Enabling Peer-to-Peer SDP in an Agent Environment

University of Maryland Baltimore County
USA
Introduction
Related Work
Overcoming Limitation of exiting DF
Cluster Based Architecture
Exploiting the Cluster Based Architecture
Implementation
Conclusion
Success Agents Paradigm
   • Negotiation, Co-operation, Collaboration

Main Challenge - Discovery agents offering them as services

Existing Approach
   • FIPA’s Directory Facilitator (DF) provides Yellow pages service
   • Agents register their services to DF in the Agent Platform
   • Agents in the Platform can query the DF for services and get back the results
   • Limitation – Search is limited to a particular agent platform

Why Global resource discovery ?
   • Increase in deployment of Agent Platform
   • Inter-Platform communication possible because of FIPA’s Standardization

Two Approaches for Global resource discovery
   • Centralized approach
   • Distributed approach
Centralized Approach
- Agents register their service to Central Repository (Global Directory Facilitator)
- Agents can query the Global DF for services
- Advantage: faster query response
- Disadvantage: Updating changes in service availability are costly, high maintenance is required, single point of failure.

Distributed Approach
- Federation between DF
- Each DF forwards its service discovery query to other DF it federates with
- Advantage: Less maintenance with respect to updating service changes
- Disadvantage: Poor query response as the query has to propagate through the network.
- Agents form Gnutella Type distributed networks
- Gnutella was designed to preserve anonymity in the network

Sycara et al. “A taxonomy of middle agents for the internet”
- Special agents that provide service to locate the resources in the network
- Similar to the Centralized approach

Peer-to-Peer Systems
- Napster
- Gnutella
Problem A: Dynamic Environment
- Environment is very dynamic, Agents may enter and leave the network
- Changes incurred due these agents should be reflected in the global searches

Solution:
- In a centralized model excessive traffic near the central server to reflect these changed
- Distributed model is ideal for environment because the changes need not be propagated to a central server

Problem B: Service Registration and De-registration
- Agents are responsible for service registration and de-registration
- Stale service registration information can be stored in Directory Facilitator

Solution:
- Leasing Mechanism: Agent registers its service for certain time, it also has to renew its registration after the time period
- Directory Facilitator can deregister the agent’s service registration if the agent fails to renew
Problem C :- Malicious Agents
• Directory Facilitator is not responsible for fake services being register

Solution :-
✓ Reputation Mechanism – DF should be responsible hosting reputation mechanism
✓ Each agent after utilizing the service may send feedback to the DF of the agent which provided the service
✓ DF will compute the reputation based on the feedback
✓ DF will tag the reputation with the search results

Problem D :- Service Description
• Current Service Description is not sophisticated for inference

Solution :-
✓ DAML S, initiative by semantic web community, provide core markups for describing services and their capabilities
✓ Example of simple inference – services which forms the composite services can also be registered individually
Problem E: Directory Facilitator Search Mechanism

- Current DF search mechanism is syntactic
- A global service description for each service provided is not possible
- A syntactic search would lead to poor search results

Solution:

- Directory Facilitator should implement semantic search mechanism
Assumption

• We assume Directory Server where agent platforms register themselves
• After registration Directory Server doesn’t play any role in resource discovery

Startup

• A node (agent platform) on entering network registers itself to the Directory Server and gets addresses of other nodes in the network
• Contacts the nodes in the network and gets the Master address of these nodes
• Contacts the master nodes
• Based on the results the node either joins a clusters or starts its own cluster.
Query Routing

- A node in cluster queries the cluster head
- Cluster head apart from searching its own index forwards the query to know cluster head
- Other cluster head on receiving forwarded query searches its index and forwards the query
- Cluster heads forwards the results to the cluster head that originated query forwarding
- The cluster head that originated the query then forwards the results back to node that originated the query
Cluster Maintenance

- Leader is in charge of updating cluster nodes if any node enter/leaves the network
- The reason for cluster maintenance is to help in the leader election

Leader Election

- Cluster nodes keep pinging the cluster head
- If (cluster head left the cluster) then
  - Leader election process takes place among the cluster nodes
  - New cluster head is selected and announces that it has replaced the previous cluster head
Advantage of the cluster based architecture

- Flexible moving the nodes from one cluster to another

Hierarchy

- One level to multiple level
- Top level nodes can act as Query Routers instead of holding the DF information

Cluster Formation

- Clusters can be based on network location to localize the traffic due to cluster maintenance and leader election
- Cluster node to be exchanged to/from an idea cluster
- We can merging and splitting of cluster so that the cluster head works on optimal load
Caching

- Caching of global results are not of any use in an one level architecture.
- Only the cluster head that started the query forwarding will have the results of global results of the query.
- If similar query is generated in the same cluster then the cluster head could return the cached results.
- If the query is generated in other cluster then even if the cluster head has with global cache result forwards the global results, it is not going to stop the query being flooded in the network.
- However the caching will be useful in multi-level architecture.
The system was implemented using JADE Agent Platform

Directory Facilitator has been modified to federate between the other Directory Facilitator