



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

(Open Access, Double Blind Peer-reviewed, Refereed and Indexed Journal)

www.iasir.net

JADE Agent framework for Distributed Data mining

Darshana Patel

Research Scholar

Gujarat Technological University

Chandkheda, Ahmedabad, Gujarat, INDIA

Abstract: *There are different approaches for distributed data mining, from which Agent framework has been proposed as effective paradigm for distributed data mining. This paper describes agent, its components and different agent frameworks. The comparison of different agent framework has been carried out on the basis of security, model, programming language, communication facility and tools provided for the framework. The best suitable framework has been identified to create the agent for Distributed Data Mining and the implementation has been performed on the selected framework.*

Keywords: *Mobile Agents; Distributed Data Mining; Jade; Cassandra*

I. Introduction

Distributed Data Mining (DDM) is a branch of data mining that offers a framework to mine distributed data closely monitoring the distributed data and computing resources. Distributed data mining originated from the need of mining over decentralized data sources regardless of their physical locations. Distributed data mining is required in two cases, 1) The data is distributed on different locations and mining process is performed and 2) Data mining, sometimes needs lots of resources to perform mining process and to distribute the load, distributed mining process is required. The mining process is carried out at the local sites and final results are aggregated at the global site.

For distributed applications there are different paradigms like client-server, remote evaluation, code on demand and mobile agent. [5] "From the different paradigms for distributed environment, mobile agents has many advantages like; reduced communication cost, asynchronous execution, easy development of distributed applications. Mobile agents make it easier, faster, and more effective to develop, manage, and execute distributed applications than other technologies. There are different applications of mobile agents in DDM like remote information retrieval, Network management, Cloud computing, Mobile computing, Software testing etc". [2]

The agent based distributed data mining is most popular and many applications are available in distributed data mining like JAM, BODHI, PADDMAS etc. In JAM-Java Agent for Meta learning; agent operating on local database and produce local classifier and then imported to a data site where they are combined using meta-learning agent. BODHI (Be-seizing knowledge through Distributed heterogeneous Induction) works on mobile agent based technology. In BODHI agents can be transferred from the one location to another location. The PADDMAS-parallel and Distributed Data Mining Application Suite (Rana, Walker, Li, Layden and Ward, 2000) is another component based system for developing distributed data mining application.

To implement agent in DDM different frameworks are available. This paper is further divided into different sections, agent in DDM, comparative study of DDM and JADE framework, implementation and conclusion.

II. Agents

Mobile agent technology is a paradigm that can take advantage of current distributed environments. [8] The mobile agent technology arise from two different disciplines: artificial intelligence, which defines agent concepts [4], and distributed systems, which deals with code mobility [5]. In DDM agent itself as code move to the local machine and perform the mining process and only result is transferred to the central site. An agent is a software component that provides an interface to an arbitrary system and/or functions like a human agent, working for some clients in search of its own agenda. Multi-agent systems (MAS) can model complex systems and introduce the possibility of agents having common or contradictory goals. These agents may collaborate with each other. Moreover, agents are an effective way for processing information and transfer over wireless networks, which have low bandwidth and high latency by moving an agent to a location, it can access the resource locally and reduces costly data transfers over networks. This decreases network traffic and improves data delivery, because it is faster and cheaper to send a small agent to a data source rather than sending all the intermediate data to the requesting site. Agent is Autonomous; it does not require human intervention. It co-operates humans and other agents to accomplish the task. It perceives the environment and responds to changes in environment. It also takes initiatives by its own and

has ability to travel between nodes. The agents are classified as Collaboration Agents, Collaborative learning Agents, Interface Agents and Smart Agents. [9]

III. Agent Components

A mobile agent consists of following three components: Code - the program that defines the agent's behavior. Attributes - Information that describes an agent like its origin, its owner, its movement history, resource requirements etc. State - The agent's internal variables etc., which allows it to resume its activities after moving to another host. [10] Some of this part is accessible by the agent, but agent must not be able to modify the attributes. Mobile agent initially resides on a computer called the home machine. The agent is then sent to the remote computer called mobile agent host and is executed on the remote host. Only the code of mobile agent is transferred to the remote host and the host provides a suitable execution environment for the mobile agent to execute. The mobile agent uses resources (CPU, memory, etc.) of the host to perform its task.

IV. Different Agent Frameworks

Agent architectures are the primary mechanisms underlying the independent components that supports efficient behavior in real-world, dynamic and open environments. There are various agent development platforms available for development of different kinds of applications. This paper also describes the study of a variety of agent technologies along with their advantages and dis-advantages. Different mobile agent frameworks are Aglets, Voyager, JADE, TACOMA, Grasshopper, SPRINGS, Tryllians Agent Development Kit, Zeus etc. Aglets is developed by IBM which is combination of agents and applets which provides mobility to the applet. Aglets is agent technology developed with use of java. Aglet is abstract class of java. Aglet can work as mobile agent, it traverse from one location to another location and complete the task. Aglet implements event handler operations. Aglet use single thread mechanism so two agents cannot send synchronous messages to each other at the same time otherwise it create deadlocks. Voyager is also java based agent framework developed by Object Space. With use of voyager technology distributed application is possible with traditional messaging or RMI is possible. Voyagers business software is not freely available and the agent is active for particular time period only. TACOMA Tromoso and Cornell Moving Agents is developed by Tromoso and Cornell University. In TACOMA architecture data files and code files are used. Data files store status of the agent and the code file stores actual code of the agent. Grasshopper is developed by IKV++. Grasshopper use MASIF (Mobile Agent System Interoperability Facilities) support for developing mobile agent application. Grasshopper provides GUI support to develop agent. The Grasshopper mobile agent framework is made of various regions. Developers can get dynamic proxy benefit due to regions. Springs is developed by DISG- Distributed Information Systems Group at the University of Zaragoza in Spain. It proposed a hierarchical infrastructure of regions but it does not support FIPA standard and does not provide graphical tools. JADE is developed by R & D department of telecom Italia and latter distributed as open source under LGPL license. It is java based, follow FIPA specifications. Jade support GUI. It provides run time environment for agent development and can develop multi agent system. It has inbuilt agent mobility service and agent management system. [6] The comparative analysis of different agent framework has been described here.

Table I Comparative Analysis of Mobile Agents

Features of Mobile Agent	JADE	Voyager	Aglet	Springs	Tacoma	Grasshopper
Security	Strong Traditional	Secured channel	Partial	Partial	Firewall agent	Partial
Model	Procedural	Procedural	Event	Procedural	Behavioral	Procedural
Programming Language	JAVA	JAVA	JAVA	JAVA	JAVA	-
Proxies	No	Yes	Yes	Yes	No	Yes
Synchronous Communications	No	Yes	Yes	Yes	No	Yes
Asynchronous Communications	Yes	Yes	Yes	Yes	Yes	Yes
Messages	Yes(FIPA)	No	Yes	Yes	Yes	Yes
Remote Calls	No	Yes	No	Yes	Yes	-
Call by Name	Yes	No	No	Yes	No	No
Available for Download	Yes(LGPL)	Not Free	IBM PL	Yes	-	Not Free
GUI Tools	Yes	No	Some	No	Yes	Yes
Level of Activity	Very High	High	Very Low	High	Very High	None

V. Jade Framework

JADE is Java Agent Development Framework. With the use of JADE, Agents can be created. JADE is purely developed in java language. The implementation of multi agent system is simplified through this middleware which provides set of graphical tools that support deployment phases. As java is platform independent same as that the agent environment can be changed run time in jade.[6] Several efforts have been done towards the standardization of agent technologies and one of major standards is the FIPA (Foundation for Intelligent Physical Agents). The

Foundation for Intelligent Physical Agents (FIPA) [7] is an international non-profit association of companies and organizations sharing the effort to produce specifications of generic agent technologies. FIPA is envisaged not just as a technology for one application but as generic technologies for different application areas, and not just as independent technologies but as a set of basic technologies that can be integrated by developers to make complex systems with a high degree of interoperability. However, the use of a common communication language is not enough to easily support interoperability between different agent systems. The standardization work of FIPA is in the direction to allow an easy interoperability between agent systems, because FIPA, beyond the agent communication language, specifies also the key agents necessary for the management of an agent system, the ontology necessary for the interaction between systems, and it defines also the transport level of the protocols. [8]

VI. Architecture of JADE

JADE agent work on the platform. The platform provides basic services to the agent like message delivery. Platform is combination of containers. The container can be executed on different host so distributed platform can be achieved. In the containers one of the container is main container from which platform start. The main container contains special platform agents like AMS which can manage platform. DF agent is also located on the main container which provides details about service of another agents. [11]

Figure 1 Architecture of JADE

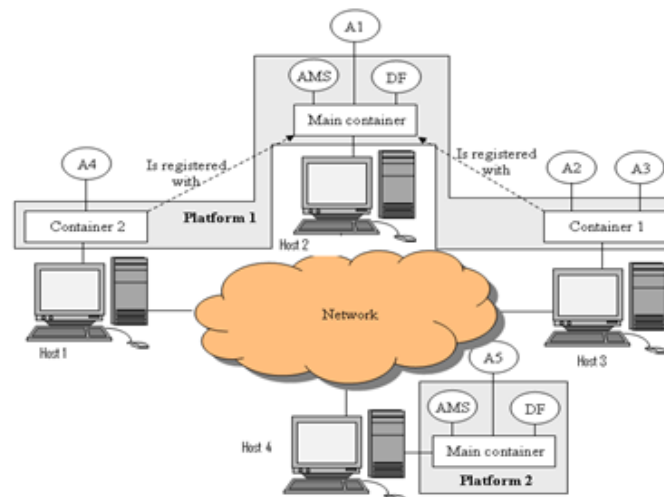


Figure 2 Booting JADE

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\jade\src>java jade.Boot -gui_
```

Figure 3 Jade Boot Completed

```
Administrator: C:\Windows\System32\cmd.exe - java jade.Boot -gui_
Aug 25, 2016 3:05:42 PM jade.imtp.leap.LEAPIMTPManager initialize
INFO: Listening for intra-platform commands on address:
- jicp://192.168.1.35:1099
Aug 25, 2016 3:05:42 PM jade.core.BaseService init
INFO: Service jade.core.management.AgentManagement initialized
Aug 25, 2016 3:05:42 PM jade.core.BaseService init
INFO: Service jade.core.messaging.Messaging initialized
Aug 25, 2016 3:05:42 PM jade.core.BaseService init
INFO: Service jade.core.resource.ResourceManagement initialized
Aug 25, 2016 3:05:42 PM jade.core.BaseService init
INFO: Service jade.core.mobility.AgentMobility initialized
Aug 25, 2016 3:05:42 PM jade.core.BaseService init
INFO: Service jade.core.event.Notification initialized
Aug 25, 2016 3:05:42 PM jade.mtp.http.HTTPServer <init>
INFO: HTTP-MTP Using XML parser com.sun.org.apache.xerces.internal.jaxp.SAXParser
impl$JAXPSAXParser
Aug 25, 2016 3:05:42 PM jade.core.messaging.MessagingService boot
INFO: MTP addresses:
http://DIKU-PC:7778/acc
Aug 25, 2016 3:05:42 PM jade.core.AgentContainerImpl joinPlatform
INFO: Agent container Main-Container@192.168.1.35 is ready.
```

Figure 4 Adding Data Mining Algorithm to Jade

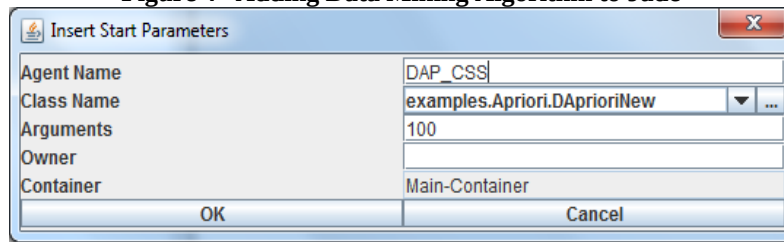


Figure 5 JADE Agent Management GUI

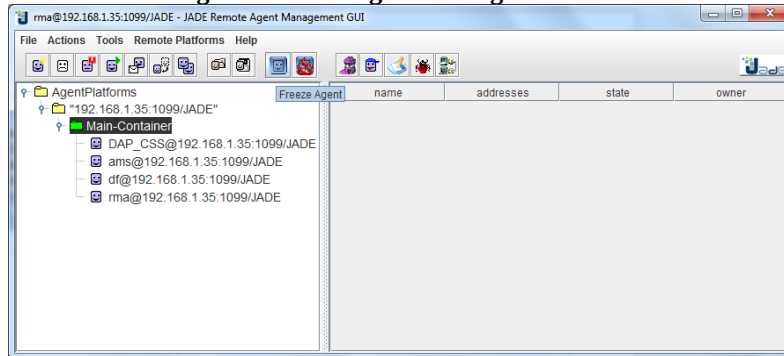
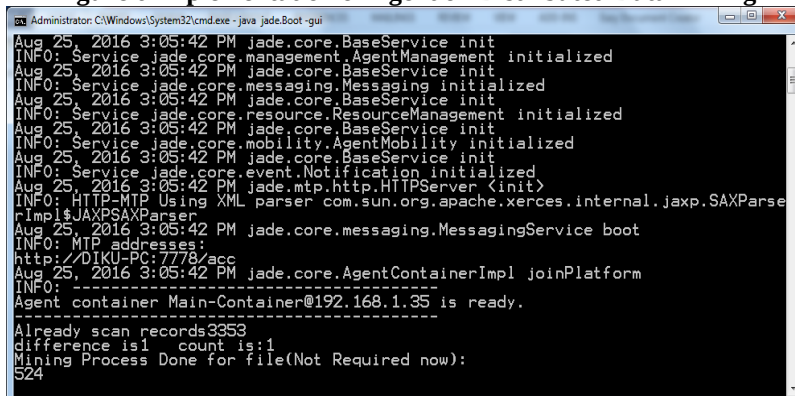


Figure 6 Implementation of Agent on Distributed Data Mining



VII. Conclusion

This paper analyzed the mobile agent frameworks and on comparison, JADE seems to be better as compared to other frameworks. Jade is open source platform, designed purely in java and supports GUI. It provides run time environment for agent development and can develop multi agent system. It has inbuilt agent mobility service, agent management system, security features and sound agent mobility. Jade is free and provides good documentation, very good GUI. Jade has been already used in many successful projects. In aglet for multi-agent synchronous message communication is not possible. In Voyager agent is active for particular time only. In Tacoma it is required for the programmer to explicitly know about the agent status before migrating it. The aglet, voyager, Tacoma does not comply with FIPA standards. So by this paper it has been identified that JADE is best suitable agent architecture for Distributed Data Mining.

References

- [1] A. Fuggetta G. P. Picco and G. Vigna, Understanding Code Mobility, IEEE Transactions on Software Engineering, Volume 24, No.5 (May 1998).
- [2] Ichiro Satoh, Mobile Agents, National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan, Retrieved March 2, 2015 from <http://research.nii.ac.jp/ichiro/papers/satoh-mobile-agent.pdf>
- [3] White James E., Telescript technology: mobile agent Mobility, ACM Press/Addison-Wesley Publishing Co.,1999
- [4] Russell, Stuart Jonathan, Artificial intelligence: a modern approach, Englewood Clis: Prentice hall, Vol. 74. , 1995
- [5] Fuggetta, Alfonso Gian Pietro Picco and Giovanni Vigna, Understanding code mobility,IEEE Transactions on Software Engineering Vol 24 No 5, 1998
- [6] jade.tilab.Com, Java Agent Development Framework., Telecom Italia (March 2016).Retrieved June 9, 2016 from <http://www.jade.tilab.com>.

- [7] fipa.org, Foundation for Intelligent Physical Agents: Specifications, Telecom Italia (March 2016), Retrieved June 9, 2016 from <http://www.fipa.org>
- [8] David Wallace Croft, Intelligent Software Agents: Definitions and Applications, Senior Intelligent Systems Engineer Special Projects Division, Information Technology, Retrieved July 14, 2015 from <http://alumnus.caltech.edu/croft/research/agent/definition>
- [9] Hyacinth S. Nwana, Software Agents: An Overview, Knowledge Engineering Review, Vol. 11, No 3, pp. 205-244, October November 1996. Retrieved June 9, 2015 from [http://www.upv.es/sma/teoria/agentes/software agents-nwana.pdf](http://www.upv.es/sma/teoria/agentes/software%20agents-nwana.pdf)
- [10] KumKum Garg, Mobile Computing : Theory and Practice, 1st. ed. Pearson Publication, South Asia.
- [11] Giovanni Caire, jade tutorial : Jade Programming for beginners, last update: 30 June 2009 , Retrieved June 9, 2015 from <http://jade.tilab.com/doc/tutorials/JADEProgramming-Tutorial-for-beginners.pdf>

Acknowledgments

Author is thankful to Dr. J. S .Shah, Gujarat Technological University and Dr. Gaurav Sharma, Rochester University for extending their support and guidance.