Intelligent Based Large Scale Multi Agent’s Resource Management on Shopping Service with Security

Surendran.R
Research Scholar, Department of Information Technology
Sathyabama University
Chennai, Tamil nadu, India
surendran.mtech.it@gmail.com

B.Parvatha Varthini
Department of Computer Applications
St. Joseph's College of Engineering
Chennai, Tamil nadu, India

Abstract — Grid computing is an Internet based large scale computing network. It works with resources and services for the purpose sharing in the network from their own or lease based. Power of Grid computing is everywhere present over the world for computing the solution for the problem through Resource management technique (RMT). RMT consist of Resource requester, resource matchmaker for a task, Resource provider and Agent-based resource manager. Agent-based resource manager produce enhanced Resource discovery service. Agent is software, decision maker for grid service. Grid service is requiring when, where, what type, alternate service arrangement and how to discover the services for public/private grid network. Multiple Agents are available in between the grid supplier and client. Multiple Agents communicated together through message passing via grid network. Large scale agent in the sense number of agents is large with big data handling capability. In each grid site active optimized agents are presented for execute the job. Each agent has a unique ID and different role in grid computing network. In some situation agents are used to migrate the one supplier to another supplier easily that is work replacement carried out by multi agent. Overloaded agent’s host is relocating the service to other agent’s host named as agent migration. Little work and idle agents are waiting for upcoming job. For deploying the application via host, first predict the best agent from a lot of distributed active agents. Online shopping application process is done through multiple Agent based Grid technique. Here, application is replaced by service. Today world every application change as grid based service. Online shopping contains product purchase and shipment service, but in offline shopping contains only product availability and details of the product known via worldwide level. Let us shopping client doesn’t know the information about where the product developed. Based on grid technique grid user without care about where the product is manufactured only receives the product at scheduled time. Same like electricity power, water supply, telephone communication, television Chanel signals and Gas distribution.

Index Terms — Agent’s Resource Management Technique; Grid service deployment Life Cycle; Service Oriented Computing; Agent based grid service for shopping service; Integrated Agent Migration based Hash Search.

I. INTRODUCTION

Grid computing is a high energy to save the power to solve the complex computation problems. This high energy production depends on Resource Management Technique (RMT). Power of Grid computing is spread over the world for computing the solution for the problem through Resource management technique. RMT features are elastic resource sharing, transparent resource discovery, high level of resource aggregation, Best resource selection, energetic resource monitoring, perfect resource allocation for a given task and well trust for resource. List of RMT components are Resource asker, matchmaker, Resource donor and Agent-based resource manager. Agent-based resource manager produce enhanced Resource discovery. [1] Large scale agent means number of agents is large. For deploying the application via host, first predict the best agent from a lot of distributed active agents. Online shopping application process is done through multiple Agent based Grid technique. Here, application is replaced by service. Today world every application change as grid based service. Online shopping contains product purchase and shipment service, but in offline shopping contains only product availability and details of the product known via worldwide level. Let us shopping client doesn’t know the information about where the product developed. Based on grid technique grid user without care about where the product is manufactured only receives the product at scheduled time. Same like electricity power, water supply, telephone communication, television Chanel signals and Gas distribution.

Figure 1, Supplier Consumer Chain or shopping process mainly depends on large scale multiple numbers of agents. First, Consumer or Grid end user asks a request service to
Intelligent Large Scale Multi Agent. Actually the Intelligent based Large Scale Multi Agent is resolving the consumer’s service request. Agents send the consumer’s request to service provider. Service Provider gives suitable response to the requesting agent. Again the agents perfectly work to send the service response to the service requester or consumer. Here, advantage is everything takes care by the agents. So difficulty of service handle is converting to easy of service. It has a two ways online and offline. Shopping processes all operations are come under one umbrella name it as shopping grid service. This Grid service is capable to discover worldwide all companies and their branches all product details. Example: In Offline shopping, find a product X details from Companies A, B, C and their company’s branches. Product details are availabilities of the product, product pricing, quality of product, contact details, and product payment way. Online shopping process use to find the product details, purchase the product and shipment of the product Y from the grid service portal. For this application collect all the content of all web sites (composite web service) get into a semantic grid service. Grid agent trust based on different security techniques like firewall, username & password, certification, biometric, mobile code verification, card/tag system, proxy credential validation encryption and decryption based.

II. AGENT BASED GRID SERVICE DEPLOYMENT

A. Intelligent large scale multi agents and their role

Tables 1, Agents are used to migrate from one supplier to another supplier easily based on grid balancing technique. [10] Overloaded agent’s host is relocating the service to other agent’s host called as agent migration. [9] Agents are dynamic, flexible, high securing, optimizing, robust and fault tolerance one. List of Intelligent large scale multi agents are, deployment agent, job or task agent, mobility agent, optimization agent, matchmaker agent, mobile agent, supplier agent, resource agent, application agent

<table>
<thead>
<tr>
<th>S.No</th>
<th>Name of the agent</th>
<th>Nature of work</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Deployment agent</td>
<td>Host the request to grid portal</td>
</tr>
<tr>
<td>2</td>
<td>Job or task agent</td>
<td>Receive a job from service requester</td>
</tr>
<tr>
<td>3</td>
<td>Mobility agent</td>
<td>Middleware for grid network</td>
</tr>
<tr>
<td>4</td>
<td>Optimization agent</td>
<td>Scheduling process</td>
</tr>
<tr>
<td>5</td>
<td>Matchmaker agent</td>
<td>Match the suitable resource for task</td>
</tr>
<tr>
<td>6</td>
<td>Mobile agent</td>
<td>Grid balancing based migration</td>
</tr>
<tr>
<td>7</td>
<td>Supplier agent</td>
<td>Resource providers</td>
</tr>
<tr>
<td>8</td>
<td>Resource agent</td>
<td>Computing device</td>
</tr>
<tr>
<td>9</td>
<td>Application or service agent</td>
<td>Shopping service</td>
</tr>
</tbody>
</table>

B. Agent based Grid Service Deployment Life Cycle

In Figure 2, to discuss the intelligent large scale multi agent based grid service deployment lifecycle. In this lifecycle consumer or grid user submit the job or product request to the Deployment agent grid portal with my proxy credentials. Consumer is create a own proxy for deploying purpose from the proxy server. After validation, only the authorized consumers are possible to access the deployment agent. [2] Deployment agents are hands over the task to the task agent for preprocess purpose. Task agent evaluates and classifies the task based on Evaluation Cum Classification (EC2) Algorithm.

Mobility agent receives a preprocessed task from task agent. These stages select the suitable grid middleware for a task process. Optimization agent get a task from middleware, then schedule the task for computation based on efficient scheduling algorithm. [4] Matchmaker agent matches the suitable resource for a task based on matchmaking algorithm. [11] Mobile agent balance the grid based on load balancing and balancing of the many Quality of Service (QoS) like balance the dynamic price (on demand based), balance the execution time, balance the job’s memory space, etc. Agent migration process happens from the mobile agent. Grid balancing technique is used to migrate the agent from one host platform to another host platform. Multiple numbers of supplier agents handle the many tasks at time for resource provision purpose. Many resource agents are co allocated for a multi task scheduling. Finally, application agents are use to host for shopping application process. Same like SDLC (Software Development Life Cycle) agent based shopping service name it as AGSDLC (Agent based Grid Service Deployment Life Cycle). AGSDLC is responsible for executing the entire shopping service process in the Dynamic grid network. Here resource match maker agent is the big role because replacement of resources from alternative computing resource selection technique Intelligent large scale multi agents responsible are Request handling, Authorization and authentication, Information management, Scheduling, alternate resource suggestion, grid balancing, match making with job and resource, Query processing, task Computation, transfer
III. AGENT’S RESOURCE MANAGEMENT TECHNIQUE

Intelligent large scale multi agents are use to manage the resources for task computation efficiently.

\[ E = \sum (J, ER) \]

Where, E-Execution time ; J- Job count; ER-Energetic resources count with execution power.

A. Agent based grid service for shopping service

Shopping service contain following QoS (Quality of service) parameter consideration for shopping service are dynamic cost, agent, availability, security, performance metrics, privacy, integrity, reliability, scalability, service measurement, and alternative agent arrangement. Consumer point of view QOS parameters are Dynamic Cost for the available resource, Time duration for task completion, deadline for the task completion, memory space for service request, user friendliness, network bandwidth, latency, security, and resource power. Provider point of view QOS parameters are resources list, availability status, time duration, policy, trust, dynamic cost, network bandwidth, alternative agent arrangement( in case of suitable resource unavailable), on demand in market (production is based on demand). The applications will use services by composing or putting them together. This service is responsible to supply chain management process. Electronic commerce service aggregates the all product details and discover to the grid user for shopping. In Online shopping service, discover the online catalog data from large number of web service, product category, models, prices, availability, payment gateway for online shopping, offers, configuration details, contact details, nearest showrooms, online query, registration of grid user, delivery details. Larger number of web services combines to discover the data to user based on web service composition. Example purchase a product X from the online. All brand web services composite and join together in the grid service at worldwide level.

![Diagram of Agent Based Grid Service for Shopping Service](fig3.png)

Figure 3. Architecture of Agent Based Grid Service for Shopping Service

In online/offline shopping, product Searching categories as two ways,

Way 1: New product search based on - key word from category
Way 2: New product search based on -image/video/audio/related information

Every product registered or stored in main server with hash value only.

1) Insert X into Agent Hash table

Agent Hash function represents for hash insert as

\[ AH(I) = X1 \mod P \]

P- Product size; X1- Product first letter’s ASCII key value; I-Index value of product X in hash table

Suppose product X’s first letter (X1) similar to product Y’s first letter (Y1) means Open addressing technique based any available index position fix for product Y. Example, The given value X which has to be inserted into the hash table will be given as an input to the hashing function. The function will return the address for its storage. While storing the record/element, if the address calculated is the one which already contain some data, the collision said to be occurred. In order to overcome this problem any of the collision resolution techniques will be used. One of the most widely used techniques is linear probing. The sequential search is made in the linear probing to find out the next empty location & the value gets stored there. If all the locations are occupied by some elements then overflow is said to be occurred.
ALGORITHM AGENT BASE INSERTAHT (AHT, P, X, Z)
//AHT is the dynamic Agent Hash table
//P is the number of products
// X1 is ASCII key value of product’s first letter to be insert
//X is inserting product name
// Z is Location value in Agent based Hash Table
Begin
I ← 1
Z ← X1 MOD P
While (I <= P)
Begin
IF (AHT(Z) = ’/0’) THEN
AHT(Z) ← X
Quit
Else
Z ← (Z+1) MOD P
I ← I+1
End
Write (“Table is full”)
End

ALGORITHM AGENT BASE SEARCHHT (AHT, P, X)
// AHT is the dynamic Agent Hash table
// P is the number of products
// X is the product to be search
Begin
I ← 1
Z ← X MOD N
While (I <= P)
Begin
IF (AHT(Z) = X) THEN
Write(‘X is available ‘)
Quit
Else
Z ← (Z+1) MOD P
I ← I+1
End
Write (“X is not available and try again based on Agent Migration”)
End

In the above algorithm AHT is the Agent Hash table which has been initialized with a special value say ‘/0’. The algorithm receives three values as arguments. The given value X will be given to hashing function to get the address for its storage. The search is made in the location Z of the hash table AHT. If the specified location is empty then it will be stored. The search for the next empty location will be made if the location is having some element with it.

2) Agent Hash Search

Agent Hash Search useful for shopping application effectively in grid computing environment. Agent Hash search is one of the fastest searching mechanisms used to locate the given element in the collection. This searching method uses the concept called Hashing to get the address of the location in the table to store/retrieve the element. Hashing is the process of applying the hashing function to the given value to get the address of the location for its storage or retrieval. There are so many hashing techniques which can be used or a user defined function may also be used. These functions may do some arithmetic manipulation on the given value and return the result. The result will be treated as an address for the given element. The table where the values get stored is called Hash Table. [5] The Hash table is the collection of several buckets say AHT(0), AHT(1). … Each bucket has the capacity to hold any number of values which will be decided by the designer of the table. The buckets may have the respective number of slots and each slot being large enough to hold one element. If the bucket size = 1 then each bucket can hold only one element. The value X will be given to the hashing function to get the address for the search. If the corresponding location of the hash table contains the value X means the successful search message will be displayed. The search will be made on the remaining buckets if the first search gets failed. The Unsuccessful search message will be given to the user if the element cannot be found in the remaining P-1 buckets.

In shopping service main sever store all data (product details).
[6] The load factor (LF) value of a hash table is the ratio of the number of keys (K) in the table to the size (S) of the hash table.

If the load factor value is high, the slower retrieval process of keys from the hash table. So maintain a load factor value as balanced one. [12] Load factor (LF) balancing same as optimal grid balancing in the grid network. Otherwise, buyer satisfaction is less for unbalanced load factor. [7] Open-addressed hash tables are based on 1D state vector, to implement the table as a dynamic vector and rehash all of the keys whenever the size changes. This is an agent migration based hash search.

ALGORITHM AGENT BASE INSERTAHT (AHT, P, X, Z)
// AHT is the dynamic Agent Hash table
// P is the number of products
// X is product name
// Z is Location value in Agent based Hash Table
Begin
I ← 1
Z ← X1 MOD P
While (I <= P)
Begin
IF (AHT(Z) = ’/0’) THEN
AHT(Z) ← X
Quit
Else
Z ← (Z+1) MOD P
I ← I+1
End
Write (“Table is full”)
End

ALGORITHM AGENT BASE SEARCHHT (AHT, P, X)
// AHT is the dynamic Agent Hash table
// P is the number of products
// X is product to be search
Begin
I ← 1
Z ← X MOD N
While (I <= P)
Begin
IF (AHT(Z) = X) THEN
Write(‘X is available ‘)
Quit
Else
Z ← (Z+1) MOD P
I ← I+1
End
Write (“X is not available and try again based on Agent Migration”)
End

Integrated Agent Migration based Hash Search (agents, hash value, product, consumer, supplier)

Step 1: Agents are calculating the hash value for consumer requesting product based on hash function with load factor (LF).
Step 2: Agent check the hash value from service provider’s hash table based on hash function.
Step 3: return the requested product hash value with details

Online shopping service’s search operation is complete up to Shipment process (delivery status). Offline shopping service only checks the product details and availability details (details of where to buy).

Scheduling based on normal queue, urgent case queue, advanced reservation based queue, priority queue , circular queue (only regular consumer circular manner process), deque (double ended queue use to process the queue from requester and provider)

Security problems (data breaches, leaks, service outages) are overcome by the agents.

3) Security Provision through RandomRSA Algorithm

Pseudo Random Number generation based RSA Algorithm used to provide more security to the grid user in a grid network. Grid user’s job (shopping request) is a plain text, Requested Product details are protected through RandomRSA Algorithm. In this algorithm, Grid user A is a service requester (buyer), Grid user B is Resource provider (seller) through agent. User A’s request is encrypt with key and send to the provider as a form of cipher text. Grid resource provider decrypts that job request through a key. Public key generated based on N. where N is a pseudo Random Number. Suppose hackers know this security provision implemented from the
RSA Algorithm, but cannot hack the original plaintext. Because RSA Algorithm output values are dynamic in nature for every execution.

```
public class RandomRSA {
    private final static BigInteger one = new BigInteger("1");
    private final static SecureRandom random = new SecureRandom();
    private BigInteger privateKey;
    private BigInteger publicKey;
    private BigInteger modulus; // n is modulus
    // generate an N-bit (roughly) public and private key
    RandomRSA(int N) {
        System.out.println(N);
        BigInteger p = BigInteger.probablePrime(N/2, random);
        BigInteger q = BigInteger.probablePrime(N/2, random);
        BigInteger phi = (p.subtract(one)).multiply(q.subtract(one));
        modulus = p.multiply(q);
        publicKey = new BigInteger("65537"); // common value in practice = 2^16 + 1
        privateKey = publicKey.modInverse(phi);
        System.out.println(publicKey);
        System.out.println(privateKey);
    }
    ```

```
    BigInteger encrypt(BigInteger message) {
        return message.modPow(publicKey, modulus);
    }
    BigInteger decrypt(BigInteger encrypted) {
        return encrypted.modPow(privateKey, modulus);
    }
    public String toString() {
        String s = "";
        s += "public = " + publicKey + ",n";
        s += "private = " + privateKey + ",n";
        s += "modulus = " + modulus;
        return s;
    }
    public static void main(String[] args) throws IOException {
        int N = Integer.parseInt(args[0]);
        RandomRSA key = new RandomRSA(N);
        System.out.println(key);
        // create message, encrypt and decrypt
        DataInputStream in = new DataInputStream(System.in);
        System.out.println("Enter the message ");
        BigInteger message = new BigInteger(in.readLine());
        BigInteger encrypt = key.encrypt(message);
        BigInteger decrypt = key.decrypt(encrypt);
        System.out.println("encrypted = " + encrypt);
        System.out.println("decrypted = " + decrypt);
    }
}
```

Advantages of proposed Agent based grid service

- Adaptable, smart, Secure and flexible. Client request to be a top secrete
- Improved grid balancing- Overloaded agent’s host is relocating the service to other agent’s host named as agent migration.
- Replace the difficulty of shopping & shipment
- It is used to reduce the rescheduling time and re-negotiation time
- Resource availability is increased from agent migration and fault tolerance agents.
- Richer forms of sharing, and increased qualities of service.
- High preference rate and reduced cost for consumer from intelligent agents.
- All service provider product details display in the Grid shopping service’s output screen.
- Same type of product price is unique in a world level (To avoid the price variation for same product)

IV. EXPERIMENT

Experimental setup nodes are arranged in the LAN (Local Area Network) based on graph network. Graph network consider as a grid network via LAN. Here, graph’s vertex as computing nodes based on active agents and edges as LAN connection wires. Figure 4, LAN Setup normally pseudo random based 28 times executed for finding performance of Agent based on cost. Dynamic Cost is Y axis and Agent Count is X axis. Simulation based market dynamic cost is decreased based on increased number of agents. In this experiment human work (consumer) replace by the agent based software grid service for intelligent shopping service. In LAN, Simulation based consider the one node system for service request (product request), multiple number of energetic node system for active agent, and many service provider node system with hash table based databases. From a lot of Agent system one system act as a Master or head node system for maintain a SQL based databases.

Agent Hash table have a key values, index value, table size, details of all key values. In Agent hash table, key value as products name, index value as product’s storage location, key details are product information like product’s price, manufacturing date & time, product configuration, any offers, payment gateway options, … etc. Hash table use to resent the databases like inventory db, sales db, and order db. More Features are added for shopping service like product Comparison, zoom the product image, video of the product, sort the price based on low to high or high to low. Market based Dynamic cost for product is fixed for all places. This is achieved based on increased number of intelligent large scale Multi Agents. Same type of product price is unique in a world level (price variation for same product is destroyed).
Security provision experiment is done from the Random RSA Algorithm. Here, every execution time all parameter values are changeable one like p, q, private key, cipher text values. p & q are prime numbers. Example refers the fig.4, value of N is similar for both the execution process. But all parameter values changed because of random number generator (N).

Agent based resource management process contain scheduling, resource allocation, best resource selection, identify the job status.

Figure 8, LAN Setup normally pseudo random based 28 times executed for finding performance of task computation based on agent. Task count is Y axis and Agent Count is X axis. Simulation based task count is increased that is high level of task completion (execution) and increased number of agents. Actually Increase the Multiple intelligent agent get a reduced cost and increased number of task computation.

V. CONCLUSION AND FUTURE WORK

Resource Management Technique is looking like an umbrella to cover multiple functions like resource monitoring, resource selection, resource discovery, resource matchmaking, resource Allocation and Co-Allocation, Grid service creation, maintenance, updating, and implementation based on OGSA (Open Grid Service Architecture) and OGSi (Open Grid Service Infrastructure). Both OGSA and OGSi responsible are Grid service descriptions and instances, Service states, meta data, and introspection, Naming and name resolution, Service life cycle, Fault type, and Service groups. Intelligent Large Scale Multi Agent's count value increased time the performance of the grid system increased and Dynamic costs value decreased at the same time greater number of tasks is executed. All companies (service provider) product details display in the Grid shopping service's output screen with fast and secure manner. Same type of product price is unique in the worldwide level from the agent based grid service. In future design, increase the more QOS attributes of buyer and seller in shopping grid service and also consider the more functional operation like increasing degrees of virtual organization, well defined negotiation techniques.

REFERENCES