ISSN No. 0976-5697

Volume 2, No. 5, Sept-Oct 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

SE-MABKM: An Epoch For Organization Knowledge Management

Ripu R. Sinha*
Research Scholar,
NIMS University Rajasthan
Itct.ripu@gmail.com

Dr C.S Lamba
Research Supervisor,
HOD, RIET Jaipur, Rajasthan
Profflamba@gmail.com

***Abstract*:** It is true to say that current working environment is 'MFSWE' i.e. multi facet smart working environment. Such environment requires smart system that performs tasks intelligently and smart system can be developed by the smart framework. In order to support such environment the concept of SE-MABKM has been brought forward for effective management of organizational Knowledge Management. While as SE-MABKM is providing a conceptual framework for OKM especially for software organization. In the software organization knowledge management became parts and parcel. SE-MABKM framework will work as per the requirement and authorization of knowledge Management Communities to design and develop an effective tools or application for smart working environment for various life cycles like Management life cycle, Organizational process Improvement Life cycle, and Software development Life cycle as well as Software Test Life cycle. Management caters to the critical issues of organizational adaptation, survival, and competence in face of increasingly discontinuous environmental change. Essentially, it embodies organizational processes that seek synergistic combination of data and information processing capacity of information technologies, and the creative and innovative capacity of human beings. Reason of this paper writing is providing a framework for multi agent based knowledge management for software companies and after that this framework to be applied at various life cycle model in individual as well as collaborative manner for effective knowledge management tools and application development.

***Keywords*:** SE-MABKM, MFSWE, KM, OKM, MABKM, MAS

I. INTRODUCTION

Organizational knowledge to be manage in effective manner otherwise they will manage effectively in reverse direction of policy. Today tools, application to be developed in this manner that it is used in further. Better software development policy can make better software Application. Organizational knowledge base is Became important tools for strategic decision making. While as we are crossing the horizon with knowledge and approach so here Multi Agent System (MAS) 'Multiple Interacting Intelligent Agents' can be used to solve problems that are difficult or impossible for an Individual Agents or Monolithic System to solve. Intelligence, May Include methodic, functional, procedural or algorithmic search, find and processing Approach. The agents in a multi-agent system have several important characteristics: Autonomy: the agents are at least partially autonomous, Local views: no agent has a full global view of the system, or the system is too complex for an agent to make practical use of such knowledge Decentralization: there is no designated controlling agent (or the system is effectively reduced to a monolithic system) Typically multi-agent systems research refers to software Agents. However, the agents in a multi-agent system could equally well be robots, humans or human teams. A multi-agent system may contain combined human-agent teams. Multi-agent systems can manifest self organization as well as self-steering and other Control Paradigms and related complex behaviors even when the individual strategies of all their agents are simple. When agents can share knowledge using any agreed language, within the constraints of the system's communication protocol, the approach may lead to a common improvement. I.e languages are Knowledge Query Manipulation Language (KQML) or FIPA'S Agent Communication Language (ACL). While as MAS are self organized and knowledge management is necessary for organization development. Knowledge management is became the key to successes and this successes can be achieved via knowledge practices framework [1]; in my study of knowledge management approach for knowledge based Management framework. Both Industry and Academia are interested in knowledge management practices and organizational knowledge base development. As per the knowledge the successes of product life cycle depends on the three essential elements that is technology, people and process. So with the help of SE-MABKM Concepts (Support of Software Engineering via Multi Agent Based Knowledge Management) we are providing a knowledge based framework development based on the multi agent approach [2]. This framework knowledge practices to be applied based on the multi agent system. Within the organization each and every data to be recoded in centralized manner for further effective decision making. SE-MABKM is providing a framework where each and every data collected by the various agents to be recorded itself in the centralized ways. In the SE-MABKM, Knowledge management process and practices is performing based on Multi agent (MABKM). MABKM is providing a ways for effective organizational KM practices Knowledge communities which are suffering for better policy and practices to get better ROI but management policies are unable to produce desired result [2]. SE-MABKM is providing framework based on that in future software or application to be developed [3]. With the help of SE-MABKM process strategic organization policy management at software companies to be boost via centralized data storage with proper indexing, Bandwidth, warehousing and data mining approach. It also provide effective data ware housing ,data mining, proper utilization of bandwidth as well as indexed form of data for fast retrieval. In this research paper we have deals concepts

about MAS and its characteristics. MABKM and its operational ideology, SEO process and practice for optimum knowledge management and integration of MABKM Framework into various life cycle of SEO in individual as well as combinational that will gives then concepts of SE-MABKM and its regulatory framework and finally Effectiveness of SE-MABKM and Implementation factor of SE-MABKM.

II. AGENT AND MULTI AGENT SYSTEM

The various definitions presented in the literature identified the key properties that characterize an intelligent agent [5][6][7][8][9][10][11][12] :

Autonomy: agents operate without the direct intervention of humans or others, but have some kind of control over their actions and internal state using a set of tendencies. Tendencies are individual goals to be achieved by the agent.

Social ability: agents cooperate, negotiate, and communicate with other agents.

Reactivity: agents perceive their environment, and respond in a timely fashion to changes that occur in it in order to satisfy their design objectives.

Pro-activity: agents do not simply act in response to their environment; they are able to exhibit goal-directed behavior by taking the initiative. Agents are capable of handling complex and high-level tasks. The decision as to how such a task is best split up into smaller sub-tasks, and in which order, and way, the sub-tasks are best performed, should be made by the agent itself.

Temporal continuity: agents are continuously running processes.

Mobility: an agent has the ability to transport itself from one computer to another, retaining its current state.

Learning: agents are able to learn and adapt themselves to fit their environment.

We also identified classes of agents have been defined in the literature: reactive agents and cognitive agents [5][6][7][9][10][13].

Cognitive agents possess internal representation models of the world and expertise, have goals and plans, are capable of reasoning, and can cooperate, coordinate, negotiate, and communicate with other agents.

Reactive agents in contrast do not have any internal symbolic models of their environment, and they act using a stimulus/response type behavior by responding to the present state of the environment in which they are embedded. Work on reactive agents originates from research work carried out by [14] in robotics at MIT in 1985. In his paper, he objected to cognitive agents, and developed reactive agent architecture for the control of autonomous mobile robots. The most common modules of the internal architecture of an agent are: perception, execution, self-knowledge, acquaintance knowledge or social knowledge, domain knowledge, reasoning, learning, cooperation, and communication [5][6][10][11][12][17][16][18][19] These are described in more detail below.

Perception: the perception module is one of an agent's interfaces to its environment. Commonly the perception module obtains signals from the agent's sensors. But in most architecture this module is integrated into the communication interface.

Self-knowledge: the agent's self-knowledge contains agent's knowledge about itself, including its physical state, location and skills, etc.

Domain knowledge: this knowledge concerns the problem-solving domain and environment. Usually this module contains the description of the problems to be solved**.**

Social knowledge (acquaintance knowledge): this knowledge, also called beliefs, is the knowledge used by the agent to interact with its acquaintances. It describes the skills and identity of acquaintance agents. The knowledge is used by the agents to identify other agents with whom it is useful to interact, and wish to determine which agents have the skills necessary to perform a particular task. This knowledge must indeed model the role, competence, the localization (address of an agent), the goals, the plans, and the resources of these dealings to be able to interact with them.

Learning: an agent working in a dynamic environment needs to adapt to changes in that environment. It needs to learn in order to update its knowledge about its environment, other agents, and the problems to be solved

Reasoning: it is the decision making process which decides to act on the basis of the information it receives, and in accordance with its own objectives to achieve its goals.

Communication: it is the interface used by the agent to communicate with its environment and with other agents.

Cooperation: defines the models of coordination and cooperation to interact with other agents in order to perform tasks for other agents. Till now we have discuss the characteristics of intelligent agent and now we are discussing the MAS.

A. MULTI-AGENT SYSTEM (MAS)

Various definitions have been proposed for the term multi-agent system (MAS). [20] defined a MAS as a loosely coupled network of problem solvers that work together to solve problems that are beyond the individual capabilities or knowledge of each problem solver. More recently, [12] defined MAS as a system composed of a population of autonomous agents, which interact with each other to reach common objectives, while simultaneously each agent pursues individual objectives [21]. The main characteristics of MAS, defined by [10], are: Each agent has incomplete information, or capabilities for solving the problem, thus each agent has a limited viewpoint; There is no global system control; Data is decentralized; Computation is asynchronous.

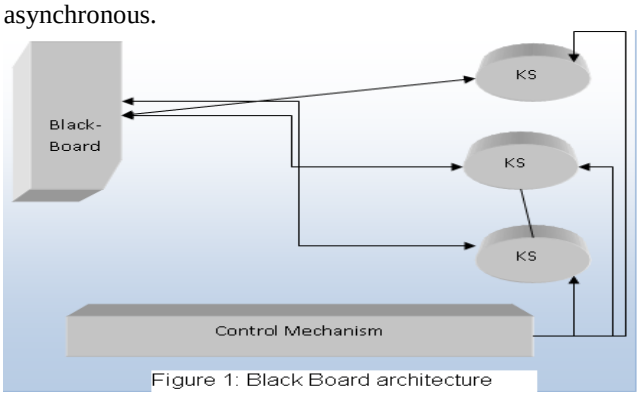

Figure 1: Black Board architecture

Two main multi-agent architectures have been addressed in the literature: blackboard and autonomous architectures

[10][17][19]). Early MAS were based on the blackboard model proposed by Hayes-Roth [22], See Figure 1. They developed the metaphor of a blackboard based on the idea that problem solving could result from the opportunistic activation of specialists, the KSs (knowledge sources). The activity of the KSs consists of putting down, modifying, and withdrawing solution elements within a common working area, called a blackboard. A centralized control mechanism is used to activate the KSs. The main drawback of the blackboard architecture is its relative inefficiency because of its very centralized control mechanism and its lack of local memory. As stated by [11], blackboard architectures cannot be considered as multi-agent systems as they do not respond to the characteristics of MAS. However, they are still used in many applications because of their ease of implementation. In autonomous architectures, illustrated in Figure 2**,** the agents are not controlled or managed by any other agents; rather they communicate and interact directly with any other agent in the system to achieve the global objective. Knowledge and control are distributed, in the sense that each agent embodies its own knowledge and control.

In order for MAS to solve common problems coherently, the agents must cooperate, coordinate, and communicate amongst themselves. Cooperation, coordination, and communication are central to MAS. Agents need to interact with other agents to achieve their objectives either because they do not have sufficient capabilities or resources to complete their problem solving alone, or because there are interdependencies between the agents that follow from being situated in a common environment [24]. These interactions can vary from simple information interchanges to requests for particular actions to be performed, cooperation (working together to achieve a common objective), coordination (arranging for related activities to be performed in a coherent manner), and negotiation (a process by which a group of agents come to a mutually acceptable agreement on some matter) [25].

B. MULTI AGENT CHARACTERIZATION

In above section We deals the concepts of agent and MAS now we are focusing on some more Special characteristics of MAS which makes it different from others.

B1. COOPERATION AND COORDINATION

Work on cooperative distributed problem solving began with the work of [20,26] where he defined that cooperative distributed problem solving studies how a loosely coupled network of problem solvers can work together to solve problems that are beyond their individual capabilities. Each problem-solving node in the network is capable of sophisticated problem solving and can work independently, but the problems faced by the nodes cannot be completed without cooperation. Cooperation is necessary because no single node has sufficient expertise, resources, and information to solve a problem, and different nodes may have expertise for solving different parts of the problem. According to [27], cooperation occurs when the actions of each agent satisfy either or both of the following conditions:

Agents have a possible goal in common, which no agent could achieve in isolation, and their action tends to achieve that goal.

Agents perform actions, which enable or achieve not only their own goals, but also the goals of agents other than themselves. There are several reasons why agents need to cooperate and coordinate their activities [19][17][28][13,31][29][30] Distributed expertise, resources, and information. No individual agent has sufficient competence, resources, or information to solve the entire problem. Many problems cannot be solved by individual agents working in isolation because they do not possess the necessary expertise, resources, or information. Different expertise may need to be combined to solve a very difficult problem that is out of the scope of any individual agent. Different agents may have different resources which all need to be scheduled to produce a final product. Finally different agents may have different viewpoints of a problem. Agents in MAS possess different capabilities and expertise. Coordination is necessary because with decentralization in agent-based systems anarchy can set in easily. No agent possesses a global view of the entire agency to which it belongs, as this is simply not feasible in any community of reasonable complexity. Consequently, agents have only local views, goals, and knowledge that may conflict with others.

Meeting global constraints: Global constraints exist when the solution being developed by a group of agents must satisfy certain overall conditions if it is to be deemed successful. There are dependencies between the agents. Interdependence occurs when goals undertaken by individual agents are related either because local decisions made by one agent have an impact on the decisions of other community members, or because of the possibility of conflict interactions amongst agents.

Efficiency, Cooperation and coordination can significantly increase efficiency. Even when individuals can function independently, thereby obviating the need for cooperation, information discovered by one agent can be of sufficient use to another agent that both agents can solve the problem twice as fast. The main issues to be addressed in inter-agent cooperation include the following [12]:How tasks are distributed and allocated among agents?

How to coordinate the activities of the agents to avoid conflicting situations (coordination)?

How can the overall problem-solving activities of the agents be optimized so as to produce a solution that maximizes the coherence metric? so for this question we are dealing the concepts if Collaboration by task allocation.

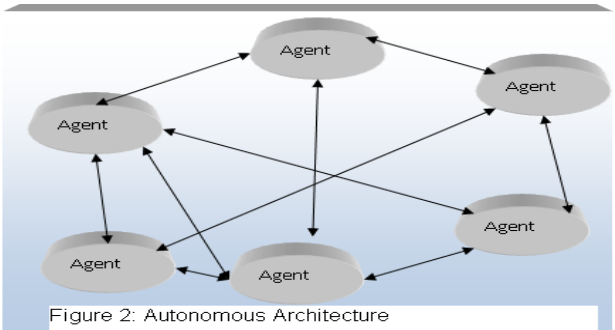

Figure 2: Autonomous Architecture

C. COLLABORATION BY TASK DISTRIBUTION OR ALLOCATION

Task distribution, also called task allocation, involves the definition of the organizational mechanisms through which agents can combine their skills to perform collective work. Task allocation can be managed by centralizing the distribution process or by distributing it among all the agents concerned [13][31][11][19][12].

C1. TASK ALLOCATION THROUGH MEDIATION

In centralized task allocation, allocation involves special agents, mediators or traders to manage the allocation of tasks to the agents, but not directly between agents. In this case, the mediator agent should have the necessary knowledge of all the agents in the system, including their competence and their availability. Figure 3 shows a mediator agent allocating a task T to a suitable agent through its knowledge about the agents.

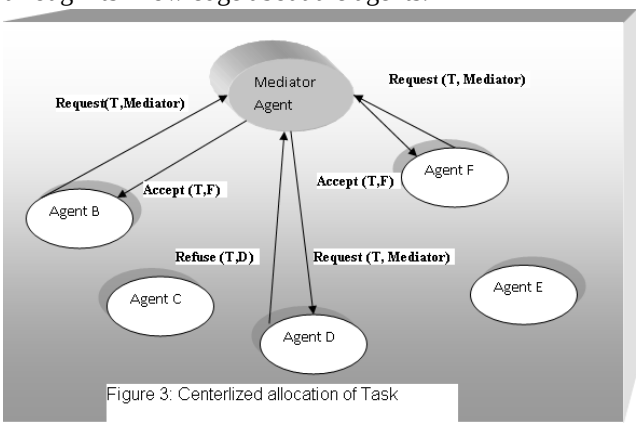

Figure 3: Centerlized allocation of Task

When the mediator agent receives a request from an agent to carry out the task T, then its sends an appropriate request to the relevant agents. If only one of the agents accepts to carry out the task, the mediator agent can simply allocate it to that agent. If more than one agent accepts the task, the mediator agent selects one according to relevant criteria. If all the agents refuse the task, the mediator agent informs the originator of the request that it could not find a suitable agent for the task. The advantage of task allocation through mediation is that the mediator possesses sufficient knowledge about its coordinated agents and their competence to ensure global consistency. The main drawbacks are its bottlenecks, and the centralized structure is not fault tolerant because failure of the mediator causes failure of the whole system.

C2.DISTRIBUTED TASK ALLOCATION

Through the Contract Net Protocol In distributed task allocation, each agent individually finds the suitable agents that are able to carry out its task without any degree of centralization. A classical technique for distributed task allocation is the contract net protocol. It is the most common and best-studied mechanism for distributed task allocation in agent-based systems [19][12]. The contract net protocol is a high level protocol for achieving efficient cooperation introduced by [32] based on a market-like protocol. The basic metaphor used in the contract net protocol is, as the name of the protocol suggests, contracting. Smith took his inspiration from the way that companies organize the process of putting contracts out to tender in public markets.

The contract net protocol received an enthusiastic welcome from the distributed artificial intelligence community. In this approach, a decentralized market structure is assumed and agents can take on two roles: a manager and a contractor. The basic premise of this form of cooperation is that if an agent cannot solve an assigned task using local resources/expertise, it will try to find other willing agents with the necessary resources/expertise to solve the tasks.

The tasks are assigned using a contracting mechanism. The manager agent advertises the task by a task announcement to other agents in the net, Figure 4. In response, contracting agents evaluate the task with respect to their abilities and engagements and submit bids. A bid indicates the capabilities of the bidder that are relevant to the execution of the announced task. The manager agent evaluates the submitted bids, and selects the most appropriate bidder to execute the task, which leads to awarding a contract to the contractor with the most appropriate bid. The contractor assumes the responsibility for the execution of the task. After a task has been completed, a report is sent to the manager. The advantages of the contract net protocol include the following: dynamic task allocation via self-bidding which leads to better agreements, it provides natural load balancing (as busy agents need not bid), agents can be introduced and removed dynamically, and it is a reliable mechanism for distributed control and failure recovery. In the basic contract net protocol, it is assumed that the manager waits for all the bids before evaluating them, this can cause the manager to wait indefinitely. To overcome this problem, [11] proposed to assign for each task announcement a deadline for the receipt of bids, and all those arriving after this date are directly rejected and are not taken into account in the selection step. Another problem is that between the time a bidder submits its proposal to the manager and the time it is awarded the contract or rejected, it is committed to carrying out the task. While submitting any future proposals, it must take into account the earlier commitments thought not awarded yet. To solve this problem, [33] extended the contract net protocol to a time bound negotiation framework or a finite time-guarantee protocol by attaching the commitment duration to the task announcement and bid messages. Contracts in the basic contract net protocol have been binding, i.e. once an agent agrees to a contract, it should honor it full commitment and has to follow through with it no matter how future events unravel. Although a contract may be profitable to an agent when it is established, it may not be profitable after some future events have occurred. In the majority of realistic scenarios, agents are situated in dynamic environments, where agents may become aware of new information, another agent may attempt to interact with it, and so on. As a response to these practical situations, the contract net protocol has been recently extended to the leveled commitment contracts as another method attaching commitments to the negotiation protocol for capitalizing on future events[30][34][35][36][37][38][39]. Instead of conditioning the contract on future events, a mechanism is built into the contract that allows de-committing. This is achieved by specifying the level of commitment by de-commitment conventions. De-commitment conventions describe circumstances under which an agent should de-commit. They also specify the appropriate contract's alternative actions to retain, rectify or abandon the commitments. If an agent wants to de commit, the agent can do so simply by proposing a de commitment convention. The method requires no explicit conditioning of the contract on future events: each agent can define its own conditioning dynamically. The coordination of actions was described by [40] as the set of supplementary activities which need to be carried out in a multi-agent environment, and which a single agent pursuing the same goals would not accomplish. According to [31], coordination is a process in which agents engage in order to insure their community acts in a coherent

manner. Coherent means that the agents' actions get well, and that they do not conflict with one another. For [30], coordination is the process by which an agent reasons about its local actions and the anticipated actions of others to try and ensure the community acts in a coherent manner. The main approaches that have been developed for coordinating activities are centralized planning, multi-agent planning, game theory, and negotiation [12][19][30][11].

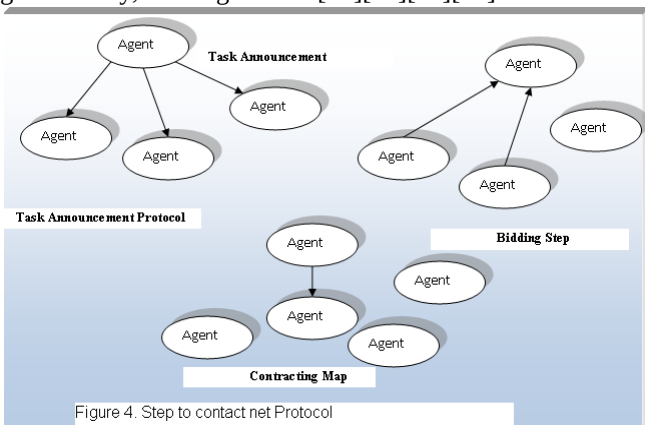

Figure 4. Step to contact net Protocol

D. CENTRALIZED PLANNING

This approach assumes the existence of a single planner agent which plans and distributes fragments of the plan to the slave agents whose role is limited to be executive only. This agent also handles the task allocation and coordination of agents. The slave agents must ultimately report their results to the master agent. In this case, while the master slave has full autonomy with respect to the slaves, the slaves have only partial autonomy with respect to their master. [30,31][11] pointed out that such centralized rigid structure is contrary to the assumptions of MAS. It presumes that one agent has a global view of the entire agency in many domains, which is an unrealistic assumption.

D1.MULTI-AGENT PLANNING

With the multi-agent planning approach to coordination, agents usually form a multi-agent plan, which specifies all their future actions and interactions with respect to achieving their goals, and may interleave execution with more planning and re-planning. Multi-agent plans are typically built to avoid inconsistent or conflicting actions. With this approach agents know in advance exactly what actions they will take, what actions their acquaintances will take, and what interactions will occur. There are two basic approaches to multi-agent planning: centralized and distributed. In centralized multi-agent planning, there is usually a coordinator agent who, on receipt of all partial plans or local plans from individual agents, analyses them in order to identify potential inconsistencies and conflicting interactions and grouped them to unsafe actions to create critical regions. The coordinator agent then attempts to modify these partial plans and combines them into a multi-agent plan where conflicting interactions are eliminated either by re-planning (re-arranging actions) or by inserting into individual plans communication primitives to synchronies the actions of the agents appropriately. For more details about synchronization see [11]. In distributed multi-agent planning, the idea is to provide each agent with a model of other agents' plans. Agents communicate in order to build and update their individual plans and other agents' plans until all conflicts are removed. The principal drawbacks of the multi-agent planning approach are that the amount of information exchanged between the agents is very high and communication is costly. In addition, the centralized planning approach presents the same limitations of centralized structure. The distributed multi-agent planning is very complex to implement in terms of detection and resolution of conflicting interactions.

D2.NEGOTIATION

Negotiation is a widely used technique for conflict resolution in multi-agent systems. It is the most fundamental and powerful mechanism for managing inter-agent dependencies. A basic definition of negotiation is that of [41], negotiation is the communication process of a group of agents in order to reach a mutual accepted agreement on some matter. For [12] negotiation proceeds in a series of rounds, with every agent making proposals, trade options, offer concessions at every round. The proposals that agents make are defined by their strategy, must be drawn from the negotiation set, and must be legal, as defined by the protocol. If agreement is reached, as defined by the agreement rule, then negotiation terminates with the agreement deal. [25] defined a generic framework of negotiation. In this framework, negotiation can be viewed as a distributed search through a space of potential agreements. For a given negotiation, the participants are the active components that determine the direction of the search. The minimum requirement of a negotiating agent is the ability to make and respond to proposals. To improve the efficiency of the negotiation process, the recipient need to be able to provide more useful feedback on the proposals it receives. This feedback can take the form of a critique (comments on which parts of the proposal the agent likes or dislikes) or a counter proposal (an alternative proposal generated in response to a proposal). From such feedback, the proposer should be in a position to generate a proposal that is more likely to lead to an agreement. Since negotiation involves exchanges of messages, negotiation protocols defining primitives of dialogue need to be defined. The most known and used dialogue primitives are found in the contract net protocol involving offers, bids, and contracts. Various negotiations methods have been defined in literature and most of them are inspired by human negotiation [30][31][42][43][19][25]:

• **Market-based negotiation**: the simplest and the most renowned negotiation protocol, and the most widely used in agent-based systems is the contract net protocol involving offers, bids, and contracts [13][31][25][19][40]. It is a high-level negotiation protocol that provides many advantages, and most importantly its flexibility and dynamic nature which suites industrial agent-based applications.

• **Plan-based negotiation**: this is based on cooperation strategies for resolving conflicts among plans of a group of agents. [46][15] described a three-phase cycle negotiation plan. This model of negotiation could be centralized or distributed. In centralized negotiation, an arbitrator agent receives the local plans of the individual plans, detects the conflicts, and initiates a negotiation process to overcome the conflicts. The arbitrator agent assists whenever agents are in conflict by reviewing their proposals and using their local plans to generate alternative proposals. Negotiation is an iterative process with a three-phase cycle:

The arbitrator agent makes a proposal to resolve the conflicts, which are evaluated by the agents in conflict. Arbitrator agent generates counter proposals if the agents in conflict are not happy with the proposals, or the original proposal may be simply accepted. Arbitrator agent submits the counter proposal for review by the agents.

In distributed plan-based negotiation, the three-phase cycle negotiation process is distributed among the agents. Every agent can be an arbitrator and can negotiate with other agents in order to arrive to a mutual agreement. Centralized negotiation presents the disadvantage of centralization of the negotiation on the arbitrator agent. Distributed plan negotiation is more effective but still very expensive in communication and difficult to implement.

Game theory-based negotiation: negotiation employs techniques based on game theory to structure and organize negotiation between the agents. The key concepts in the game theory approaches are the following: utility functions, a space of deals, strategies, and negotiation protocols. Utility is defined as the difference between the worth of achieving a goal and the price paid in achieving it. A deal is an action an agent can take which has an attached utility. The negotiation protocol defines the rules that govern the negotiation. The negotiation process involves an interactive process of offers and counteroffers in which each agent chooses a deal which maximizes its expected utility value. There is an implicit assumption that each agent in the negotiation process is an expected utility maximiser. At each step in the negotiation, an agent evaluates the other's offer in terms of its own negotiation strategy. The assumptions of game theory-based negotiation are untenable in real applications, and it is unlikely to suffice for industrial agent-based applications. Game theory-based negotiation presumes that two agents are interacting. In addition, it only considers the current state when deciding on their deal: past interactions and future implications are simply ignored.

AI-based negotiation: [47] considered negotiation as an iterative activity and she exploited case-based reasoning in this iterative process. She argued for a case-based approach, since human negotiators draw from the past negotiation experiences to guide present and future ones. [48]view negotiation as a constraint-direct search of a problem space using negotiation operators. These operators are drawn from human negotiation studies. They are used for relaxation. In their approach, negotiation involves two stages: a communication phase where all information is communicated to participating agents and a bargaining phase where deals are made between individuals or within a group. Agents negotiate via the relaxing of conflicts and constraints until agreement is reached. Alternatively, solutions may be modified until acceptable. The main limitation of this iterative approach stems from the fact that selecting relaxations to achieve a compromise is a major problem as no criteria are provided, and hence agents easily get caught in an infinite loop of exchanging offers.

D3.COMMUNICATION

In multi-agent systems, as with human beings, communication is the basis for interactions and social organizations. Communication enables the agents to cooperate, coordinate their actions, and carry out tasks jointly resulting in systems that are more coherent. A number of communication languages have been developed for inter-agent communication, and the most widely used ones are KIF (Knowledge Interchange Format) [49], KQML (Knowledge Query and Manipulation Language) [50], and ACL (Agent Communication Language) [51] KQML uses KIF to express the content of a message based on the first-order logic. KIF is a language intended primarily to express the content part of KQML messages. ACL is another communication standard emerging in competition with KQML since 1995. Nowadays, XML (Extensible Markup Language) started to show its performance as a language to encode the messages exchange between the agents, in particular in agent-based e-commerce to support the next generation of Internet commerce [14][17][11][19][53][12].

E.INFORMATION EXCHANGES THROUGH A SHARED DATA REPOSITORY

A common shared data repository, i.e. a blackboard, is used by the agents to write messages or to post partial results on, and obtain information from. This mode of communication is used in blackboard architectures.

E1.MESSAGE PASSING

Communication via some form of message passing is a widely used approach. In the message-passing approach, agents communicate with each other by sending asynchronous messages. Asynchronous communication is the primary mode of interaction in most agent-based applications. There are two basic message types[18]: assertions and queries. Every agent, whether active or passive, must have the ability to accept information. In its simplest form, this information is communicated to the agent by means of an assertion. In order to assume a passive role in a dialog, an agent must additionally be able to answer questions, i.e. it must be able to accept a query from another agent and send a reply to the agent by making an assertion. In order to assume an active role in a dialog, an agent must be able to issue queries and make assertions. With these capabilities, the agent then can potentially control another agent by causing it to respond to the query or to accept the information asserted. There are several methods of communication:

- **Point-to point**: one agent sends a message to another specific agent.
- **Broadcast:** one agent sends out a message to all other agents in the system.
- **Multi-cast:** one agent sends out a message to a selected group of agents.

E2. SPEECH ACT THEORY

Formalisms for representing communication in agent theory have tended to be based on speech act theory [6][9][19][11][18][50] as originated by Austin in 1962, and further developed by Searle in 1969. The key axiom of speech act theory is that communicative utterances are actions, in just the sense that physical actions are. They noticed that a certain class of natural language utterances or speech acts had the characteristics of actions, in the sense that they change the state of the world in a way analogous to physical actions. They observed that most things people say are not simply propositions that are true or false, but per formatives that succeed or fail. Thus the sentences uttered by humans during communication do not always simply assert a fact, but actually perform some action. Speech act theory uses per formative verbs to identify the illocutionary force of the utterance. Austin identified a number of per formative verbs including: request, promise, and inform.

The categorization of speech act by message types was initially motivated by Searl's classification of illocutionary forces into four categories in 1969:

• **Assertive**: providing information that affirms something (e.g. the machine is turned off).

• **Directive**: sending directives to receivers (e.g. turn off the machine).

• **Commissive**: accomplishing certain action in the future (e.g. I will turn the machine).

• **Declaratives**: declaring a decision or an announcement (e.g. I name this machine A).

Since the early 1990s, Speech act theories have directly informed and influenced a number of languages that have been developed for agent communication, such as KQML and ACL. In KQML and ACL each message has a per formative (a class of the message) and a number of parameters to describe the format of the message (sender, receiver, content, etc.). The most important differences between these two languages are in the collection of per formatives they provide.

In this section we have seen the concepts of Multi agent system and requirement of Multi agent based framework. Based on that we think why we are not using these concepts for most valuable assets 'Knowledge' for organization development. so we identified some agents for organizational knowledge management specially for software companies or engineering. After integration of various identified agents known as Multi agent based knowledge management (MABKM) [1]. Now we are dealing some concepts Of MABKM at it's integration.

III MABKM

In my research context 'Letrature Studies and survey [1], [2], [3]' we strongly observed the need of integrated concepts for Multi agent and Knowledge Management. Here in this section we bring the MABKM concepts for knowledge management. The details of this architecture are discussed here. Knowledge management (KM) is widely recognized as a critical issue in any kind of organization. **"**Knowledge is a mix of framed experience, values, contextual information, and expert insights that provides a framework for evaluating and incorporating new experiences and information." i.e., it is "the task of developing and exploiting an organization's tangible and intangible knowledge resources. It has to do with structuring information, ensuring that it is available to all potential users, easily accessible, and presented in such a way that all data relevant to the requesting users are effectively returned in a reasonable amount of time. When dealing with such issues one technology that comes in handy consists of software agents. Agents are software components featuring some nice properties that prove quite helpful to perform routine tasks, which are normally carried out by human users. These include processing of large quantities of information, searching over multiple sources spread all over the world, extracting selected portions of documents and so on. Agents can even move on the network carrying along the tasks they were assigned; they can even reduce processing times by self organizing into societies by spawning children agents acting in parallel [3]. We are Now Focusing on various agents and their contribution in KM practices. In the knowledge management domain, agents have been largely used in a multiplicity of projects and applications, to address a number of functions, roles and activities. So, we are introducing agents for the effective utilization of knowledge at various step of software engineering. Here, we are discussing agents and their characteristics in Context of KM. we have identifies knowledge agent as per my research finding.

A. DOMAIN KNOWLEDGE AGENT

'DK, as the name suggests, is associated with capturing, storing that information which is related to domain. Domain Expert stored their knowledge and expertise the database with the help of DKA into OKB. New Users can assess the experience and information here. Domain knowledge is that valid Knowledge used to refer to an area of human Endeavour, an autonomous computer activity, or other specialized discipline. Domain knowledge is knowledge about the environment in which the target system operates, for example, Software Agents. Domain knowledge is important, because it usually must be learned from software users in the domain (as domain specialists/experts), rather than from software developers. Expert's domain knowledge (frequently informal and ill structured) is transformed in computer programs and active data, for example in a set of rules in knowledge bases, by Knowledge Enablers. Communicating between end-users and software developers is often difficult. They must find a common language to communicate in. Developing enough shared vocabulary to communicate can often take a while. The same knowledge can be included in different domain knowledge. Knowledge which may be efficient in every domain is called domain-independent knowledge, for example logics and mathematics. Operations on domain knowledge are performed by Meta-Knowledge.

B. ORGANIZATIONAL KNOWLEDGE AGENT

OKA work as per the information carried out by the knowledge enabler on pro rata basis. This information can be used for internal monitoring purpose and organizational process to be improved.

C. PROCESS KNOWLEDGE AGENT

Process Knowledge Agent (PKA) record, evaluate and stored information into OKB. And this information is then utilized by the knowledge enabler as well as decision maker. The role of a process agent is a vitally important service for many businesses dealing with suppliers or tenders in the worldwide. Process agents accept service of notices, proceedings or documents on behalf of their overseas clients in situations where, usually because of contractual obligations, it is not possible for them to be served abroad. A process agent can act in a broad capacity for this company including but not limited to; acting as a process agent for court actions, receiving documents in connection with arbitration proceedings and receiving notices under contracts where an independent party is needed. A typical example of a process agency arrangement is where an overseas entity raises a loan from a city institution. Immediately the lending bank will require the appointment of a UK based process agent to receive formal notices should the borrower default on the loan.

D. DISTRIBUTED CASE BASED AGENT

Distributed Case base Agent Stores the information as per the Situation and case based scenario. This agent is responsible for Store, Monitor and Evaluate information into OKB and help to the Knowledge Enabler and well as Decision maker in the future project and its effectiveness. In information technology a reasoning system is any software application, hardware device or combination of software and hardware whose computational function is to generate conclusions from available knowledge using logical techniques of deduction, induction or other forms of reasoning. Reasoning systems are a subset of a broader category of intelligent systems. They play an important role in the practical implementation knowledge engineering and artificial intelligence. A reasoning system manipulates previously acquired knowledge in order to generate new knowledge. Knowledge is typically represented symbolically as informational facts and propositional statements that capture assertions, assumptions, beliefs and other premises. Sub-symbolic (connectionist) knowledge representations may also be used (e.g., trained neural nets). Reasoning systems automate the process of inferring or otherwise deriving new knowledge via the application of logic. In a concrete implementation, reasoning systems may support procedural attachments and built-in actions to process or apply knowledge within some given domain or situation. Reasoning systems have a wide field of application that includes scheduling, business rule processing, problem solving, complex event processing, intrusion detection, predictive analytics, robotics, computer vision and natural language processing. Reasoning systems apply logic in order to generate knowledge. However, they demonstrate significant variation in terms of systems of logic and formality. Most reasoning systems implement variations of propositional and symbolic (predicate) logic. These variations may be mathematically precise representations of formal logic systems (e.g., FOL), or extended and hybrid versions of those systems (e.g., Courteous logic). Reasoning systems may explicitly implement additional logic types (e.g., modal, deontic, temporal logics). However, many reasoning systems implement imprecise and semi-formal approximations to recognised logic systems. These systems typically support a variety of procedural and semi-declarative techniques in order to model different reasoning strategies. They emphasise pragmatism over formality and may depend on custom extensions and attachments in order to solve real-world problems. Many reasoning systems employ deductive reasoning to draw inferences from available knowledge. These inference engines support forward reasoning or backward reasoning to infer conclusions via modus ponens. The recursive reasoning methods they employee are termed ‘forward chaining’ and ‘backward chaining’, respectively. Although reasoning systems widely support deductive inference, some systems employ abductive, inductive, defeasible and other types of reasoning. Heuristics may also be employed to determine acceptable solutions to intractable problems. Reasoning systems may employ the closed world assumption (CWA) or open world assumption (OWA). The OWA is often associated with ontological knowledge representation and the Semantic Web. Different systems exhibit a variety of approaches to negation. As well as logical or bitwise complement, systems may support existential forms of strong and weak negation including negation-as-failure and ‘inflationary’ negation (negation of non-ground atoms). Different reasoning systems may support monotonic or non-monotonic reasoning, stratification and other logical techniques. Many reasoning systems provide capabilities for reasoning under uncertainty. This is important when building situated reasoning agents which must deal with uncertain representations of the world. There are several common approaches to handling uncertainty. These include the use of certainty factors, probabilistic methods such as Bayesian inference or Dempster–Shafer theory, multi-valued (‘fuzzy’) logic and various connectionist approaches.

E. ONTOLOGY AGENT

An ontology formally represents knowledge as a set of concepts within a domain and the relationships between those concepts. It can be used to reason about the entities within that domain, and may be used to describe the domain. In theory, ontology is”formal, explicit specification of a shared conceptualization" ontology renders shared vocabulary and taxonomy, which models a domain with the definition of objects and/or concepts, and their properties and relations. Ontology’s are the structural frameworks for organizing information and are used in artificial intelligence, the Semantic Web, systems engineering, software engineering, biomedical informatics, library science, enterprise bookmarking, and information architecture as a form of knowledge representation about the world or some part of it. The creation of domain ontologies is also fundamental to the definition and use of an enterprise architecture framework. There is also generally an expectation that there be a close resemblance between the real world and the features of the model in ontology. Contemporary ontologies share many structural similarities, regardless of the language in which they are expressed. As mentioned above, most ontologies describe individuals (instances), classes (concepts), attributes, and relations. In this section each of these components is discussed in turn. Common components of ontologies include:

Individuals: instances or objects (the basic or "ground level" objects)

Classes: sets, collections, concepts, classes in programming, types of objects, or kinds of things

Attributes: aspects, properties, features, characteristics, or parameters that objects (and classes) can have

Relations: ways in which classes and individuals can be related to one another

Function terms: complex structures formed from certain relations that can be used in place of an individual term in a statement

Restrictions: formally stated descriptions of what must be true in order for some assertion to be accepted as input

Rules: statements in the form of an if-then (antecedent-consequent) sentence that describe the logical inferences that can be drawn from an assertion in a particular form

Axioms: assertions (including rules) in a logical form that together comprise the overall theory that the ontology describes in its domain of application. This definition differs from that of "axioms" in generative grammar and formal logic. In those disciplines, axioms include only statements asserted as a priori knowledge. As used here, "axioms" also include the theory derived from axiomatic statements

Events: the changing of attributes or relations Ontology engineering (or ontology building) is a subfield of knowledge engineering that studies the methods and methodologies for building ontologies. It studies the ontology development process, the ontology life cycle, the methods and methodologies for building ontologies, and the tool suites and languages that support them.Ontology engineering aims to make explicit the knowledge contained within software applications, and within enterprises and business procedures for a particular domain. Ontology engineering offers a direction towards solving the interoperability problems brought about by semantic obstacles, such as the obstacles related to the definitions of business terms and software classes. Ontology engineering is a set of tasks related to the development of ontologies for a particular domain.

F. USER INTERFACE AGENT

It is one of the important agents this agent stores the User Interface worked performed by the previous personal and stored it into as a template of reference as per particular project as well as client interest areas. Marketability and good visibility need concepts and vision for user Interface. The user interface is the space where interaction between humans and machines occurs. The goal of interaction between a human and a machine at the user interface is effective operation and control of the machine, and feedback from the machine which aids the operator in making operational decisions. Examples of this broad concept of user interfaces include the interactive aspects of computer operating systems, hand tools, heavy machinery operator controls, and process controls. The design considerations applicable when creating user interfaces are related to or involve such disciplines as ergonomics and psychology. A user interface is the system by which people (users) interact with a machine. The user interface includes hardware (physical) and software (logical) components. User interfaces exist for various systems, and provide a means of: Input, allowing the users to manipulate a system, and/or Output, allowing the system to indicate the effects of the users' manipulation.

Generally, the goal of human-machine interaction engineering is to produce a user interface which makes it easy, efficient, and enjoyable to operate a machine in the way which produces the desired result. This generally means that the operator needs to provide minimal input to achieve the desired output, and also that the machine minimizes undesired outputs to the human. Ever since the increased use of personal computers and the relative decline in societal awareness of heavy machinery, the term user interface has taken on overtones of the (graphical) user interface, while industrial control panel and machinery control design discussions more commonly refer to human-machine interfaces.

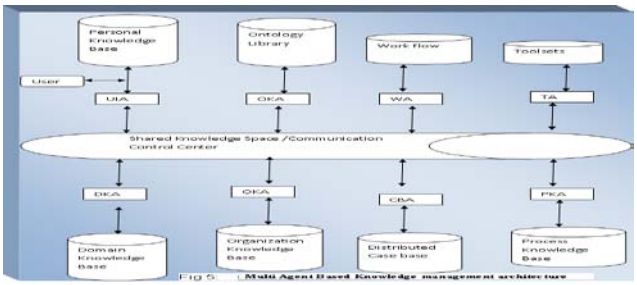

Fig 5. Multi Agent Based Knowledge management architecture

G. WORKFLOW AGENT

Workflow agent is responsible to monitor the various workflow activities and this activity and can be utilized to KM practices. Then after it can be utilize for further decision making. A workflow consists of a sequence of connected steps. It is a depiction of a sequence of operations, declared as work of a person, a group of persons, an organization of staff, or one or more simple or complex mechanisms. Workflow may be seen as any abstraction of real work. For control purposes, workflow may be a view on real work under a chosen aspect thus serving as a virtual representation of actual work. The flow being described may refer to a document or product that is being transferred from one step to another. A workflow is a model to represent real work for further assessment, e.g., for describing a reliably repeatable sequence of operations. More abstractly, a workflow is a pattern of activity enabled by a systematic organization of resources, defined roles and mass, energy and information flows, into a work process that can be documented and learned. Workflows are designed to achieve processing intents of some sort, such as physical transformation, service provision, or information processing. Workflow concepts are closely related to other concepts used to describe organizational structure, such as silos, functions, teams, projects, policies and hierarchies. Workflows may be viewed as one primitive building block of organizations. The relationships among these concepts are described later in this entry. The term workflow is used in computer programming to capture and develop human-to-machine interaction. The term workflow is more commonly used in particular industries, such as printing, and professional domains, where it may have particular specialized meanings.

Processes: A process is a more specific notion than workflow, and can apply to physical or biological processes, for instance. In the context of concepts surrounding work, a process may be distinguished from a workflow by the fact that it has well-defined inputs, outputs and purposes, while the notion of workflow may apply more generally to any systematic pattern of activity (such as all processes occurring in a machine shop).

Planning and scheduling: A plan is a description of the logically necessary, partially-ordered set of activities required to accomplish a specific goal given certain starting conditions. A plan, when augmented with a schedule and resource allocation calculations, completely defines a particular instance of systematic processing in pursuit of a goal. A workflow may be viewed as an (often optimal or near-optimal) realization of the mechanisms required to execute the same plan repeatedly.

Flow control is a control concept applied to workflows to divert from static control concepts applied to stock, that simply managed the buffers of material or orders, to a more dynamic concept of control, that manages the flow speed and flow volumes in motion and in process. Such orientation to dynamic aspects is the basic foundation to prepare for more advanced job shop controls, as just-in-time or just-in-sequence. In transit visibility is a monitoring concept that applies to transported material as well as to work in process or work in progress, i.e., workflows.

A workflow management system is a computer system that manages and defines a series of tasks within an organization to produce a final outcome or outcomes. Workflow

Management Systems allow you to define different workflows for different types of jobs or processes. So, for example, in a manufacturing setting, a design document might be automatically routed from designer to a technical director to the production engineer. At each stage in the workflow, one individual or group is responsible for a specific task. Once the task is complete, the workflow software ensures that the individuals responsible for the next task are notified and receive the data they need to execute their stage of the process. Workflow management systems also automate redundant tasks and ensure uncompleted tasks are followed up. Workflow management systems may control automated processes in addition to replacing paper work order transfers. If for example the above design documents are now available as AutoCAD but the workflow requires them as Catia an automated process would implement the conversion prior to notifying the individual responsible for the next task. This is the concept of dependencies. A workflow management system reflects the dependencies required for the completion of each task.

H.TOOLSET AGENT

These agents capture various toolset. Various tool set are available for Knowledge management like Document Manager Etc. this toolset agent capture information via GUI based Application. The aforesaid groups of agent possess some specific feature. They are autonomous, social, Objected oriented and Interactive. These agents are capable to communicate either for the performance of any tasks. These agents, when put together accomplish particular task. They are able to perform their part and release the task from complexity in effective manner. The figure 5 explains that multi agents are combined together to perform a task in effective ways. Shared knowledge space/Communication controls center are the areas through which these agents communicates with each other to perform the action. This multi agent's architecture provides assistance to users by performing the task in effective manner by the help of managing knowledge.

IV. OPERATIONAL IDEOLOGY OF MABKM

We have already discussed various agents and there functionalities and this section we are going to present the work currently underway for Multi Agent based Knowledge Management, whose purpose is to realize an information management and knowledge sharing system that allows users with different perspectives on a common set of concepts to access heterogeneous information spread over a number of distributed sources on the Intranet as well as Internet. MABKM is a process to manage Information as well as knowledge at effective manner for future needs of organizational development. In this part of the thesis we are dealing a combinational approach of agent discuss earlier in Figure 5 are discussing multi agent architecture for knowledge management that will known as Multi agent based Knowledge management architecture. In this architecture user will access the knowledge from database with personalization approach [4]. Every agent like DKA: Domain Knowledge Agent OKA: Organization Knowledge Agent (OKA), Process Knowledge Agent (PKA), Distributed Case Base Agent (CBA), Ontology Agent (OA), User Interface Agent (UIA), Workflow Agent (WA) and Toolset Agent (TA) will work for Knowledge capture, Distribution and Evaluation.

V. SE-MABKM

The discussion made in previous Section of this paper relates us to software engineering organization where knowledge is considered as a main resource and various process applied to manage such 'resource' (Knowledge management)[1][2][3][4]Software engineering Organization is typical knowledge-intensive discipline that evolves very fast and involves a large number of people, different phases and different activities. Under this discipline knowledge and experience gained during the course gives basic fundamental support to any project undertaken. Here changes are very fast and new technology and methods constantly appear and modified existing knowledge. The knowledge flows with high 'bandwidth' in SEO. Therefore such organization is very much benefited from Knowledge management. The Surrounding of SEO is termed as Software Engineering environment. SEO Comprises of various life cycles these life cycles describe the working of SEO. Under these life cycles knowledge and experience are captured, explicit and structured. The knowledge related to the any project is collected here. Such Knowledge is managed and thus making it more useful in accordance with the need of that project. This way knowledge management provides the organization with considerable benefit.

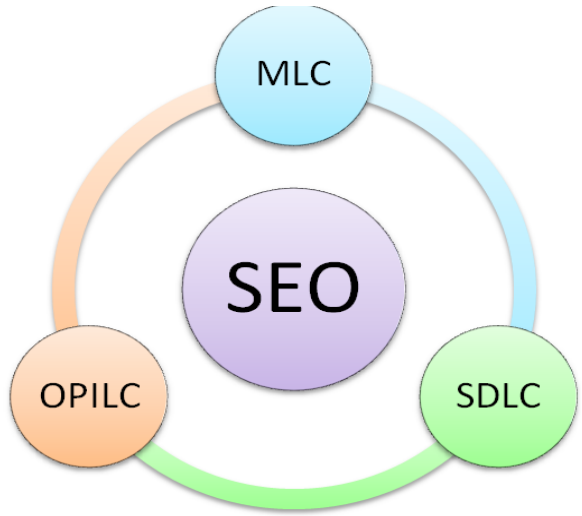

Figure 6: Software Engeeniering organizational life cycle

In this Section we are integrating the MABKM Architecture in SEE. Such Integration of MABKM into SEE is termed as SE-MABKM[2][3]. Before we discuss SE-MABKM and its Objective it is necessary to go through SEE. So, we discuss the working of SEE first. Among all the life cycle of SEE, we are focusing on the there only because these life cycles have direct impact on management policy and productivity.

These life cycles are[3]: Management Life Cycle (MLC) ,Organizational Process Improvement Life Cycle (OPILC) and System Development Life Cycle(SDLC) The details study of these life cycles we are dealing one by one .

A. MANAGEMENT LIFE CYCLE (MLC)

Management Life cycle perform management function. It controls the organization. This life cycle ensure that things are going as they should. Actual performance is compared with previously set goal here. Monitoring, Comparing and Potential Connecting is Covered under MLC. This ways MLC is very important part of SEO. Under MLC planning function encompasses defining the organizational goals. It

Establish an overall strategy for achieving these goal and developing the comprehensive hierarchy of plans to integrate and co-ordinate activities. MLC are responsible for designing the organizational structure. This includes determination of what tasks are to be done, who is to do them, how the tasks are to be grouped, who reports to whom and when decisions are to be made. We know that Knowledge is valuable resource in SEO. Under MLC knowledge and its management rotates from beginning to the end. MLC begins of Various Information (as Knowledge). Knowledge received here is huge in amount. It is not necessary that all received information is always useful, sometimes it is worthless, also may be, it is so that information received is not properly structured. So a Need of knowledge management arises. Knowledge management includes exploration, analysis and evaluation of knowledge. Management life cycles received information, explore, make analysis and evaluate (collectively called Process). These processes convert the information into useful knowledge. Such knowledgeable information provides help in decision making. The winning strategy for any knowledge management needs to be one that addresses many different organizational requirements; fiscal, cultural and operational, and yet has virtue of simplicity. Since, the flow of knowledge is very much wide at MLC here KM requires a strategy focus on valuable knowledge, Concentrating on knowledge that will contribute to the improvement of organizational performance. That is to say KM Strategy comes with certain complexity. The MABKM Architecture can be useful to tackle these complexities. The MABKM Architecture suggests KMS along with Multi agent System. Multi agents are the Group of Intelligent agent that allows each user involve in SEO, to assess the possible available information at the time of requirement. In MABKM Architecture, all agents autonomously collect and refine knowledge as information. This would provide the assistance and exploration of knowledge under MLC. Again these agents are capable of communicating with each other to achieve a common objective, this feature would help in analysis and evaluation of Information and thus providing help in decision making [3].

B. ORGANIZATIONAL PROCESS IMPROVEMENT LIFE CYCLE (OPILC)

Process is termed as collection of activities that takes one and more kinds of input and creates an output that is of value to customer. Examples of some process are procurement, sales and product development. A process is followed while managing organization where knowledge (as Information), as a right time plays a vital role in business activities. Knowledge has come to play an important role and regarded as a critical resource in view of contemporary, fiercely competitive business environment, both local and global. It is therefore, imperative that organization develop and continue to have 'current and appropriate knowledge' to ensure organizational effectiveness, efficiency and competitiveness. Knowledge systems have, therefore to be developed in the peculiar and specific organizational context. These activity related with development of knowledge are refer as process development[2][3]. The organizational process life cycle involves development of process. The OPILC begins from collection of various information. This information is updated in terms of Exploration, analysis and evaluation. By all this process knowledge become current and appropriate for decision making. These are then release as decision as per desired requirement. Today we are the part of global world where knowledge is everything. We need to make this knowledge current and appropriate for better surviving; accordingly OPILC is responsible for doing so. We have various processes Improvement Concept like TQM, Six Sigma, ISO, CMMI and BS etc. All these are providing guideline for process improvement in organization. It is known that OPILC was process development. Under this life cycle there is introduction of New Process (Knowledge) and deletion of obsolete one. OPILC rely on decision makers to produce mission critical decision on the basis of input from multiple domains. The decision makers' needs an understanding of many specific sub domains that influence decision-making, coupled with the experience that enable quick decisive and action based on such knowledge. This might be benefited form MABKM Architecture. The MABKM Architecture can help to determine where expertise resides in the organization. It can (MABKM Framework) create social environment. This would leverage recent advances in social network analysis. This analysis is based on observation of people, interaction patterns, communication and workflow. MABKM Architecture is design so that it identify expert by reveling the organizational knowledge network.

C SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)

System development life cycle is responsible for development of New and/or Legacy product or application. This is an era of globalization and organization needs to cope up with the dynamic and inevitable changes which takes place very often. Because of this change the competition among organization is becoming intense. The SDLC has been built around the organism metaphor in which organizations are analysis as if they were leaving organism operating in an environment which they need to adopt to ensure survival. The SDLC involves people, technology, information, Communication, Competition and social trends. System Development life cycle evolve planned change and improvement of organizations through the application of knowledge of behavioral science. This cycle is based on the systematically changes of processes. It focuses on managing the culture of organizations. working of SDLC involves many teams and numerous processes. It starts from communication among various teams. The process of communication under SDLC covers survey, feedback technique; involve gathering data, analyzing and Summarizing. The second phase of SDLC is planning. Here the gather data is bought again in front of employee and group for discussion to identify and solve problem and hence get solution (Software Requirement Specification). The planning also covers preparation of a roadmap for entire project, which include cost monitoring, scheduling and many more. The third phase of SDLC Covers Designing of Solution (SRS). Here solution or new application or development of legacy application is converted in the form of prototype, Forms/Function Design etc. The fourth phase is construction of proposed Solution (Front end and Back End) with the help of Desired Language and Database. In parallel testing is also there (what is going too developed and is Work is going on) as per the desired. It's also known as Implementation. SDLC Involves systematic application

of knowledge at various levels. SDLC runs with an objective to bring higher quality, productivity, adaptability and its effectiveness. SDLC encompasses all the activities which go into producing/providing new developed application.

VI. INTEGRATION OF MABKM INTO SEO

While discussing the above cycles of SEO we have seen that these encompasses all the activities, which go into producing/providing knowledge when organization's problem and opportunity. Here knowledge management is a structured problem solving endeavor consisting of distinct activities, which generally takes place in sequential and chronological order.

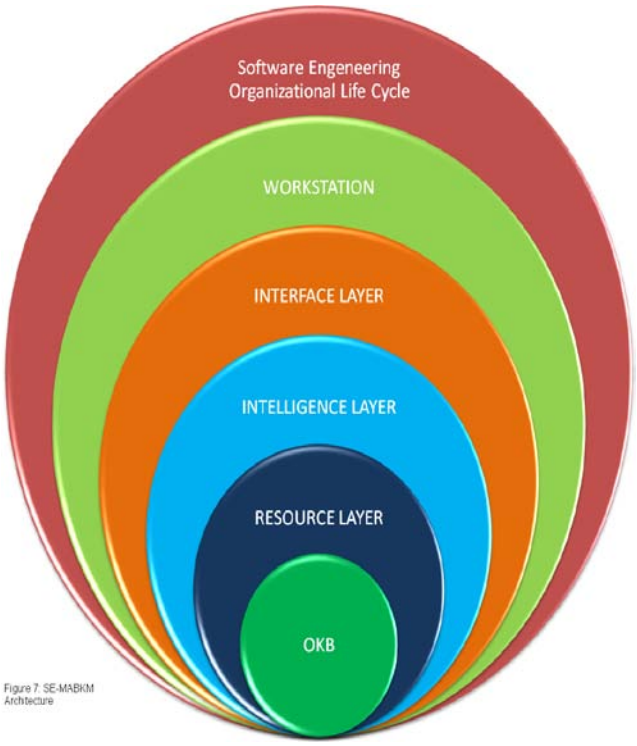

Figure 7: SE-MABKM Architecture

Life cycles begins with the collection of knowledge from various sources for instances people mind. If the organization could capture the mind of intelligent people, all it would need a better knowledge management strategy. Management of knowledge is art of creating and processing knowledge. We use the term 'art' because some companies do a very good job of creating, processing and managing their information, other do such a poor job that these tasks become a detoriating factor for the successes of the organization. The proposed MABKM Architecture can be useful for better knowledge management Strategy. Life cycle under SEO starts from acquisition of knowledge. Here tacit knowledge came to be regarded as a challenge for knowledge acquisition. It implies that human to human transfer (through collaboration and storytelling) would remain necessary because knowledge management software can represent only codifiable knowledge. If knowledge acquisition procedure provides proper scaffolding, expert came verbalized their tacit knowledge and express concepts that they had never explicitly express before, including information about their procedure and strategy. In other word, verbalizing ability does not seem to be unavoidable problem in knowledge gathering practice. Life cycle includes decision making. Also here knowledge intensive organization relies on decision makers to produce critical decision on the basis of inputs from various areas. The decision maker needs to have an understanding of many specific sub areas that influence making of a decision. Again the decision makers need to couple with the experience that enables quick and decisive action based on received information. In this context MABKM architecture has been proposed. MABKM Architecture when integrated to different life cycles of Software Engineering Organization can provide solution to tackle the above problem. The MABKM Architecture contains a set of Multi agent System that offers a new dimension for cooperation and coordination in distributed collaboratelly environments. Under this architecture each agent is autonomous in making decision on behalf of each function. This means that each agent can autonomously collects and process knowledge. We have already said that integration of MABKM into various life cycles i.e. SE is Called SE-MABKM. In further section of this paper we are going to discuss the regulatory framework of SE-MABKM. Before regulatory framework it is necessary to get familiar with all the term used in SE-MABKM Architecture.

VII. SE-MABKM ARCHITECTURE

SE-MABKM architecture is designed to store any type of data needed to convey that context of decision and the discussion involved in making decision. The various components of SE-MABKM Architecture are discussed herein under.

A. ORGANIZATIONAL LIFE CYCLE

In SE-MABKM, Organizational Life Cycle is a structure imposed on the development of a software product. It is often considered a subset of Management Life Cycles, Organizational Process Improvement Life Cycles and systems development life cycle. There are several models for such processes and Practices, each describing approaches to a variety of tasks or activities that take place during the process of Product Development. Responsibility of this life cycle to monitor the whole activity of Organization and try to improve it in unfavorable condition.

B. WORKSTATION

In SE-MABKM, workstation is a high-end User Interactive system for Organizational Users. Within these environments organizational users can interact, share and perform work as per assigned role and responsibility. They are commonly connected to a local area network and run multi-user operating systems and/or Wide area Network. The term workstation (group of heterogeneous and/or homogeneous member of organization life cycle) has also been used to refer to a terminal or a PC connected to a network. The workstation can be understood as a platform where users share their knowledge (Whether explicit or tacit)

C. INTERFACE LAYER

The top most of SE-MABKM is interface layer. It moves information in and out of the Knowledge management System. When this information is relevant, timely and actionable it represents knowledge. At the interface layer the KM System Users interact with the system to create explicit use, retrieve, and shared knowledge. The interface layer provides a universal mechanism for assessing all the layers

and underlying processes for delivering the information. agents present in this layer gradually learns how to better assists the user by observing and imitating the user, understanding user's interest and need.

D. INTELLIGENCE LAYER

This layer consist of multi agent Middleware Infrastructure which remain active all the time and behaves concurrently in and autonomous manner to achieve a common goal regarding consistently changing user interest and heterogeneous knowledge resources. Agent can check of the dynamic condition of Knowledge management Environment, reason to interpret those perception, solved problem, determine action and finally act accordingly. Some agent have an ability to learn from past mistakes at an explicit level which is something very much in line what a SE-MABKM is intended to help with.

E. RESOURCE LAYER

The bottom most layer in a SE-MABKM architecture is the one which contains organization intellectual assets (Knowledge and Experience). The considerable size of an information space and variety of resource residing in it, make network information asses a daunting task. Therefore, knowledge should be organized by a appropriate taxonomy for the ease of its retrieval. By enhancing the existing information sources with Meta data, the agents are able to recognize about information. This is because agents understand and agrees on the meaning of term the other agent is speaking because the term is officially described in a public ontology that can be refers to.

F.ORGANIZATIONAL KNOWLEDGE BASE (OKB)

We observed through a survey counts that in last few years companies have downsize and flattened their organization. Many of the employees who were laid off had been with the company for years. When they walk out the door, took experience, education, contacts and information with them. The companies have to put effort again to train the new employee. Maintaining organizational knowledge is a key management factors in retaining and promoting key employee but certain factors like distance, turnover and challenges of finding experts can make it difficult to maintain and shared the knowledge. In this context, SE-MABKM brings the concepts of OKB. Organizational Knowledge base (OKB) can be understood as a giant database which is included in SEE. OKB is design to store any type of decision and knowledge involves in making decision. It combines all knowledge of various life cycles of SE. It represent the combinational approach of all database (where refine knowledge is available) of various life cycle. The next section of this paper brings the theory based concept of working of SE-MABKM under various life cycles.

VIII.REGULATORY FRAMEWORK OF SE-MABKM

The working strategy of SE-MABKM can be divided into three layers. These are interface, intelligent and structured resource layer. All these layer of SE-MABKM contain set of Multi agents which are identified by their active roles that is serving user, cooperating work and etc. the collected information comes with the contact of interface layer. In interface layer a virtual work environment is created which enables tacit sharing of knowledge. Here the personal knowledge base agent intimates user interest and Build up user profiles. The main functions of personal assistance are: Collaboration with other workplace agent and reasoning over the suggested Information, responding to users query proactively (based on its knowledge base of prior request of users), actively updating it's knowledge based on the information fetched and users response. Making decision under certain condition that some specific information should be pushed to the user although it is not demanded explicitly. The second layer that is intelligent layer contains multi agent middleware infrastructure. Here, agents remain active all the time and behave concurrently in autonomous manner to achieved a common goal with consistently changing user interface and heterogeneous knowledge resource. All the information received explored and analyzed here. The intelligent layer contains task management agent, information processing agent and resource agent. These agents are responsible for the performance of work that comes under this layer. Each agent is identified with their feature. the functionalities of task management agent includes; behaving like a manger agent to handle the organization of all other agents which takes part in some specific KM Task, administrating and controlling the collaboration among users and agent during the execution of task. Information processing agent plays a significant role in life cycle of SEO. It helps in exploring and analysis of information the main function of information processing agents are; retrieving and merging information from heterogonous distributed information source filtering irrelevant content under information overflow condition. Evaluation of information in life cycles of SEO comes under structured resource layer, which is the Third layer. Here, by enhancing the exiting information source the agents are able to recognized and understand about information. Resource agents function under this layer. Their functionalities comprises of protocol availability through which knowledge resource accept queries. Extracting relevant information for a given request. Managing the status of whole knowledge repository, actively proposing resources to other agents based on their knowledge of other agent's need. Practically it convenient to have a separate resource agent for each resource, for that it is easy to include new resource one by one into a organizational knowledge base (OKB) and also exclude resource that are no longer required.

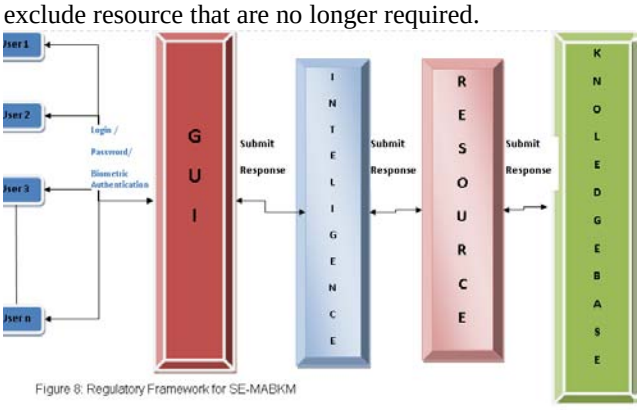

Figure 8: Regulatory Framework for SE-MABKM

IX. EFFECTIVENESS OF SE-MABKM FRAMEWORK

In many cases, decisions are too subjective and too large. Yet, decision makers can still use some help. SEO makes

the same difficult decision every month and every year. Difficult decision can require the participant of dozens of employees and analysis of terabytes of data it would be nice if the organization could keep the knowledge gained from every decision and applied in to the similar problem in future. SE-MABKM Framework is design to store any type of data needed to convey the context of the decision and the discussion involves in making decision. SE-MABKM is a theoretical framework for organizational knowledge management. This framework is proposing a combinational approach. Previously we have seen that MABKM Architecture can help to convert knowledge from distributed environment into useful information collaboratelly. Now, SE-MABKM is combining all the useful information from various life cycles into OKB. OKB is primarily a giant database. SEO Always seeks for a better management Policy to get optimum ROI and Maximize the effect of Balance sheet with the help of this SE-MABKM framework, organization can get right information at the right time to right people. As OKB Contains already processed information from various life cycles it can be very much helpful in performance of any task in a shot time. Even new user can avail benefit from this because it contains experience of previous task. Under SE-MABKM Framework all life cycles stores information about task (to be performed or have already been performed) together into OKB. If management needs to provide any 'knowledge' (which is related to any task) can take reference from decision of same kind of task available in OKB. SE-MABKM is providing theoretical concepts from knowledge base framework where every single data is collected by agent to store in a centralized way. There are some benefits of this theoretical framework, these are listed below:

- From SE-MABKM Framework, it is easier to developer to create new knowledge. In this ways the organizational memory is not closed, it is always evolving.
- A major concern for SE-MABKM is to capture information during software process without developers extra efforts. Thus, the SE-MABKM is actively into work process.
- SE-MABKM offers and open environment. Close system do not give organizational control over their own knowledge, since gap between knowledge creation and integration.
- SE-MABKM Users are no longer passive receiver of knowledge, but are active researchers, constructers, and communicators' of knowledge.
- In SE-MABKM knowledge can be constructed colloboratelly in the context of the work. Attention of Knowledge requires attention to people, including their task, motivation, and interest in collaboration. The heart of intelligent human performance is not the individual human mind but groups of mind interacting with each other along with tools and artifacts.
- The SE-MABKM Provides information according to workers needs and at the time when they need it. It plays an active role in knowledge dissemination.
- SE-MABKM monitors the actions of users as they work, and inform about potentially relevant knowledge for work.

The next section of this paper we are discussing the factors that should be taken into consideration while implementing SE-MABKM.

X. IMPLEMENTATION OF SE-MABKM

Simply throwing a computer on an employee's Desk does not make him or her instantly more productive or instantly smarter. We have to train people on the best use of the system. Prior to implementation of SE-MABKM we have to take a proper feedback from the working people. Feedback will show trained and untrained people of the organization. If more people are untrained regarding computational skill then we would trained first through internal trained people or outsource the trained people from remote organization. After the training completion take a exam of Computational skill and if people get or secured more then 60% or equivalent Grade then implement application into software Engineering environment. After implementation we have to prepare a questionnaire in fortnightly basis to get People mind knowledge. It will help to understand management regarding what is going on within the organization. Based on that information they can take appropriate decision or action to improvement in the organizational policies. After the storing information compare it from previous or old information and if any improvement then update otherwise wait from another forth nightly. And at the end of the month analyze it for appropriate decision. However, there are some factors that should be taken into consideration while implementing SE-MABKM. These factors leading to organizational success.

A. EMPLOYEE TRAINING

Numerous studies have pointed out on the importance of employee training to knowledge management implementation success. So, if a company wants to become a truly knowledge-based organization, it must start with quality training. This is true because in virtually every market, customers are demanding high quality, lower costs and faster cycle times. To meet these requirements, firms must continually improve their overall organizational performance. Rapid advances and technology and improved processes have been important factors helping businesses meet this challenge. However, the most important competitive advantage to any firm is its workforce – one that must remain competent through continuous training and development efforts. Training provides employees and managers the skills and information to fulfill their responsibilities. Improved performance is a strategic goal for organizations in order to achieve the bottom line purpose through training and development. For the same reason, a number of organizations have become or are striving to become learning organizations. This is because one of the reasons for the failure in effective work behaviors would be insufficient training to support knowledge management principles.

B.EMPLOYEE INVOLVEMENT

Employee involvement in making organizational decisions is a well-researched area. It describes how employees can contribute effectively to meeting the organization's objectives. It refers to the degree that employees share information, knowledge, rewards and power throughout the organization. Creating a high involvement organization involves making choice about organizational design that creates a world in which individuals know more, do more

and contribute more. The recognition of the importance of employee tacit knowledge is based on the assumption that successful performance improvement may not only depend on how work is organized, and the skill of the worker, but on the willingness of employees to convert tacit knowledge of the work process into continuous process improvement and innovation. Employee involvement is an array of techniques aimed at sharing information, knowledge, rewards and authority. It is thus the right way to gather knowledge from various levels of management and essential for an organization to survive.

C. OPEN AND TRUSTWORTHY SPIRIT OF TEAMWORK

Another factors for successful implementation of SE-MABKM is creation of Open and trustworthy spirit within team member. Teams are the units that actually carry out the work in many knowledge-intensive organizations. They are the ones that must access and apply distributed knowledge effectively. Teamwork is an essential source of the knowledge generation process. A well staffed team is crucial for successful implementation of SE-MABKM. This is because knowledge that individuals possess may be difficult to articulate because it is so deeply embedded in routines and practices that are taken for granted. By creating teams, it allows organizations to apply diverse skills and experiences towards its processes and problem-solving. After all, the focus of business and knowledge management application is on providing an environment in which knowledge workers of various disciplines can come together and create new knowledge.

D. EMPOWERMENT

Empowerment refers to a feeling of control and self-efficacy that emerges when people are given power in a previously powerless situation. It means eliminating the bureaucratic controls and creating a sense of freedom so that people can commit all their talents and energies to accomplish their shared goals. Empowered employees are given autonomy 'the freedom', independence and discretion 'over their work activities'. They are assigned work that has high levels of task significance important to themselves and others. Empowered employees also have control over performance feedback that guides their work and also a feeling of self-efficacy; that is, they believe that they are capable of successfully completing the task. Empowerment is regarded as one of the critical factors for successful implementation of SE-MABKM. If employees are to feel empowered, they need knowledge that will enable them to comprehend and contribute to the performance of the organization. This is because when individuals are empowered, they begin to take extra responsibilities to solve organizational problems by learning new skills in their jobs, which will eventually lead to them being more competent. Effective creation and sharing of knowledge will fail if employees do not have a sense of ownership in the overall aim of the organizational knowledge management system. After all, most organizational knowledge comes from the expertise, learning and experience of their employees. Through empowerment, employers can value their employees' expertise and help them communicate their knowledge by creating ways to capture, organize and share knowledge.

E.TOP MANAGEMENT LEADERSHIP AND COMMITMENT

Top management leadership and commitment are the most critical factors for a successful SE-MABKM, particularly in knowledge creating and culture sharing activities. Top management is increasingly recognizing that the knowledge inherent in an organization is an extremely valuable asset, and that it is no longer sufficient to leave it unmanaged and underleveraged. The effective management of knowledge is increasingly seen as an important basis for competitive advantage. In fact, poor leadership quality has been identified as a threat to successful implementation of SE-MABKM. Leadership commitment to the knowledge management process is essential. Leadership is responsible for creating the knowledge vision of the organization, communicating that vision, and building a culture that regards knowledge as a vital company resource. It is therefore important that senior management recognizes its importance. Without the support of top-level managers, the success of SE-MABKM activities is cumbersome. Only strong leadership could provide the necessary direction, where an enterprise will need to implement and effectively deploy SE-MABKM. To realize the potential of SE-MAKM, enterprise leadership must provide the proper environment to motivate its workers to enable the creation, organization and sharing of knowledge.

F. INFORMATION SYSTEMS INFRASTRUCTURE

Many researchers have supported the notion that effective and efficient knowledge management is unthinkable without information systems. A majority of business managers believe in the powers of computers and communication technologies that lead to knowledge management implementation success in organizations. An effective information systems infrastructure is necessary for the organization to implement the SE-MABKM. Information technology can provide an edge in harvesting knowledge Structural capital includes the databases, organizational charts, process manuals, strategies and routines and anything whose value to the company is higher than its material value. As a matter of fact, in literature it is also points out two most critical factors for the successful knowledge management project, one is the establishment of a broad information computing and communications. The second is being the utilization of the network technology infrastructure such as the Internet, Lotus Notes and global communications systems for effective transfer of knowledge. Knowledge bases and Intranets are the most popular ways of implementing knowledge management reports Information systems have provided knowledge management with capabilities that were not possible before. It has helped an organization to manage and leverage its knowledge systematically and actively. Without information technology and computers, knowledge cannot be stored. As storage forms an important part of knowledge management activities, the inefficiency of this part will disable knowledge management.

G. PERFORMANCE MEASUREMENT

Performance measurement is another milestone in SE-MABKM Success factors. Performance measurement as the collection of information about effectiveness and productivity of individuals, groups and larger organizational

units. Performance measurement is related to the key areas of the organization, such as expansion, innovation and productivity, which is critical to the development of prosperity of an organization. Many researchers have found a positive relationship between performance measurement and successful knowledge management implementation. Traditional management and measurement techniques that focus only on financial performance can be misleading and counter productive in a development environment. As such, the new theory of the organization must be expanded to capture the impact of knowledge on organizational performance. Besides financial performance, organizations can measure some of its intangible assets and use non-financial ratios or indicators for measuring management efficiency. in another side effective knowledge delivery can be achieved by finding the right system of measurements, as well as better ways of building and delivering the right information to the right people at the right time. Knowledge must be measured because the intellectual capital of an organization includes the brain of its employees, their know-how, the processes and customer knowledge that they create. Thus, it is clearly necessary to include performance measurement system as a key factor for the successful SE-MABKM implementation.

H. KNOWLEDGE-FRIENDLY CULTURE

In organization or at the time of SE-MBKM implementation a knowledge culture should be created. Due to Culture is a set of beliefs, which provides an identity for the organization, which in turn defines how the organization runs day to day. The set of beliefs includes organizational purpose, criteria of performance, the location of authority, legitimate base of power, decision-making orientation, leadership style, compliance, evaluation and motivation. There is a general agreement that a knowledge-friendly culture must be present or nurtured in order for knowledge management implementation success, after having primarily focused efforts on information technology, practitioners are now realizing the importance of the "soft" aspects of knowledge management initiatives. Culture practices reflect how the organizations view and facilitate both learning and innovation, including how it encourages employees to build the organizational knowledge base in ways that enhance values for the customers. Organizational culture as a concept is considered to be a key element of managing organizational change and renewal. Thus, since knowledge management is a radical innovation or changes the operations of an organization, it is regarded as an intervention to the organization's culture. It has been identified that the biggest challenge in knowledge management is not a technical one but a cultural one. To create a knowledge friendly culture, it is important to consider the cultural environment of a company before implementing SE-MABKM. An open culture built around integrating individual skills and experiences into organizational knowledge will be more successful. A culture of confidence and trust is required to encourage the application and development of knowledge within an organization.

I. KNOWLEDGE STRUCTURE

Knowledge creation can be based on numerous sources. Knowledge can be created individually, in groups and on an organizational level. Specifically, reliable, useful, up-to-date and timely knowledge can be captured and created by sharing knowledge with other members of work groups, suppliers and customers. So we have identified knowledge structure as one of the critical factor for successful knowledge management implementation. Since organizations are striving to improve their bottom line, many of them have realized the importance of customers and suppliers are their sources of product and service innovation. Many organizations have in fact brought suppliers and customers into the organization fold to share ideas for their product development and refinement decisions and to come up with new, innovative products and services. Organizations are striving to form strategic partnerships with customers so that the relationship becomes a long-term proposition. Knowing the importance of customers and suppliers, there must be a well-established knowledge structure, which includes knowledge about internal and external customers, suppliers as well as organizational work groups in order to implement SE-MABKM successfully

J. ELIMINATION OF ORGANIZATIONAL CONSTRAINTS

Successful SE-MABKM implementation may not be achievable if organizations cannot eliminate organizational constraints that present in an organization. This is because organizational constraints can affect negatively the perception and/or attitudes toward knowledge management success. Organizational constraints lead to inefficiency, ineffectiveness and powerlessness. They tend to create hierarchical bureaucracy with few incentives to innovate. Hierarchical bureaucracy means every task is broken into simple parts; each has the responsibility of a different level of employees, and each defined by specific rules and regulations. Organizational constraints result in not only a rigid preoccupation with standard operating procedures, but vertical chains of command and slow response as well. Rigid regulations, lack of incentives to be creative and lack of commitment in budgeting and funding would be problems for the SE-MABKM implementation. Thus, for a SE-MABKM to be successful, organizations must strive to eliminate all the constraints that impede SE-MABKM.

Till now, we have discussed our research finding in terms of MABKM and SE-MABKM concepts and their implementation strategy, Regulatory Framework and its implementation factor. In next chapter we are presenting the discussion and analysis of our research.

XI. DISCUSSION AND ANALYSIS

We will now discuss the results of our studies. We return to our research themes and discuss how our studies have contributed towards these. We will not discuss the concrete research questions, since those discussions have been covered in the individual papers[1][2][3][4]. For discussions on the validity of our contributions we also refer the reader to the reference part. For each research theme we discuss which of our contributions have an impact on it. We relate these contributions to the state-of-the-art, both showing how they fit with existing literature, and how they have extended the field.

A. EXPLORATORY STUDY OF SOFTWARE ENGINEERING AND KNOWLEDGE MANAGEMENT

Software Engineering is a discipline where 'knowledge' is consider as the most valuable asset. Knowledge carries some characteristics that make it different from any other assets. Some important of them are[2]:

Extraordinary: Knowledge is not subject to diminishing return when it is use, it is not consumed. Its consumers can add more to it, and thus increases the value.

Uncertain Value: it is difficult to estimate the impact of an investment in knowledge and

Rooted in time: the utility and validity of knowledge varies with the time.

In software engineering, the knowledge and experience, acquired, during the course of many years play a vital role. The software engineering is mostly benefited from knowledge; therefore, the management of knowledge is very much essential ingredient. The management of knowledge i.e. knowledge management is the process that helps organization to identify, select, organize, disseminate and transfer important information and expertise that are part of the organization memory and that typically resides within the organization in an unstructured manner. This structuring of knowledge enables effective and efficient problem solving, dynamic learning, strategic planning and decision making. The areas of KM, Which is often called knowledge management initiatives are

- knowledge creation
- Knowledge sharing
- Knowledge seeking.

Before discussion of KM initiative it is necessary to have a look at various Kind of knowledge they are:

Tacit Knowledge: These kinds of knowledge are usually in the domain of subjective, cognitive, and experimental learning. It is very much personal and difficult to formalize.

Explicit Knowledge: Explicit knowledge, on the other hand deals with more objective, rational, and technical knowledge.

Various Initiatives have taken in the same theme these are:

Knowledge creation: Knowledge creation is the generation of new insights. There are four mode of knowledge creation they are:

Socialization: It refers to the conversion of tacit knowledge to new tacit Knowledge through social interaction.

Combination: It refers to the creation of new explicit knowledge by merging, categorizing, reclassifying, synthesizing existing explicit knowledge.

Externalization: it refers to converting tacit knowledge into explicit knowledge and finally,

Internalization: It refers to create new tacit knowledge into explicit knowledge.

Knowledge Sharing: Knowledge Sharing is a will-full explication of ones ideas, insight, solution, and experience to other individual through intermediary.

Knowledge Seeking: It is often called knowledge sourcing. It refers to search of organizational knowledge by any mode.

Knowledge management approaches of knowledge management can be broadly classified into two categorized they are process and practice. the former approach attempts to codify organizational knowledge through formalized control process and technology and later approach assumed that the great deal of organizational knowledge is tacit in nature and that format controls process and technology.

In other side the knowledge management life cycle under software engineering undergoes 6 processes (Create, Capture, Refine, Store, Manage and Disseminate).

The knowledge in good knowledge management system never finished as, over the time the knowledge is updated. The; knowledge management framework in SE. Lots of models have been provided in relation with knowledge management. Some of them are [1][2][3]: Biosets Knowledge Category models ,Nonaka's Knowledge management models ,Headlund and Nonaka's Knowledge management Model,Skandia intellectual capital model of knowledge management, Demarest's Knowledge management model, Frid's Knowledge management models ,Stankosky and baldanza's knowledge management framework , Cought and Zander's knowledge management model.All the models are common in their perspective areas of creation, exploration and management of knowledge. By this exploratory study of Knowledge management and software Engineering, we think the goal of knowledge management is for an organization to aware to individual and collective knowledge so that it may make the most effective use of the knowledge it has. So in software engineering optimum knowledge practice can be occur if multi agent based knowledge management practices can be carried out for whole organizational life cycle.

The emergent general understanding is that systems, more than effective technology, represent indeed a novel general purpose paradigm for software engineering. Agents carried out all the actions and exhibit all the behavior within the knowledge flow. Agent can be placed into three categorized individual Agent, automated agent and Organizational agent based. Chapter 4 includes deep studies of these agents and their operational ideology (which results MABKM Framework) as well.

B. MULTI AGENT BASED KNOWLEDGE MANAGEMENT

Previous discussion of my research work relates with exploratory study of Software engineering and Knowledge management. This study is discussing about software engineering which includes software design, construction, testing and maintenance tasks. All these sector of SE required 'Knowledge'. So Knowledge is considered as a Valuable asset. Thus, the management of such valuable asset is very important. The management of knowledge often said knowledge management includes a lot's of challenges and issues. These challenges and issues can be tackle by the use of multi agent in knowledge management framework. This multi agent based knowledge management framework has been discussed in this section. The section explains multi agent based knowledge management model, Operational ideology of MABKM and Its framework. The section has presented different issues in knowledge that required to be managed and therefore leads to the concepts of KM. these issues are: Knowledge creation ,Knowledge Storage,Knowledge Distribution,Knowledge Application

We can also, Say that all the above are different phases of Knowledge Life cycle. The management of all these phases of knowledge is really a tough one. So, there are various models in this context (As Discussed earlier).They all focus on the fact that why knowledge management is necessary

but still many work is required on how management of knowledge can be done in that way so that the knowledge is utilized to full extent. One of the solutions can be addition of Intelligent Multi agent to knowledge management. The Multi agent technology is the emerging event in management of intelligent resources like knowledge. Multi agents System are the group of Smart agents. These agents are capable of defining their goals and action. They flow themselves as an effective key source to perform large complex task. Such as, workflow control, Knowledge search, and many more especially in distributed collaboratelly environment. The multi agent systems have recently emerge as a powerful technology to face the complexity of a variety of tasks. The concepts of multi agent system are very much wide, and there are number of agent. In this Section, Light is thrown in few of them. Reason behind it is that, they all have contributed significant role in MABKM Framework architecture (Discussed in chapter 4) that is the part of my research areas. They are:

Domain Knowledge agent: these are responsible for capturing, Storing Information related to domain. Travels, Banking etc are its example.

Organizational Knowledge Agent: These agent work for organizational internal knowledge and it's effective utilization

Process Knowledge agent: These agents record, evaluate and store information into organizational knowledge base.

Distributed case base agent: These agent stored information as per the situation and case based scenario.

Ontology agent: They offered a way to cope-up with heterogeneous representation of web resources.

User Interface Agents: These agent stores the user Interface work perform by previous user and stores it as a template of references.

Workflow agent: These agents are responsible for monitor the various workflow activities.

Toolset agents: These agents capture various agent and toolset.

In software engineering the knowledge is treated as 'resource' and its management i.e. 'knowledge management' is consider as a concept in which organization gather, organizes, shared and analyze its 'resource'. Such concepts can become clearer when it constitutes these aforesaid agents. In other words, if knowledge management framework is added to these agents it gives worth to the MABKM Architecture. Under this architecture Distributed knowledge management Structure constitutes a set of agent. These agents are recognized by their active function (their functions have been already explained).The MABKM is the concepts in which knowledge management is based on agent system. The operational ideology of MABKM architecture includes these agents which is Autonomous in making decision on behalf of each function. These agent autonomously gather, refine knowledge information in accordance with the requirement of a user. Share knowledge space and communication control center in the architecture, are the principle area of knowledge exchange and interaction during development task. The Agent under MABKM architecture is very important as they work for the said task and provide an effective platform for coordination and co-operation to help the team members to manage knowledge. The details structure of operational ideology of MABKM Framework Splits it into three layer, they are

Interface layer: Under this layer, the personal knowledge Based agent intimate user interest and build up user profile. Through, this layer, a virtual work environment is created which enables tacit learning.

Intelligent Layer: This layer consists of Multi-agent Middleware Infrastructure. The agents here, remain active all the time and behaves concurrently in an autonomous manner to achieve a common goal in consist wit consistently changing user interfaces and Heterogeneous knowledge resources.

Structure Layer: Such layer contains organizational intellectual assets. By enhancing the existing information source are with Meta data, the agent are able to recognized and understand about information.

The MABKM Framework supports the design and implementation of Multi agent module of flexible distributed system. It (MABKM Framework), Consists of there sub systems they are workspace (WAS), repository (ARS) and design supports (ADS). Workspaces (WAS): is an agent's operational environment on a distributed platform. According to the structured and function of MABKM to be design a lot of AWS can be installed on many platform. Repository (ARS): is a mechanism to manage and utilize the reusable agents.Design Support (ADS): Provides the facilities for designers to design and implement various agents that are based on MABKM Model. From a view point of implementation of MABKM the agents are classifying as repository and workplace agents. These agents that is to say ARS and AWS work together. the working strategy of MABKM is discussed as under : The AWS sends a message of requesting a service to ARS. In the ARS, the received message is sent to the repository agents to construct an organization of agents through AORP, to attend to the requested service. In this way, workplace agents are instantiated on a designated AWS as an instance of repository agents in ARS, to realize an executable component of MABKM. Thus, activating the workplace instance agents, the requested service is provided dynamically to the user. The workplace agents which run on the AWSs can communicate with each other by using the Communication/Cooperation Protocol (ACCP) which has a set of customized per formatives of the agent communication protocol of KQML. A multi-agent system is an ideal structure to support knowledge management, since each typical service required by the system can be implemented as a service agent, and each user can be assigned a personal assistant agent. The GUI (Graphical User Interface) enables the communication between the user and workplace agents. The agent comes under the name of workplace agents are discussed below:

Personal Agent (PA): The main functions of such agent are –collaboration with other agents and reasoning over suggested information.

Task Management agent: the function of these agent includes behaving like a manager agent to handle to organization of all other agent which take part in some specific knowledge management task.

Information processing agent: they perform retrieving and merging information from heterogeneous distributed information sources.

Resource agent their function comprises of protocol agents through which knowledge resources accept queries. The regulatory framework or we can say working structure or

operational ideology of MABKM is discussed. The above discussion brings the idea that creation, storage, distribution and application of knowledge can give better return on yield when they are properly managed. The agents are when used in KM model it became Multi agent based Knowledge management. The MABKM can be treated as theoretical concepts that can be used as a solution in relation of KM. The objective behind introducing such framework is the use of every segment of knowledge whenever required. MABKM Framework is given to support Software Engineering. This is because it is SE where knowledge is treated as the key resource. As discussed earlier the discipline of software engineering includes development, testing, implementation, Maintenance and whole management. All these discipline includes tools, methods and most important knowledge. Therefore MABKM Framework can prove its worth in SE. So here brings the concepts of SE-MABKM. SE-MABKM stands for support for Software Engineering via Multi agent Based Knowledge management.

C. SE-MABKM

SE-MABKM as the name suggests is a theoretical framework which is design to provide assistance in relation of knowledge management at SEO. SE-MABKM brings the concepts of Integration of MABKM architecture into various life cycles of SE. There are number of life cycles under SEO which describe working of SE, but concentration have been made in three of them mainly, this is because these life cycles have direct impact on Management Policy and Productivity. A brief Introduction of these life cycle are given below[2][3]:

Management life cycle: the management life cycle is responsible for the performance of management function in an organization. MLC is very much important, as it covers the broad areas of plan, organize, co-ordinate, control and command of information. The MLC under SEO is always put effort to gather information and make it useful further for other life cycles. The noteworthy point here is all the received information by the segment is really authenticated or just a waste. The MABKM Approach works here. It is discussed earlier that agent used in MABKM Architecture are vary much smart. So they provide help to put a line of demarcation in the information which is useful in accordance with the need. Therefore MABKM architecture is to be added in SEO to give SE-MABKM.

Organizational Process Improvement Life Cycle: The process Improvement is another milestone for SEO. The knowledge information plays a significant role under this cycle. These cycles involves introduction of advanced processes and deletion of obsolete one. The MABKM can be useful here this is because the agents introduced under MABKM Architecture are intelligent (As Explained Earlier). These agent processes the information accordance with the need.

System Development Life cycle: these life cycles involves development of new and legacy application and product such application is based on approaches even the product that develops in SDLC begins from communication between various teams. The team includes development, planner, tester and more according to the project requirement. The MABKM Framework significantly acts on gathering all the present and past expert opinions. The intelligent agent, under the MABKM Architecture stores such information. These stored data can be utilized at the time when needed most.

The integration of MABKM Architecture into various organizational life cycle of SEO (SEE) is known as SE-MABKM Framework. The SE-MABKM Framework includes following components:

Organizational Life cycle (OLC): in can be understood as a group of various life cycle under software engineering environment. The working of SEO is regulated and monitored by these life cycles. These life cycles covers all the activities of collecting and processing knowledge. OLC runs with an objective to control whole activity of SEO to provide better and improved service/Product.

Workstation: It can be understood as an effective platform where user communicate and shared the knowledge

Interface layer: it is topmost layer of SE-MABKM Framework. It takes information in and out of Knowledge management system. Here the users interact with system to create explicit use, retrieve and shared knowledge.

Intelligent layer: this layer is present in Middle of SE-MABKM Framework. This layer consists of a set of MAS which remain active all the time and behave concurrently in autonomous manner to achieve in a common goal.

Resource layer: it is the third layer. It is available in bottom of SE-MABKM. The resource layer consists of Set MAS that are able to evaluate the information.

OKB: OKB is a giant database which is design to store any type of decision and knowledge involve in making decision with effective reporting structure for organizational development. It includes the information of all life cycle under SEE. It contains knowledge and experience of various domain of SEO. So it is very helpful in performance of any task.

We have discussed the various components of SE-MABKM Framework. Now we are looking at the regulatory framework of SE-MABKM. Working of SE-MABKM can be categorized into three layers namely interface, Intelligence and Resource layer. Various MAS are present in these layers. These MAS are sets of smarts agent which are identified by their active contribution in serving users and cooperating work. These agents provide assistance in Managing knowledge. When the users share knowledge at workstation he comes with the contact of interface layers. Interface layers consist of personal agents. These personal agents intimate user interest and build up user's profile. All the knowledge received is taken into intelligent layers from interface layers. Intelligent layers included MAS namely TPA, IPA and RA. These agents are remain active all the time and behave concurrently in autonomous manner to achieve a common goal. The information is processed through various agents in intelligent layers. Now information again taken into structured resource layers. Here the information is evaluated. The present in this layers recognized, Understand and extract relevant information. Now the information became Manage knowledge and is stored into OKB. OKB Stores these process information from various life cycles. OKB includes new knowledge one by one in also exclude the information which is no longer required.

SE-MABKM framework is helpful in managing knowledge at software engineering organization in efficient ways. From

SE-MABKM it can easier for developer to create new knowledge, in this ways organizational memory is not closed, it is always evolving. Well, success factors of SE-MABKM depend upon the successful implementation of it. So, we have to consider some factors while implementing SE-MABKM. Some important among them are listed below:

Employee training: Training provides employee and managers the skills and information to fulfill their responsibility. Failure in effective work behavior would be insufficient training to Support SE-MABKM. Therefore, Employee training is the most important factors while implementing SE-MABKM.

Employee involvement: Employee involvement is an array of techniques aimed at sharing information, knowledge, rewards and authority. It is thus the right ways to gather knowledge from various levels of management and essential for an organization to survive.

Open and trustworthy sprit of team work: Creation of team allows organization to apply diverse skills and experience towards organization process and problem solving. After all SE-MABKM Focuses on Knowledge management application in which knowledge workers of various discipline can came together and create new knowledge.

Empowerment: through empowerment, employers can value their employee's expertise and help them communicate their knowledge by creating ways to capture, organize knowledge and shared knowledge.

Top Management Leaderships and Commitment: to realize the potential of SE-MABKM, Enterprise leaderships must provide the proper environment to motivate its workers to enable the creation, organization and sharing of knowledge.

Information system infrastructure: an effective information system infrastructure is necessary for the organization to implement the SE-MABKM. Without information technology and computers knowledge can not be stored. As storage forms are important part of SE-MABKM Framework, the inefficiency of this part will disable SE-MABKM.

Knowledge Structure: Knowing the importance of vendee and venders, there must be a well established knowledge structure, which include knowledge about internal and external organizational work group in order to implement SE-MABKM Successfully.

XII. CONCLUSION AND FURTHER WORK

Through this paper we have explored and reported research related to optimum knowledge management practice in software engineering organization by using the concepts of Multi Agent System. The research which we have carried out throughout this thesis has provided valuable insights into three main research themes, and resulted in eight major contributions. We now sum up our main findings and outline possible future works based on our results.

A. SOFTWARE ENGINEERING AND KNOWLEDGE MANAGEMENT: AN EXPLORATORY STUDIES

Our first research theme investigated the previous research in the field of knowledge management in software engineering. We have one major contribution in this theme:

C1: An extensive literature studies on knowledge management in software engineering.

Through a systematic review we created an overview of the research literature to identify what had been investigated and where the holes in the field were. We overviewed number of work have published in relation with Knowledge management framework, approaches to knowledge management ,knowledge management initiative, factors that are taken into consideration when implementing knowledge management strategy. Comparing the rest of our studies to the framework we used for integration of the concepts of Multi agent systems to knowledge management strategy .we also discovered that agents have possible goal in common, which know agent could achieve in isolation and their action tends to achieved goal.

B. MULTI AGENT SYSTEM

Our Second research theme overviewed the use of multi agents in KMS in SE. We focused on the functions and special features of multi agents within this theme. This lead the following

C2: Concept of MABKM

Through an action research study, we gained deep insight into the process of 'KM' with MAS in SEO. Our result where contrasted and strengthen by online survey, interview and systematic literature studies. Through this contribution, we realized the importance of focus on particular agents group properly to achieve satisfactory results.

C3: Proposal of Multi agent architecture for Organizational Knowledge Base (OKB).

We proposed the use of multi agent architecture for organizational knowledge base. The most important improvement was to increase optimum knowledge management practice.

C. SE-MABKM

Our third research theme investigated the SE-MABKM. In this context we investigated three specific life cycles in SEO, leading to the following contribution.

C4: Proposal of MABKM Architecture for Management life cycle (MLC)

Through, the action research studies; where, MABKM Architecture was applied to define the Management Life Cycle of Software engineering organization (SEO),We gained a deeper understanding of how the MLC affected the results of this architecture. Our main finding related to MABKM Architecture was its active contribution for managing knowledge in SEO, MABKM Produced list and description of activities.

C5: Proposal of MABKM Architecture for Organizational Process improvement life cycle (OPILC)

Through the action research studies where MABKM was applied to define the objective of OPILC in SEO, we gained the deeper understand of how OPILC affected the result of this architecture. Our main finding was the effect of MABKM Architecture in OPILC.

C6: MABKM Framework in System Development Life Cycle

Through an action research study, we gained a deeper understanding of SDLC in SEO. We also discovered how the SDLC affected the result of MABKM Architecture. The main finding related to MABKM Architecture was that the level of Knowledge management strategy in SEO affected

and consequently the final documented results. They produced list and description of activities.

C7: Design a model for Software engineering organization life cycle for effective utilization of organizational knowledge.

We proposed to design a model for software engineering life cycle to make their optimum knowledge management practice of internal knowledge. We prepared an open qualitative questionnaire to find out the general view of person working in knowledge communities via Open online survey. The result was more effective. It discovered deeper and more explicit cause which explained clearly the requirement of such kind of model. This model was named as SE-MABKM.

C8: Overview of Propose SE-MABKM Model along with it's utility.

Through the action of research studies we brought the concepts of SE-MABKM on the basis of analytical survey and research work. We discovered that SEO always looks for More ROI and better management policy. We concluded that SE-MABKM Could get right information to right people at the right time and thus contributing to increase ROI, Which were considered as the base line of knowledge management.

D. RESEARCH GOAL

Returning, finally, to our research goal for this thesis: How can Multi Agent Based Knowledge Management are applied to Software Engineering organizational life cycle for effective utilization of organizational knowledge? We found that by taking a Multi Agent Based knowledge management perspective on software Engineering life cycle, we could identify and increase learning effects, a key factor in getting developers to improve their practices. Our studies also showed that most research within software engineering has been directed towards the codification strategies, and that research on transfer of tacit knowledge through multi agent is lacking even though the learning effect on the individual level seemed greater through these. Further, our studies showed that communities of practice sprung up around OPILC, MLC, SDLC efforts. Participation in these communities seemed to be the key factor for the impact of the Organizational Knowledge base. A key challenge is to involve and keep the developers in these communities and make sure they don't drift out of them, once their involvement has ended. As we have seen there are many possible applications of knowledge management in software engineering, and we have tested but a few during the work on this thesis. But, as previous researchers have pointed out, there are many possible routes to the goal, and no single approach is necessarily the best for all possible contexts. Our studies have contributed towards the state-of-the-art by contextualizing some methods, but there are still a lot of possibilities for research within the field.

E. FUTURE WORK

Our three research themes lend themselves nicely to possible future directions for research we have started in this thesis. Our overview of the field does currently only include studies in industrial contexts, and can be greatly expanded by adding prescriptive studies from academia. There are also possibilities in extracting more information from the studies already identified, concerning contexts and method impact. As we found in our literature study on knowledge management and Software engineering organization. Software Organization likes to create Organizational Database and few have created in this direction but still some work is there. In further this work can be utilize by the knowledge management Communities to it practical implementation for Industrial purpose. In future we will design and develop this work 'MABKM and SE-MABKM framework' for it's practical implementation and issues and challenge at the time of implementation. In our point view in this knowledge economy environment, every organization needs SE-MABKM concepts for their knowledge base design i.e. OKB based on Multi agent based Knowledge management. We are concluding this research but still lot's of scope are there of MABKM and SE-MABKM model. at the time of Implementation it may be some challenge in terms of Infrastructure, technical storage, Technical Infrastructure like Bandwidth as well as some Security scope are also be considered. We are including some recent paper in terms of Nobel Data Encryption Algorithm (NDEA)[54] for security aspect of SE-MABKM as well as Integrated Web Enabled System for web based personalization approach for application development 'combinational approach', so management communities as well as research world can utilize this research work in their future application development or if they can like to create new and/or upgrade legacy system for Knowledge management Practices.

XIII. REFERENCES

[1] Dr.C.S Lamba, Ripu R Sinha, "SE-MABKM: Support for Software Engineering via Multi Agent Based Knowledge Management",International journal of Computer Science and technology Vol 2 issues 2. June 2011,pp382-386.

[2] Dr.C.S Lamba, Ripu R Sinha, "Knowledge Management and Software Engineering: an exploratory Study", International journal of Advanced Research in Computer vol 2 issue 4,july-aug 2011,pp456-468.

[3] Dr.C.S Lamba, Ripu R Sinha, "Operational Ideology of SE-MABKM for prop up of software Engineering" International journal of Advanced Research in Computer vol 2 issue 4,july-aug 2011,pp479-487.

[4] Ripu R Sinha et al., "Intelligent web enabled system" International Journal of Logistics and Supply chain Manangement Vol 3 Issues 3 Jan-Jun 2010.pp1-6

[5] Labidi, S. and Lejouad, W. (1993) De l'intelligence artificielle au syst`emes multi-agents. In: Writing Technical Report, INRIA, France.

[6]Wooldridge, M. and Jennings, N. R. (1994) Agent theories, architectures, and languages: a survey. In: Proceedings of ECAI-Workshop on Agent Theories, Architectures and Languages, pp. 1-32, Eds. Wooldridge, M. and Jennings, N. R., Amsterdam, The Netherlands.

[7]Wooldridge, M. and Jennings, N. R. (1995) Intelligent agents: theory and practice. Knowledge Engineering Review, 10 (2), 115-152.

[8] Franklin, S. and Gasser, A. (1997) Is it an agent, or just a program: a taxonomy for autonomous agents. In: Proceedings of the 3rd International Workshop on Agent Theories, Architectures, and Languages, pp. 21 35, Spring-Verlag.

[9] Wooldridge, M. (1998) Agent-based computing. Interoperable Communication
Networks, 1 (1), 71-97.

[10] Jennings, N. R., Sycara, K. and Wooldridge, M. (1998) A roadmap of agent research and development. Autonomous Agents and Multi-Agent Systems, 1 (1), 7-38.

[11] Ferber, J. (1999) Multi-agent systems: An introduction to Distributed Artificial Intelligence. Addison-Wesley, London.

[12]Wooldridge, M. (2002) An introduction to multi-agent systems. John Wiley & Sons, Ltd., Chichester, England

[13] Nwana, H. (1996) Software agents: an overview. Knowledge Engineering Review, 11 (3), 205-244.

[14] Brooks, R. A. (1986) A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, 2 (1), 14-23.

[15] Khoualdi, K. (1994) Filtrage d'alarmes pour un syst`eme automatis´e par une approche multi-agents. In: Writing Thesis, University of Paris VI.

[16] Noubissie-Tchako, J. F. (1994) Contribution `a la conception d'un syst`eme de pilotage distribu´e pour les syst`emes automatis´es de production. In: Writing Thesis, Valenciennes University, France.

[17] Ouelhadj, D. (1996) Etat de l'art sur les syst`emes multi-agents. In: Writing Master report.

[18] Weiss, G. (1999) Multi-agent systems: A modern approach to distributed artificial intelligence. The MIT Press, Cambridge, Massachusetts, London, England.

[19] Shen, W., Norrie, D. H. and Barthes, J. P. A. (2001) Multi-agent systems for concurrent intelligent design and manufacturing. Taylor & Francis, London.

[20] Durfee, E., Lesser, V. R. and Corkill, D. D. (1989a) Cooperative distributed problem solving. In: Handbook of Artificial Intelligence, pp. 83-147, Eds. Feigenbaum, E. A., Barr, A. and Cohen, P. R., Addison-Wesley.

[21] Oliveira, E., Fischer, K. and Stepankova, O. (1998) Multi-agent systems: which research for which applications. Robotics and Autonomous Systems, 7 (1-2), 91-106.

[22] Hayes-Roth, B. (1985) A blackboard architecture for control. Artificial Intelligence, 26 (1-2), 251-321.

[23] Ferber, J. (1997) Les syst`emes multi-agents: Aper¸cu g´en´eral. Techniques et Sciences Informatiques, 16 (8), 979-1012.

[24] Faratin, P., Sierra, C. and Jennings, N. R. (1998) Negotiation decision functions for autonomous agents. Robotics and Autonomous Systems, 24 (3-4), 159-182

[25] Jennings, N. R., Faratin, P., Lomuscio, A. R., Parsons, S., Sierra, C. and Wooldridge, M. (2001) Automated negotiation prospects, methods and challenges. International Journal of Group Decision and Negotiation, 10 (2), 199-215.

[26] Durfee, E., Lesser, V. R. and Corkill, D. D. (1989b) Trends in cooperative distributed problem solving. IEEE Transactions on Knowledge and Data
Engineering, 1 (1), 63-83.

[27] Doran, J. E., Franklin, S., Jennings, N. R. and Norman, T. J. (1997) On cooperation in multi-agent systems. The Knowledge Engineering Review, 12 (3), 309-314.

[28] Cammarata, S., MCArthur, D. and Steeb, R. (1983) Strategies of cooperation in Distributed Problem Solving. In: Proceedings of 18th International Joint Conference on Artificial Intelligence, 2, pp. 767-770.

[29]Bourron, T. (1992) Structures de communication et d'organisation pour la coopration dans un univers multi-agents. Thesis, University of Paris 6

[30] Jennings, N. R. (1996) Coordination techniques for distributed artificial intelligence. In: Foundations of Distributed Artificial Intelligence, pp. 187-210, Eds. O'Hare, G. M. P. and Jennings, N. R., Addison-Wiley.

[31] Nwana, H. S., Lee, L. C. and Jennings, N. R. (1996) Coordination in software agent systems. The British Telecom Technical Journal, 14 (4), 79-88.

[32]Smith, R. (1980) The contract net protocol: high level communication and control in distributed problem solver. IEEE Transactions on Computers, 29 (12), 1104-1113.

[33] Lee, K. J., Chang, Y. S. and Lee, J. K. (2000) Time-bound negotiation framework for electronic commerce agents. Decision Support Systems, 28 (4), 319-331

[34] Jennings, N. R. (1993) Commitments and conventions: the foundation of coordination in multi-agent systems. The Knowledge Engineering Review, 8 (3), 223-250.
[35] Sandholm, T. W. and Lesser, V. R. (1996) Advantages of a leveled commitment contracting protocol. In: Proceedings of the National Conference on Artificial Intelligence AAA, pp. 126-133, Portland.

[36]Anderson, M. R. and Sandholm, T. W. (2001) Leveled commitment contracts with myopic and strategic agents. Journal of Economic Dynamics and Control, 25 (3-4), 615-640.

[37]Panzarasa, P. and Jennings, N. R. (2001) Negotiation and joint commitments in multi-agent systems. In: Proceedings of the 2nd International Workshop on Modelling Artificial Societies and Hybrid Organisations,

Vienna, Austria. Also in http : // www. ecs .so to n .ac .uk / ˜nrj/pubs.html.

[38] Sandholm, T. W. and Lesser, V. R. (2001) Leveled commitment contracts and strategic breach. Games and Economic Behaviour, 35 (1 2), 212-270.

[39] Sandholm, T. W. and Zhou, Y. (2002) Surplus equivalence of leveled commitment contracts. Artificial Intelligence, 142 (2), 239-264.

[40]Malone, T. W. (1988) What is coordination theory. In: Proceedings of National Science Foundation Coordination Theory Workshop, MIT.

[41] Bussman, S. and Muller, J. (1992) A negotiation framework for co-operating agents. In: Proceedings of CKBS-SIG (Cooperative Knowledge Based Systems), pp. 1-17, Eds. Deen, S. M., Dake Centre, University of Keele.

[42] Kraus, S. (1997) Negotiation and cooperation in multi-agent environments. Artificial Intelligence, 94 (1-2), 79-97

[43]Faratin, P., Sierra, C. and Jennings, N. R. (1998) Negotiation decision functions for autonomous agents. Robotics and Autonomous Systems, 24 (3-4), 159-182.

[44]Beer, M., Inverno, M., Luck, M., Jennings, N. R., Preist, C. and Schroeder, M. (1999) Negotiation in multi-agent systems. Knowledge Engineering Review, 14 (3), 285-289.

[45]Shen, W. and Norrie, D. H. (1999) Agent based systems for intelligent manufacturing: a state of the art survey. International Journal of Knowledge and Information Systems, 1 (2), 129-156.

[46] Laasri, B. and Lesser, V. R. (1990) Negotiation and its role in cooperative distributed problem solving. In: Writing Technical report of Computer and Information Science Department, University of Massachusetts.

[47]Sycara, K. (1991) Cooperative negotiation in concurrent engineering design. In: Cooperative Engineering Design, pp. 269-297, Spring-Verlag Publications

[48]Sathi, A. and Fox, M. (1989) Constraint-directed negotiation of resource allocations. In: Distributed Artificial Intelligence 2, Eds. Gasser, L. and Huhns, M., Morgan Kaufmann.

[49]Genesereth, M. R. and Fikes, R. E. (1992) Knowledge interchange format. In: Writing Technical report (www.cs.umbc.edu/kse/kif/), Computer Science Department, Stanford University.

[50]Finin, T., Fritzson, R. McKay, D. and McEntire, R. (1994) KQML as an agent communication language. In: Proceedings of the 3rd International Conference on Information and Knowledge Management (CIKM'94), pp. 456-463, Gaithersburg, MD, http:/ / www .cs. umbc. edu /˜ finin //papers/kbks.pdf.

[51]Labrou, Y., Finin, T. and Peng, Y. (1999) Agent communication languages: the current landscape. IEEE Intelligent Systems, 14 (2), 45-52.

[52]Korzyk, A. D. (2000) Towards XML as a secure intelligent agent communication language. In: http://csrc.ncsl.nist.gov/nissc/2000/proceedings/papers/025.pdf.

[53]Turowski, K. (2002) Agent-based e-commerce in case of mass customization. International Journal of Production Economics, 75 (1-2), 69-81.

[54] Ripu R Sinha et al "novel data encryption algorithm" international journal of computer science issues, Vol 8, issue 4,No 2 July 2011 pp561-564

XVI. AUTHORS

Dr. C. S Lamba has PhD degree in Computer Science and currently working as a Head of department, RIET Jaipur Rajasthan India. He has more than 14 years of experience and holding diversified knowledge in the field of Information technology. He has published and Participated research paper / article in various conference / seminar globally.

Ripu R Sinha did Master Diploma in Software Technology (MDST) from IEC Software Ltd New Delhi in 1997. Then after he got Bachelor degree in Computer Application in 2001 from St Xavier College of Computer Studies Patna, India. He is M.C.A from India Gandhi National Open University India, in 2003, and currently pursuing Ph.D. degree in Computer Science engineering from National Institute of Medical Science University Jaipur India. He is an active member of International association of Engineers Hong Kong and International Association of Computer Science and Information Technology Singapore. He is having experience of SAP HR Consultant for ERP Technologies, Manager Technical recruitment from Tire 1 Companies like SAP Labs, IBM, NIIT Technologies, Accenture for bestinfo system ltd. Academic head and Sr. Teaching staff cum coordinator at Universal Institute of Computer technologies Delhi and Currently associate with Jaipur National University as a Assistant Professor at department of Computer and System Sciences. His research interests include Knowledge Management, Enterprise resource Planning, Software Engineering Practices and Project Management. He has published and Participated research paper / article in various conference / seminar worldwide. At present, He is engaged in Multi agent Based Knowledge Management