3. GOALS AND EXPERIMENTATION METHODOLOGY

3.1 INTRODUCTION

“Web Service based Systems promise to bring significant business value out of existing IT assets through increased operational efficiencies, optimized business processes, and the ability to adapt and change quickly,” (Utah CIO Stephen Fletcher 2008). Platform and language independent and flexible access to information is indeed a complex task and resource intensive too. Web Service based Systems simplify this through standard protocols which treat all platforms equally. Through such systems, it is possible to offer data Services to a wide variety of business partners and the requests can originate from anywhere. The terms Enterprise Architecture (EA), Services Oriented Enterprise (SOE), Service-Oriented Architecture (SOA) and Service Oriented Computing (SOC) are being exposed in connection with the Web Service based Systems to an ever wider and more influential audience.

Web Service based Systems promise to be a significant innovation that will provide the ability to pick and choose business and technology Services, and will allow the trade out of Services based on organizational re-design, new strategic intent, legislative requirements, or business process modifications (USA National Association of State Chief Information Officers (NASCIO) 2011).

From its inception Web Service based Systems have been a lightning rod for dissention among enterprise architects, solution architects and application architects. Enterprise architects view such systems as a business initiative that should be guiding what information technology assets receive investment and how they relate to the business’ goals and mission. Solution architects view such systems as a means to
deliver solutions faster using the tenets of loose-coupling and finer-grained Services, which enable faster construction. Finally, application architects see Web Service based Systems as an infrastructure on which to deliver applications based on Service interfaces (JP Morgenthal 2011).

Having seen the wider adaptability and acceptance of Web Service based Systems by enterprises, they offer the following advantages over traditional approaches to distributed computing (Jagadish Chatarji 2004).

- Web Service based Systems offer business Services across the platforms and thereby the application architects are kept convenient to choose a particular Service for any kind of platforms on which the system need to be developed.

- Location independence is considered to be a major advantage through which the application developer can make use of the Web Services transported from anywhere to anywhere.

- Services need not be at a particular system or particular network is a greatest benefit of using Web Services for application development. Such benefit really brings out the developer from the pressure of locating all necessary Services from a single location and use.

- Web Service Systems are built on a completely loosely coupled approach which brings in high level of modularity and the developer need not re-
architect the entire application for a minor change in a particular section of the application.

- Web Service based Systems also offer the advantage of Authentication and Authorization support at every level of the application building which makes the entire system highly secured.

- The search and connectivity to other Services is dynamic which makes the system developer to find the Services which are just fit for the applications being developed and the connectivity extends the usage of a particular Service by getting it coupled with another Service.

**Short-term benefits of implementation:** Web Service based Systems have been the methodology by the application developers for its short benefits like Enhanced Reliability, Less hardware acquisition costs, Leveraging the existing development skills, Accelerated movement to standards-based server and application consolidation and the availability of a data bridge between incompatible technologies.

**Long-term benefits of implementation:** The Web Service based Systems also offer the Ability to build composite applications, Self-healing infrastructure that reduces management costs, Truly real-time decision-making applications and Possibility of the compilation of a unified taxonomy of information across an enterprise and its customer and partners as long term benefits of implementation.

**Benefits from the perspective of Business Value:** From the perspective of the Business Value, the Web Service based Systems provide the benefits of the Ability to
more quickly meet customer demands, Lower costs associated with the acquisition and maintenance of technology, Management of business functionality closer to the business units, Leveraging the existing investments in technology, and Reduced reliance on expensive custom development.

The field of Web Service based Systems and Service Oriented Architecture are the current topic of research and development in Computer Science, which is especially oriented by the needs of gaining and processing information in large-scale distributed systems, such as the Internet. It is expected that Web Services will be employed to a high extent for the realization of new kinds of application systems.

SOA is an architectural style for building Web Service based Systems that use Web Services available in a network such as the web. It promotes loose coupling between software components so that they can be reused. A Web Service is an implementation of a well-defined business functionality and such Services can then be consumed by clients in different applications or business processes (Qusay H. Mahmoud 2005). Web Services are

- Software components with well-defined interfaces and implementation-independent. An important aspect of SOA is the separation of the Service interface (the what) from its implementation (the how). Such Services are consumed by clients that are not concerned with how these Services will execute their requests.

- Self-contained piece of code which can perform predetermined tasks and loosely coupled which offer the benefit of independence.
Web Services can be dynamically discovered from the Service Providers and if need be, the Web Services can be combined and the Composite Services can be built from aggregates of other Services.

The Universal Description, Discovery, and Integration (UDDI) specification defines a way to publish and discover information about Web Services. UDDI is a public registry designed to house information about businesses and their Services in a structured way. Through UDDI, one can publish and discover information about a business and its Web Services. This data can be classified using standard taxonomies so that information can be found based on categorization. Most importantly, UDDI contains information about the technical interfaces of a business's Services (Karsten Januszewski, Microsoft Corporation).

Web Service based Systems have stretched the Universal Description, Discovery and Integration (UDDI) Web Services standard to the limit, and that it's time for a new standard (Joe McKendrick, IBM, 2007). IBM says that the UDDI standard for registries isn’t cutting it, and the time is now for a new registry standard more focused on today’s Service based Systems realities. "Our clients are telling us that they have an integration pain point," (Andrew Hately, a manager at IBM's Software Group). "We need to create a new standard and the time is now". Colin Atkinson (2009) says that so far all attempts to set up publicly accessible UDDI-based Service brokers have not been very successful.

For these reasons, UDDI has been chosen the area of elaborate study and intensive research by many researchers and so as me. UDDIs are interesting to users only if those architectures address the issues of interest to the users. After the reviews on Web Service based Systems and Service Oriented Architecture including UDDI,
the following concerns have been identified. These concerns are expected to be the merits of different UDDI technologies and are the basis and motivation for this thesis.

- The available connection information is just not enough in UDDI registries of federated organizations, but it should be able to share more about the Service requester’s connection information to make better predictions (Chen Zhou 2004).

- UDDI Server Architecture should be enhanced from its current state and it should support automated Web Services Registration, Availability and Quality Assessment (Witold Abramowicz, BIS 2009 International Workshop).

- Maintaining data integrity should be the top prioritized concern and the UDDI registries should not have any weakness in maintaining the data integrity of Services once they have been registered with the system (Colin Atkinson 2009).

- The Web Services’ information on an UDDI registry is to be updated as and when the changes are done by the Web Services Providers and so, the UDDI registries are expected to keep the information for registered Services up-to-date (Colin Atkinson 2009).

- It is highly necessary that for an UDDI registry to be successful as a resource recovery Service, it must be able to respond to requests from the Service Consumers in least possible time with fewer message passes across the network.
An UDDI registry must ensure high availability of the registered information to its users and the available information must be consistent across all the registries, which are maintaining information about same Services.

The concerns of an effective UDDI are well understood and that is the reason why the aim of this thesis is to deliver an effective model and an error free approach in the UDDI architecture and thereby to provide Services information which is consistent. Due care has been taken to see that the consistent information reaches the users in minimum response time as explained in Section 1.4.

3.2 GOALS

UDDI is a universal public directory where business can register and search Web Services. It facilitates businesses to easily and dynamically find and transact with one another through their preferred Services. For example, participating in UDDI can help a B2B e-Commerce organization to expand its Services into new markets and allow the company to accelerate toward a world-class business. UDDI facilitates an open framework for sharing information globally which accelerates the growth of Web Services based Systems. UDDI is major advancement in the field of e-commerce that will benefit the business organizations by providing a platform independent framework for sharing of resources. UDDI registry is a publicly accessible specification that allows organizations to extend their business across worldwide.

The primary purpose of UDDI registry is the representation of data and metadata information about Web Services provided by Service Providers. For registry
operation, UDDI uses World Wide Web Consortium (W3C) and Internet Engineering Task Force (IETF) Internet standards such as XML, HTTP, and DNS protocols.

The UDDI registry can be described as a Framework for describing and discovering business Services, and Service Providers. It defines data structures and APIs for publishing Services descriptions to the registry and querying the registry. Determining the security and transport protocols supported by a given Web Service is facilitated by UDDI registry. It contributes to the expanded offerings and thereby the extended e-commerce market.

The core component of the UDDI registry is the UDDI business registration. The UDDI structure improvises the online markets by providing them with standardized formats for Service discovery. UDDI specification describes a platform-independent registry and programmatic interfaces for publishing, retrieving, and managing information about Services provided by the Service Providers. UDDI specification includes Internet standards such as XML, HTTP, and DNS protocols. UDDI also uses WSDL to describe interfaces to Web Services.

The UDDI specification describes a hierarchical relationship between a single instance of a UDDI node and other instances of UDDI node to which it is related. Technically, UDDI registry can be classified into three major categories which are as given below.

- An UDDI node is a server that supports a least a set of functionality defined in the UDDI specification. It is a member of only one UDDI registry.
An UDDI registry is a conceptual composition of one or more UDDI nodes and it performs all the functionalities which are defined in the UDDI specification.

There are affiliated registries which are separate UDDI registries that are implemented based on policy for sharing information among them.

The UDDI has an Application Programming Interface (API) which includes the interfaces for Web Services and provides access to Services information stored in the UDDI registry. The capabilities of APIs are as discussed below.

- **UDDI Publishers API**: This API is used to publish the Web Services information in the UDDI registry.

- **UDDI Inquiry API**: A Web Service Requester makes use of this API to Search for Web Services information from the UDDI registry.

- **UDDI Security API**: Security of the Web Services information stored in the UDDI registry and the user authorization are provided through this API.

- **UDDI Custody and Ownership Transfer API**: Transferring ownership of data about a particular Web Service from one Provider to an another Provider is done through this API.

- **UDDI Subscription API**: The Web Service Providers subscription and registration to the UDDI registry is done through this.

- **UDDI Replication API**: The Web Services information replication and replication management is facilitated through this interface.
In order to address the Service discovery challenges, the big SOA vendors (Microsoft, Oracle, IBM etc) created a standard with the purpose of modeling Service metadata information that could be used to enable Service discovery capabilities. The standard was baptized as Universal Data Discovery and Integration (UDDI) and, unfortunately, it became the cornerstone of SOA governance products (Jesus Rodriguez 2007). Jesus Rodriguez also says that the UDDI has proven to be an incredibly ineffective mechanism to enable Service publishing and discovery inspite of many efforts of researchers on the subject.

The objective of this research is to establish a new methodology in the way how Web Services information are registered and inquired in an UDDI registry. This research also intends to address the core issue, the maintenance of Service Information Consistency, across all nodes of the Distributed UDDI environment, wherever the replicas of same Service have been placed. The goals of this work have been derived on the basis of improving the overall response time of the UDDI registry both for Service Inquiry and Service Publish in a Distributed UDDI environment. The goals are much narrowed and more measurable as follows.

1. To Maintain and Manage the Replica Consistency across the nodes of the Distributed UDDI environment.

2. Reduce the overall response time for Service Inquiry, Registration and Updation in a Distributed UDDI environment.

The first goal area is operation and performance oriented and the second one describes the efficiency of the proposed approach. In seeking to argue for these goals, it is clear that this work differs in flavor from the majority of the research works
carried out so far. It presents no new theorems, but has experimental results, and describes a novel approach with quantitative outputs. Note that this work compares and analyses different implementations of Distributed UDDI environments in order to identify their distinct advantages and disadvantages. The aim of this analysis is to provide the quantitative justification theory of precisely why an Optimized DST based P2P networked Distributed UDDI nodes are well suited for Web Services Registration and Querying systems. Admittedly, the goals stated here are quite ambitious. In no way it is implied that this study can definitively answer all of these questions for all environments. It is intended, however, that the statistically significant analysis presented lends insights into their answers and into the merit and appropriateness of the said system in the field of precisely the Distributed UDDI and broadly in the field of Web Service based Systems.

This approach is a first step towards proposing an effective methodology for Web Services Registration and Discovery, which incorporates Optimization and possibly a new method of having DSTs in P2P networked nodes for Distributed UDDI. The following sections describe the empirical study undertaken to pursue these goals and questions, including the selection of simulators, experimental design, and the overall operation of the study.

3.3 EXPERIMENTATION METHODOLOGY

This section first discusses the proposed Distributed UDDI Architecture wherein the proposed architecture and the connection between its constitutes are discussed one after other. After the detailed discussion on the proposed DUDDI architecture, various Replica Management related issues like Replica Placement Strategies and Replica Consistency is discussed before presenting the Overall
Experimentation Framework. Each layer of the proposed experimental framework is thoroughly presented to set the base for the further discussions on the experimentation and performance criteria evaluation, both theoretical and experimental.

### 3.3.1 Distributed UDDI Architecture

The Distributed UDDI architecture proposed in this research consists of three components, which are Service Provider, Service Consumer and Distributed UDDI System. Figure 3.1 illustrates the architecture of DUDDI with its components and the interaction among its components. Each component of the proposed architecture is explained as given below.

![Figure 3.1 Distributed UDDI Architecture](image-url)
3.3.1.1 Service Provider

In general, the Service Provider is considered to be the owner of a Service. From the application perspective, the Service Provider is a Service, a network-addressable entity that accepts and executes requests from consumers. It can be a mainframe system, a component, or some other type of software system that provide the various Services that a consumer requests. In the proposed DUDDI architecture, Service Providers are the collection of nodes that stockpile the Service code/component which they provide to its intended consumers. Service Provider maintains the catalog of various Services, segregated based on different versioning, it provides. To publish a Service, the provider engenders a Web Service Description (WSD) detail about the Service and sends it as a SOAP message to its nearest Service Replicator and Locator (SRL). As a response, the provider will receive the unique Service Identification Number (Sid) for the corresponding Service it intends to publish. The assigned Sid will be the reference for all further operations, later on, on the Service like updation, versioning or discovery.

3.3.1.2 Service Consumer

The Service Consumer, in general, is considered the user of a Service. From the application perspective, the Service Consumer is an application / Service, or some other type of software module that requires a Service. It is the entity that initiates the locating of the Service in the registry, binding to the Service over a transport, and executing the Service function. In this architecture, the consumers are the collection of nodes which, on demand, request its nearest SRL using the SOAP message which constitute the Service requirements details of the consumer. As a response, the consumer will be getting a reply SOAP message which contains the ranked Services
that are appropriate (or) matched exactly with its Service requirement specifications. The received SOAP message also contains various details such as the address of the provider corresponding to the Service, Service identification number, Service versioning, Service updation, binding and transport information. Based on the detailed received from the SRL, the consumer contacts the Service Provider which suits to its requirement and other information gathered from the response message.

3.3.1.3 DUDDI system

The DUDDI system acts as an intermediate between the Service Provider and the Service Consumer to publish and discover the Service. The DUDDI system consists of collection of dedicated nodes called Service Replicator and Locators (SRLs) to achieve the task accomplished by the traditional UDDI component in a distributed manner. The critical functions of SRL are,

- Generating Search Ontology (keywords) for any submitted Service.
- Assigning the Unique Identification Number for the published Service.
- Segregating each Service based on Domain, Ontology and other quality attributes.
- Maintaining the Service details and its corresponding provider information.
- Replicating the information such as Service Search Ontology, Service ID, Service Version and Updation history, Service and its corresponding Provider details on all other or some of the SRLs.
- Performing Replica Management activities such as placement of Service information on different SRLs, maintaining consistency of the Service information across different SRLs, providing security to the Service access, transparency of replica.
- Fast retrieval of consistent and suitable Services which matches user Service requirements and requests, across different SRLs and also perform ranking of Services for better Service selection by the consumer.

There are two lists maintained at each SRL namely, DUDDI list and Service Locator list. DUDDI list is for performing Service replica management Services such as Placement of Replica, Replica Consistency and Service Retrieval Consistency. Service Locator list is for performing activities like Publish, Discovery, Ranking and Selection which is further elaborated as follows.

Every SRL node maintains two lists; one, the DUDDI List, is to store the DUDDI node identity of all the other SRL nodes where a replica of each of its own Services published and the other list, the Service Locater List, is to maintain identities of those replicas which are placed by all the other SRLs and their identities. It is not necessary that all Services’ information of a particular SRL have to be replicated in all the other SRLs and at the same time each Service’s information should be replicated at least half of the nodes of the total SRL nodes to assure Service availability and reliability. This list facilitates to keep track of the identity of SRL nodes, which plays vital role to avoid inconsistency on Service updation, versioning and on discovery of the Service.

The Service Locator List of a SRL node maintains the essential information about the Services that are actually published by the other SRL nodes. It holds the information such as Service Provider addressing, binding and transport details, Service identification number, search ontology (keywords), Service segregation based on domain, ontology and platform, Service version and updation history. Based on the information stored in this list, Service operations such as Service updation, Service
versioning and Service functionalities such as Service publish and discovery is performed. With respect to a single Service, a SRL can be classified as ‘Home SRL’ where the Service is actually published and ‘Foreign SRL’ where the replicas of the corresponding Service is placed.

The research performed in this thesis is to illustrate the consistency of Service information maintained across different SRL nodes on distribution and updation, so that same set of Services are retrieved for same query on any SRL. To perform effective Service replica management in the DUDDI architecture, a communication structure between the SRL nodes has been established. This is also to reduce the number message passes required across different SRL nodes for Service replica management operations.

3.3.2 Replica Management Using the Proposed DUDDI

Service Replica Management is the core issue to be addressed in this thesis, as aimed. Providing the consistent Services’ information to the requested Consumers should be one of the prime objectives of a UDDI registry. Various challenges that are concerned with Service Replica Management in the proposed DUDDI architecture are discussed in detail below.

3.3.2.1 Replica Placement

Service replica placement implies to the strategy of placing replicas of the Service information on the selected SRL nodes so that it will assure the availability of the replica and reduce the response time for Service discovery. Various challenging
factors that are considered while placing the replica on SRL nodes are discussed below.

**Methodology Based:** In most of the cases, some specific methodologies have been followed for replica placement such as placing replica in all available SRL nodes, random placement and another efficient $N/2$ placement and $(N/2)+1$ retrieval method. First two methods are straight forward whereas the third one is placing the replica in at least at $N/2$ SRL nodes and retrieving the Services at least from $(N/2)+1$ number of SRLs where ‘$N$’ is the total number of SRL nodes in the entire DUDDI structure.

**Workload Based:** In this technique, workload on the DUDDI is considered to be a key measure to choose the replica placement. The workload on a DUDDI depends on the total number of Service requests received, accumulation of Service transaction messages and the number of SRL nodes to Service the consumer. As and when the workload on a DUDDI increases beyond a threshold, the system senses it and take necessary steps to tackle the crisis by activating a mirror SRL or by transferring the most requested Service replica to the nearby SRL node.

**Proximity:** Service replicas are chosen to be placed in any SRL nodes based on the hop distance of those SRL nodes. When two nearby SRL nodes, with least hop count, holds the same replica, one of them would be effectively utilized whereas the other would be underutilized or no utilization at all. Therefore, the replica placement is done in such a way that the SRL nodes which are going to hold the replicas are separated from each other a predefined number of hops and that ensures an effective utilization of all SRL nodes having the replicas of a same Service’s information.
*Consumer to Replica Latency:* This is based on the consumer’s interest, in which, when there are frequent requests for one or more Services by a consumer in any SRL node, and if the requested replica is not available there, then a replica of those frequently requested Services are transferred from the nearby SRL node to the SRL node to which the frequent requests are being made.

### 3.3.2.2 Replica Consistency

*Tracking Replica Placement:* Each SRL node tracks and manages the identity of all those SRL nodes, where the Service replicas of their own Services are placed. This tracking mechanism facilitates the maintenance of the Service updation and Service replica consistency.

*Service Version Consistency:* Every time when the Service Provider changes the Services’ information, its version identity is also updated. Both versions of the Service are available for the consumers with different version identity and this enables the consumer to still make use of the earlier versions of the Services too. Each web Service may have any number of versions but each with different characteristics and identity. Same characteristics with different version identities would increase the overhead.

*Service Updation Consistency:* Service updation consistency is to maintain the Service information invariant across all SRL nodes where the replica of the corresponding Service being maintained. Once the provider updates the Service information in the Home SRL node, the updated information is propagated to all the Foreign SRL nodes to ensure replica consistency. A systematic approach is required to
achieve the updation consistency because the erroneous and inconsistent updation would leave the whole system worthless.

**Service Retrieval Consistency:** This is one of the salient features of the proposed DUDDI system which also refers to the system’s response for every Service requests. Same set of Web Services for similar queries is guaranteed; no matter which SRL receives the request. Also the number of Services retrieved and the retrieval sequence will also be the same for the same queries.

### 3.3.3 Overall Experimentation Framework

The layered view of the experimentation framework is illustrated in the Figure 3.2. It consists of five different layers; Base Layer, Functionality Layer, Performance Attributes Layer, Operations Layer and the Assessment Criteria Layer. The responsibilities of each layer are defined and described in the following sections.

#### 3.3.3.1 Base Layer

The Base layer of the framework is the eventual propose of the work, Distributed UDDI (DUDDI) and the architecture has been discussed in the sections above. The current stage of Web Services Publish and Discovery model is largely regulated based on Centralized UDDI registries (Wenly Dong 2007). Although centralized registries can provide effective methods for Web Services discovery, they suffer from problems associated with centralized systems such as single point of failure, and bottlenecks, as discussed in Chapter 1 and Chapter 2. In addition, other issues relating to the scalability of data replication and handling versioning of Services
from the same provider have driven researchers to find other alternatives (Eyhab Al Masri 2007).

The main advantages of P2P systems are their very high robustness and scalability, due to inherent decentralization and the ability to utilize large amounts of resources available on peers connected to the Distributed UDDI network (Zakaria Maamar 2007). The P2P-based Distributed UDDI framework makes the decentralized system more scalable than traditional Web Service centralized UDDI systems by way
of distributing the system function among few peer UDDI nodes and it does not focus on only one UDDI server (Fu-zhen SUN, 2010).

3.3.3.2 Functionality Layer

This layer is responsible for addressing various functionalities that the DUDDI concerned with. The main functionalities of DUDDI are Composition, Selection, Ranking, Discovery and Publication. Web Service publishing means registering a Web Service in the UDDI registry and making it available to the Service Consumers. This will also guarantees the transfer of Web Services description to the consumer, which will be useful for the consumer to learn the way how to interact with that Web Service. Web Service composition refers to a process of adaptively compose a set of available Web Services into a business process flow, according to predefined business requirements. Web Service Discovery deals with finding a set of Services that corresponds to a predetermined user request while Web Service selection deals with choosing a Service from a set of discovered Services. Web Service ranking is the process of assigning rank to the discovered Web Services based on user requirements to enable easy Service selection.

3.3.3.3 Performance Attributes Layer

This layer possesses the core characteristics of this research, whose components are various performance attributes of the replicas, which are managed in the DUDDI system, achieved through proposed research. The attributes concerned with this layer are Security, Transparency, Reliability, Latency, Consistency, Placement, and Scalability and the proposed research considers the last four. Replication transparency enables the Service Consumers to access the Service information replica from a Home
SRL or Foreign SRL but unaware of the source of that retrieval. Service Latency refers to the actual Communication Delay (or) time delay between the Service request and the Service response experienced by the Service Consumer.

Service Replica Reliability is the probability of that particular replicated Service information being continuously accessible over a given period of time. Service Consistency refers to the Service information stored in each SRL being similar. In other words, if the information at the Home SRL node is updated, that change should be propagated to all replicas before allowing any consumer access operation. Replica Placement refers to the strategy of placing replicas on selected SRL Nodes in such way that the overall consumer-to-replica latency is reduced to the minimal possible limit. Service Scalability is with respect to the size of the system in terms of the number of Services that the system can accommodate and the volume and frequency of queries that it can take up. All these attributes have been identified but this research work focuses only on select performance attributes, due to the time constraint, which has maximum effect on DUDDI model, such as Latency, Consistency, Placement, and Scalability. The rest of the attributes are left as a future scope for the researchers to continue upon.

### 3.3.3.4 Operations Layer

This layer is responsible for different types of operations performed in the system. Distribution and Updation are the two major operations identified for research on this layer. Distribution is an operation by which the replicas of the Services’ information are copied in a set of SRL nodes. The distribution of the replicas are done keeping in mind the issues related to Latency, Scalability and Availability of the Services’ information. Updation operation refers to the process of maintaining the
different versions of the same web Service which is being updated by its providers from time-to-time. Service distribution occurs only once when the Service is published, whereas updation occurs whenever the Service specification changes and versioning are preformed. Operations in this layer are performed in such a way that the system experiences an appreciable performance in terms of the performance attributes discussed in the performance attributes layer.

### 3.3.3.5 Assessment Criteria Layer

The Assessment Criteria Layer of the framework is proposed to take care of the responsibilities of ascertaining whether the proposed DUDDI framework functions and delivers the Services as envisaged. It is proposed to assess the framework on the following basis; Propagation based, Message Pass based, Message Density based and Processing based. Basically the assessments are done in the performance attributes layer to prove that the Base layer of the proposed framework is able to effectively achieve the functions proposed in the functionality layer for processes identified in the operation layer. Various methods of assessments of the proposed system for the expected and proposed qualities are discussed below.

- Propagation based assessments are done to basically assess the transmission delay in sending the request message from the Service Consumer to the DUDDI system and receiving back the reply from the DUDDI system to the consumer in terms of the number of hops in between the two.

- Message pass based assessments are performed to verify the total number of message passes required for successful web Service transaction and it is
expected to be as minimal as possible in the proposed DUDDI than that of the traditional UDDI model.

- Message density based assessments are to evaluate the performance of the proposed system in terms of the volume of messages being live, shuttled back and forth, in the system and it is expected to be bare minimum in the proposed DUDDI.

- Processing based assessments indicate the consumer’s request-response time delay. In other words, it is the time spent by the DUDDI system to retrieve Service information corresponding to the user request specification.

Based on the discussions above, on the different methods to be employed for the assessment of the proposed system, it is proposed to demonstrate the better performance of the proposed DUDDI system based on the analysis of the experimental results on the above discussed assessment criteria.

### 3.3.4 Experimentation for Replica Consistency Management

In this section, the experimentation set up of the proposed research work is discussed. The experimental results are to be analyzed and assessed based on the assessment strategies given in the Assessment Criteria Layer, discussed in the Section 3.3.3.5. The node distribution structure in the Distributed UDDI environment has been presented in the Section 3.3.4.1 and thereafter, in the experimentation procedure how the Services are published or updated and the way how it is distributed in the DUDDI is explained.
3.3.4.1 Distribution Structure of DUDDI Nodes

A Distributed Structure of the nodes in the Distributed UDDI environment, the Service Provider nodes, Service Consumer nodes and a set of SRL nodes connected in random, is shown in the Figure 3.3. To simulate the experimental setup, OMNeT++, which is an object-oriented-modular-discrete event network simulation framework, has been used. For every different assessment methodology, proposed in the Section 3.3.3.5, the system parameters like the number of nodes in total, number of SRL nodes in particular and the number of published Services are changed to various ranges to assess the system in a highly effective manner. The SRL nodes, Service Consumer nodes and the Service Provider nodes are clearly differentiated by having different colors to each in Figure 3.3. The methodology of attaching every Service Consumer node and the Service Provider node to its nearby SRL node is clearly defined in this research through algorithms, which will be discussed from Chapter 4.

Figure 3.3 Distribution Structure of DUDDI Nodes
3.3.4.2 Experimentation Procedure

A set of operations in a sequence in different phases have been performed in the experimental setup before the system is subjected to assessment. At start, in each SLR node, both lists, namely the DUDDI list and the Service Locator list are found to be empty. The experimentation is planned to be done in two different phases with discrete sub tasks in each phase. Phase 1 takes care of the Service Publication and Distribution in the DUDDI environment. Service Replication and Replica Maintenance in the distributed nodes have been explained in this Phase. Service Updation related experimentation is explained in Phase 2. How the services are updated in the Home SRL node and how the same service’s replica is updated in the foreign SRL nodes has been well explained in Phase 2.

**Phase 1: Service Publish / Distribution:** During this phase, a set of operations have been performed to publish the Services on the Home SRL node and replicate them on Foreign SRL nodes.

**Service Publish on Home SRL Node:** The Service specification information of the Service (like provider address, binding and transport, domain, platform and version) is prepared by the Service Provider and the same is published and disseminated to its nearest SRL node, which becomes the Home SRL node for that Service. The Home SRL node will now return its identity and the unique Service identification details to the Service Provider.

**Service Replication:** On successful Service publish process at the Home SRL node, the Home SRL node will now replicate the Service information and distribute it to some selected SRL nodes which is based on the selection criteria, which are based
on a given methodology or based on the workload or based on how close the SRL node to the publisher or based on the consumer to replica latency, which are as discussed earlier in the Section.3.3.2.1.

**Replica Maintenance by Foreign SRL Nodes:** The Service information is published in the Home SRL node as discussed in the section above. After publication of the Services information in the Home SRL node, select SRL nodes, which will be the foreign SRL nodes as far as this particular service is concerned, have been identified based on select methodologies to replicate the published Services information. After the Home SRL node replicates the Services’ information to the select Foreign SRL nodes, now it is the responsibility of every one of the Foreign SRL nodes to maintain and manage the replicas placed on it. Whenever there is an update in the Services information in a Home SRL, such updations are to be properly carried out in the Foreign SRL nodes also to ensure consistency.

**Phase 2: Service Updation:** During Phase 2, the operations have been performed to ensure the updations of the Services information in both the Home SRL node and the Foreign SRL nodes. Updation carried out in Home SRL node but failed to do in the Foreign SRL node will lead to inconsistency. Phase 2 operations take note of it and accordingly, after the updations in the Home SRL node, it is ensured that the replicas stored in the Foreign SRL nodes also updated accordingly.

**Updations on Home SRL Node:** Whenever there is a necessity for a change in the already published Services’ information, the corresponding Service Provider makes the necessary changes in the Services information by way of updating it and the Service update/version information along with its identity is sent to the Home SRL node of the updated Service. The Home SRL node updates its entry and proceeds
further with the operation to update the replicas which are placed in all the Foreign SRL nodes and the details of all such Foreign SRL nodes are available in the DUDDI list of the Home SRL node.

**Updations on Foreign SRL Nodes**: The Home SRL node, after completing the updation process, propagates the same to all the Foreign SRL nodes, as per the details available in its DUDDI list, of the corresponding Service. Each Foreign SRL node updates its Service replica information and acknowledges the same to the Home SRL node on success. It is ensured that the replicas placed in all Foreign SRL nodes are updated otherwise it will lead to the problem of replica inconsistency.

### 3.3.5 Performance Criteria Evaluation

The Performance of the Proposed System has been assessed both theoretically as well as experimentally, through a simulation in OMNET++ and from the results of that simulation. The former is concerned with the notional evaluation of the proposed system and the latter is concerned with the data collected from the simulation of the proposed system. A separate assessment set up has been carried out and information gathered from the simulation is used to justify the significance of the research work in terms of the said assessment criteria.

#### 3.3.5.1 Theoretical Analysis

In this section, analysis has been performed by assigning variables for the factors such as the number of message passes between the Home SRL node and the Foreign SRL node(s) which is having the latest replica, the number of message passes between one SRL node to another SRL node, for every Service request by the Service
Consumer, who is near to the former SRL node, the number of message passes between SRL node and the Service Consumer node, the number of nodes in the system and the number of Service Consumer nodes under each SRL node (consider same number of consumers for every SRL node). Based on these aspects, the total number of message passes required for the following aspects are derived theoretically.

- Read operation in Traditional P2P based Distributed UDDI (P2P DUDDI)
- Read operation in Distributed Spanning Tree P2P DUDDI (DST P2P DUDDI)
- Write operation in Traditional P2P DUDDI
- Write operation in DST P2P DUDDI

### 3.3.5.2 Experimental Analysis

This section discusses the experimental setup established for investigating the proposed research work. The entire experimental methodology has been depicted in the Figure 3.4 OMNeT++ has been used, which is an object-oriented, modular, discrete, event network simulator for simulating the experimentation environment. A P2P model network is simulated with 100 nodes (comp1, comp2... comp100) connected in random fashion. Unit propagation delay of the transmission medium has been assumed as 10 ms. The experimental simulation and analysis have been carried out in two different phases in which, the Phase 1 analyses a Traditional P2P based DUDDI with a DST structured P2P DUDDI and in Phase 2 the analysis concerned to a Traditional P2P based DUDDI with the DST structured P2P DUDDI and an ACO (Ant Colony Optimization) optimized DST P2P DUDDI. Figure 3.4 explains the experimentation with the Traditional P2PDUDDI, DST structured P2PDUDDI and ACO optimized DST P2P DUDDI for the Service Publish or Update (a write operation), Service Read (a read operation) along with details of the message passes, consistency and response time.
Figure 3.4 Experimentation Methodologies for the Proposed Research
**Phase 1: Traditional P2P DUDDI Vs DST P2P DUDDI:** This phase of analysis has been carried out in two methods one for assessing the role of Distributed Spanning Tree in the proposed P2P based DUDDI registry and the other method is to study the scalability of the proposed system with respect to the Replica Management system. The Method 1 and Method 2 have been discussed below.

**Method-1:** Method 1 has been designed such that to prove the role of DST in improving the performance of Replica Management in terms of reducing the message passes in a P2P DUDDI environment. The DST structured P2P DUDDI has been assessed based on the number of message passes. A message pass can be defined as the transfer of message from a node to one of its adjacent nodes in the system. Each message pass experiences its own delay time which is based on the propagation delay of the physical medium being used.

Since the P2P structure is diverse in nature, usually the number of message passes required to perform any task is larger and consequently the time taken is also higher. In general, the replica consistency management of Service information is not attractive because of the high volume of message passes, which overload the DUDDI system and would cause congestion. Therefore, the proposed methodology is expected to perform the consistency management with reduced number of message passes in the system and thereby the proposed methodology can be proved effective and efficient. Therefore, in this phase, the message pass based assessments are performed in order to evaluate and justify that the proposed research technique reduces the total number of message passes required to achieve the effective consistency management of the Service replica and thereby to ensure consistent Service retrieval.
**Method-2:** In Method 2, experiments have been designed such that to assess the scalability of the anticipated environment, particularly from the Replica Management perspective. The system has been simulated with the number of nodes equal to 10 times the size that of Method-1 experimental environment.

The analysis has been carried out for four different scenarios which are Sparse Traditional P2P DUDDI, Sparse DST structured P2P DUDDI, Dense Traditional P2P DUDDI and Dense DST structured P2PDUDDI. For each scenario, analysis has been carried out in terms of message density in the system at a particular time, total number of message passes, processing delay (time taken to complete task) and success ratio of Read/Write operations.

**Message Density Based Assessments:** Message density in the system refers to the total number of messages stagnate in the system for various tasks to be achieved. With increase in the number of messages getting accumulated in the system, the overall performance of the system is reduced. It is identified that a tree like structured graph is required which would contribute in reducing the possibility of receiving the same message repeatedly by a node. This will in turn reduce the message density in the system, as identified through this research. Message density based assessments have really been proved beneficial in order to show that the amount of messages getting accumulated in the system is highly reduced in the proposed DUDDI technique.

**Processing Based Assessments:** Processing based assessments are to study and understand the performance of the system in terms of the consumer’s request-response delay time in the proposed system. In other words the time spent by the DUDDI system to retrieve a set of Service information for the Service requests given by the consumers is analyzed and proved to be better than the existing systems.
**Success Ratio Based Assessments:** Success ratio, the ratio between the number of read/write request made and the number of successful read/write performed in a particular interval of time. This assessment is helpful to understand the performance of the proposed system in terms of the volume of read/write, the system can perform per unit time, compared to the existing systems, and it is proved that the proposed system shows an improved performance.

**Phase 2: P2PDUDDI Vs DST P2PDUDDI Vs ACO DST P2PDUDDI:** In this phase of analysis, the proposed system has been examined in three different scenarios based on important issues of Service replica management such as consistency, availability and cost in terms of response time. For analysis, the same experimentation setup as used in the previous phase is used with different values of Service request generation. Here, additionally ACO optimization technique has been introduced. In Phase 2 of analysis in addition to the Message Density Based, Processing Based and Success Ratio Based Assessments, as was done in Phase 1, Cost Based Assessment has also been carried out to understand and appreciate the encouraging performance of the DST structured P2P DUDDI system after the application of ACO optimization. ACO optimization has been useful to find the optimal path for the consumer to reach the requested Service information in the DUDDI system. The other assessments have already been explained as part of the Phase 1 analysis except the Cost Based assessments which is given as follow.

**Cost Based Assessments:** This assessment helps studying the proposed system for its performance based on the number of nodes traversed and the number times each node is been accessed for getting the requested Service’s information. Cost effective traversal will lead to better response time, which is a critical performance measure that refers to the process of visiting each node in a DST, exactly once, in a systematic way at comparatively less number of hops. This has got its direct influence on the performance of the proposed DUDDI system. An
effective way of traversing the DUDDI nodes obviously reduces the number of hops between the Service Requester and SRL node, which intern improves the performance of the proposed system. In order to get impressed upon this performance measure, the number of intermediate nodes traversed before identifying the nearest SRL node for each Service item request sent by the Requester node is studied.

3.4 SUMMARY

As a result, an experiment was designed and carried out to examine the proposed DUDDI system for various performance attributes identified in the Assessment Criteria Layer of the proposed framework (Section 3.3.3). While many interesting and tough research questions remain open, preliminary results suggest that an UDDI system which is Distributed in the way how the Services’ information are available can be made consistent in terms of the information it provides about the Services. Also, based on the results, it seems viable that a Distributed UDDI, which is DST structured, P2P based and ACO optimized can show better performance than the existing Centralized UDDI, in terms of the overall response time for providing the requested Services’ information to the Service requesters/consumers. There are couple of challenges faced during this research. The first and the foremost was the simulation of the proposed DUDDI environment. The simulator, the OMNET++, is a network simulator but the study at which it is used is Web Service based Systems.

Initially considerable amount of time was spent to see the similarities between P2P based networking environment and a P2P based DUDDI environment, and after a thorough analysis and through literature survey, these two systems were found analogous. But later, the time spent was realized helpful and necessary because, such analysis only made us feel easy to have a simulation
environment whereby the proposed DUDDI environment was simulated and used for performance analysis. Other challenges were related to the amount of time the simulator consumed to perform every simulation, because, every time when the values for the system variables like the number of nodes, number of providers, SRL nodes, consumers etc., are changed, each time the simulator took its own time to give the final setup for the study and analysis. Finally, it is considered to be an useful effort because the proposed system has been proved to be performing extremely better in terms of the overall response time, compared to the existing centralized UDDI.