YC. Chung, "Direction-aware resource discovery in large-scale distributed computing environments," *The Journal of Supercomputing*, mar 2013, vol. 66, pp. 229-248. [Online]. Available: https://link.springer.com/article/10.1007/s11227-013-0899-6

[33] M. Sharma, A. Gupta and J. Singh, "Blockchain-Based Resource Discovery for the Intercloud," International Conference on Next Generation Computing and Information Systems (ICNGCIS), dec 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8520318

[34] M. Sharma, J. Singh and A. Gupta, "Intelligent Resource Discovery in Inter-cloud using Blockchain," *IEEE (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, aug 2019, pp: 1333-1338. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9060267

[35] M. Crosby, Nachiappan, P. Pattanayak, S. Verma and V. Kalyanaraman, "BlockChain Technology," *Sutardja Center for Entrepreneurship & Technology Technical Report*, oct 2015. [Online]. Available: https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf

[36] SHA-256 Cryptographic Hash Algorithm, accessed: 2018-05-20. [Online]. Available:https://www.movable-type.co.uk/scripts/sha256.html.

[37] What Is Hashing? [Step-by-Step Guide-Under Hood Of Blockchain], accessed: 2018-03-20. [Online]. Available: https://blockgeeks.com/ guides/what-is-hashing/

[38] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[39] Proof-of-work, accessed: 2018-05-14. [Online]. Available: https://en.wikipedia.org/wiki/Proof_of_work.

[40] Proof-of-stake, accessed: 2018-05-14. [Online]. Available: https://en.wikipedia.org/wiki/Proof-of-stake.

[41] Smart Contracts, accessed: 2017-12-10. [Online]. Available: https://en.wikipedia.org/wiki/ Smart_contract

[42] S. Ratnasamy, M. Handley, R. Karp and S. Shenker, "Topologically-Aware Overlay Construction and Server Selection," *IEEE Computer and Communications Societies*, jun 2002, pp. 1190-1199. [Online]. Available: https://ieeexplore.ieee.org/document/1019369

[43] Z. Xu, C. Tang and Z. Zhang, "Building topologically aware overlays using global soft-state," *International Conference on Distributed Computing Systems*, may 2003, pp. 500-508. [Online].vailable: https://ieeexplore.ieee.org/document/1203500

[44] Weighted Moving Average, accessed: 2018-12-24. [Online]. Available: https://www. Investopedia.com/ask/ answers/071414/whats-difference-between-moving-average-and-weighted-moving-average.asp.

[45] B. Rochwerger, D. Breitgand, A. Epstein, D. Hadas, I. Loy, K. Nagin, J. Tordsson, C. Ragusa, M. Villari, S. Clayman, S. Levy, A. Maraschini, P. Massonet, H. Muoz and G. Tofetti, "Reservoir - When One Cloud Is Not Enough," *Journal of Computer*, mar 2011, vol. 44, pp. 44-51. [Online]. Available: https://ieeexplore.ieee.org/document/5719569

[46] X. Yang, B. Nasser, M. Surridge and S. Middleton, "A business-oriented Cloud federation model for real-time applications," *Future Generation Computer Systems*, oct 2012, vol. 28, pp.1158-1167. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0167739X12000386

[47] SM. Habib, S. Ries and M. Muhlhauser, "Towards a Trust Management System for Cloud Computing," *International Conference on Trust, Security and Privacy in Computing and Communications*, nov 2011. [Online]. Available: https://ieeexplore.ieee.org/document/6120922

[48] TH. Noor and QZ. Sheng, "Trust as a service: a framework for trust management in cloud environments," *International Conference on Web Information System Engineering, LNCS*, 2011, pp. 314-321. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-24434-6_27

[49] E. Kotsovinos and A. Williams, "BambooTrust: Practical scalable trust management for global public computing," *Proceedings of ACM Symposium on Applied Computing (SAC)*, apr 2006, pp. 1893-1897. [Online]. Available: https://dl.acm.org/doi/10.1145/1141277.1141723

[50] J. Abawajy, "Determining service trustworthiness in inter-cloud computing environments", 1*0th International Symposium on Pervasive Systems, Algorithms, and Networks*, dec 2009, pp. 784-788. [Online]. Available: https://ieeexplore.ieee.org/document/5381739

[51] ZF. Filali and B. Yagoubi, "Global trust: a trust model for cloud service selection," *International Journal of Computer Network and Information Security*, apr 2015, vol. 5, pp. 41-50. [Online]. Available: http://j.mecs-press.net/ijcnis/ijcnis-v7-n5/IJCNIS-V7-N5-6.pdf

[52] Consensus Algorithms, accessed: 2018-12-15. [Online]. Available: https://medium.com/coinbundle/ consensus-algorithms-dfa4f355259d

[53] Analytic Hierarchy Process, accessed: 2017-12-15. [Online]. Available:https://en.wikipedia.org/wiki/ Analytic_hierarchy_process.

[54] Globally Unique Identifier, accessed: 2017-07-15. [Online]. Available: https://searchwindowsserver.techtarget.com/definition/GUID-global-unique-identifier

[55] R. Winter, T. Zahn and J. Schiller, "Random Landmarking in Mobile, Topology-Aware Peer-To-Peer Networks," *IEEE International Conference on Future Trends in Distributed Computing Systems (FTDCS)*, may 2004, pp. 319-324. [Online]. Available: https://ieeexplore.ieee.org/document/1316633

[56] Bootstrapping node, accessed: 2017-07-15. [Online]. Available: https://en.wikipedia.org/wiki/Boot strapping_node

[57] Digital signatures and certificates, accessed: 2017-07-20 [Online]. Available: https://support. office.com/en-ie/article/digital-signatures-and-certificates-8186cd15-e7ac-4a16-8597-22bd163e8e96

[58] Y. Aoki, K. OtsuKi, T. Kaneko, R. Banno and K. Shudo, "SimBlock: A Blockchain Network Simulator," IEEE Conference on Computer Communication Workshops, may 2019.[Online].Available:https://ieeexplore.ieee.org/document/8845253

[59] bitcore, accessed: 2018-06-15. [Online]. Available: https://bitcore.io/

[60] TIERION, accessed: 2018-06-15. [Online]. Available: https://tierion.com/