Proceedings of IDETC/CIE 2006
ASME 2006 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
September 10-13, 2006, Philadelphia, Pennsylvania, USA

DETC2006-99698

AN INTERNET BASED FRAMEWORK FOR MICRO DEVICES ASSEMBLY

N. Gobinath
Graduate Assistant
Center for Information Based Manufacturing (CINBM)
New Mexico State University
Las Cruces USA
gobinath@nmsu.edu

J. Cecil
Associate Professor
Industrial Engineering
Center for Information Based Manufacturing (CINBM)
New Mexico State University
Las Cruces USA
jcecil@nmsu.edu

T. Son
Assistant Professor
Computer Science
Center for Information Based Manufacturing (CINBM)
New Mexico State University
Las Cruces USA
tson@cs.nmsu.edu

ABSTRACT
This paper outlines the design of an Internet based collaborative framework to support the rapid assembly of micro devices. With the help of an agent programming language called 3APL, a distributed approach to achieving the life cycle of the various phases in the assembly of micro devices has been implemented. A discussion of the various agent resources created for a VE oriented approach is also provided in this paper.

I. INTRODUCTION

Collaboration is a key issue in today's engineering environments, with more and more engineering organizations forming partnerships to gain a better share of consumer markets. However, there is a downside to managing information exchange pertaining to business processes and transactions among the partners. One of these is related to the possession of heterogeneous skills and resources by the partner companies. Levels of disparity among the partner companies pose major problems when they collaborate. In order to collaborate and efficiently interoperate, many standards such as CORBA, DCOM etc. have been developed by various industry consortiums. Though these new standards and specifications meet the goal of facilitating interoperability among various heterogeneous resources, they lack in providing support to 'semantic interoperability' issues. True interoperability is possible only when disparities among the partner companies are resolved at the semantic level. Interoperability at the semantics level paves the way for seamless collaborations. Such collaborating partner companies may follow different specifications and representations to standardize their business processes and transactions. As those specifications and standards are well established in their business life cycle, there arises a need for a more flexible collaborative framework that will allow companies to form collaborations without causing any business turbulences. One such collaborative framework which is made possible is through the use of cognitive intelligent agents. These intelligent agents will mimic human agents in the engineering and business cycle. A programming language that allows developing agents with cognitive ability is 3APL. 3APL is an acronym for An Abstract Agent Programming Language (which was developed at Department of Information and Computing Sciences, Universiteit Utrecht, The Netherlands). In this paper, the design of an Internet based approach (which uses 3APL for creating a robust framework to support the facilitation of virtual enterprises VE) is discussed. While the domain of micro assembly is the focus of interest, this approach can be adapted for any manufacturing domain which requires the adoption of a VE based approach to product development.

A Virtual Enterprise (VE) can be described as a consortium of companies with diverse resources and expertise forming a temporary partnership in order to respond quickly to changing global market opportunities [3]. This paper describes a collaborative Internet based framework involving VE partners in the context of micro devices assembly (MDA). MDA is an emerging field that involves the assembly of micro devices or parts that are extremely small (in the order of 10^{-6} meters). Part designs, which are complex in shape and require the use of material of varying properties, cannot be manufactured using MEMS technology; in such situations, assembly of micron-sized parts is necessary. However, manual assembly of micron-sized parts is difficult and tedious. In this context, the design of computer controlled micro-assembly techniques and equipment becomes crucial. Further, MDA requires numerous heterogeneous resources that cannot be accumulated and provided by a single manufacturer or site. The distributed nature of resource requirements is a key driver towards the design of a collaborative framework for various manufacturing domains including the emerging domain of MDA.

In a hypothetical situation, consider a company Micro-mirage Inc. (MMI), which is interested in producing a miniaturized surveillance device involving micron sized components. MMI is not interested in assembling these micro components; but they want to access the Web and submit their requirements. Today, in response to their query, it is not possible for the Web to intelligently come with a plan to identify various partners who can work together to assemble MMI's product design. The research outlined in this paper lays the foundation to facilitate such a VE oriented approach to product realization.

The field of MDA is used as the domain context; there are specific reasons for the domain of micro assembly to benefit from such an approach. Micro devices can be used in a wide range of applications. These include imaging applications, as tools in medical surgery, as monitoring and surveillance devices in hazardous environments (such as nuclear reactors, etc.) and in the development of sensors for use in automobiles, computers and space systems. The various approaches and research efforts in micro assembly are discussed comprehensively in [4]. However, a major problem in MDA is the lack of resources as it's an emerging and expensive domain. Various industrial and research enterprises possess a diverse resources for addressing the assembly of a range of micro sized products. The domain of Micro Assembly provides an ideal context to demonstrate the capabilities of the collaboration framework (which is the focus of this paper). As Micro assembly resources are expensive and various partners typically possess different physical equipment and software resources, there is a need to form collaborative partnerships to address specific customer requests related to Micro Assembly. Given this context, the development of an IT based approach (which would address semantic interoperability issues for a VE based approach) is a necessity.

The importance of the research issues can be highlighted in the context of the following scenario:
Imagine a Virtual Enterprise, involving partner organizations coming together to produce parts, using MDA techniques and technology. The partner organizations may have different resources and areas of expertise. For instance, partner A and B may have the physical expertise and equipment (such as micro grippers, micro positioners and other sensors of different types and capabilities; note that A and B may possess different micro assembly capabilities); partner C may have simulation expertise and software tools related to that function; partners D and E may posses assembly planning skills and have different software module, which can generate candidate plans using specific approaches. With the help of an enterprise agent manager, the requirements can be studied; alternate plans involving one or several industrial organizations who will work (together) to produce MMI's required micro device can then be proposed. Depending on the nature of the micro device to be assembled, candidate assembly plans can be first generated based on specific design and the assembly resources and constraints; their feasibility can be studied using VE partner resources such as a Virtual Reality based simulation environment. Based on these outcomes, a specific assembly plan can be selected; finally, the assembly can be completed by a specific partner in a VE using innovative assembly and gripping approaches.

II. REVIEW OF OTHER RELATED RESEARCH EFFORTS

Various collaborative architectures and frameworks used in the development of virtual enterprise based manufacturing applications are discussed in this section. As many distributed manufacturing partners come together to form virtual enterprise, there arises incompatibility for their collaboration because of the variant data structures. This issue is addressed by [2]. National Industrial Information Infrastructure Protocols (NIIIP) is working with the Shipbuilding Partners and Suppliers (SPARS) Consortium and Integrated Shipbuilding Environment Consortium (ISEC) to apply the NIIIP technologies to the United States Shipbuilding community. In [5], a framework, based on an abstract object model (partitioning the application domain into components), providing distributed CIM system environment for semiconductor manufacturing domain is designed. This system is a web-based design tool with geometric representation module. An agile manufacturing information system, based on three-tier information communication model is described by [23] and [25].

In the manufacturing domain, agent based architectures and systems are deployed for various manufacturing activities; that include manufacturing control by [11], process coherence and collaboration by [12], scheduling by [16] and [20], decision support by [18], information retrieval by [19] and [21], distributed contracting by [2], conflict resolution in concurrent engineering design by [6], concurrent design and planning by Sun [22] and process planning by [8]. In [9], architecture based on the integration of CORBA and mobile agent for e-commerce is implemented.

The key role of agents in supporting distributed collaborations and negotiations is explained in [7], [12], [14] and [17]. The Next Generation Manufacturing Project, which can be viewed as a continuation of the Agile Manufacturing Project, is discussed by [13]; in this publication, four frameworks are discussed including the NIIIP architecture, the National Advanced Manufacturing Test bed, the Common Operating Environment and the SEMATECH CIM framework. A detailed review of distributed approaches and architectures in design and manufacturing contexts is provided in [3]. None of the above mentioned initiatives adequately address the issue of semantic interoperability. Other researchers have discussed the notion of a Semantic Web [26]; a semantic web based approach to MDA is also outlined in [27].

III. DESIGN OF INTERNET FRAMEWORK FOR MDA

In this section, the need for a collaborative framework in the context of virtual enterprise is discussed with the various identified activities. As discussed earlier, companies which are participating in a VE possess diverse resources which may vary from different product manufacturing capabilities to different software applications used by them. Also, each company may either follow their own way of specifying activities involved in the life cycle of product development or adopt any other industry renowned standards. This nature of diversification causes heterogeneity which slows down the process of forming collaborations among them. In order to hasten the process of forming collaborations and to achieve seamless flow of information exchange among the partners in a virtual enterprise, a collaborative framework is needed. Such a collaborative framework should have the following characteristics.

1. Knowledge about different specifications adopted by the participating companies.
2. Robust and flexible enough to include new specifications and standards.
3. Resolving the issue of heterogeneity

The kernel of the overall approach proposed can be summarized as follows: Suppose that there is a user company that wants to perform physical manipulation of micron sized cams using a micro gripper. Based on the user company's input and preference constraints specifications, a user agent searches the service directory for service providers. The service directory returns the list of service providers to the user agent. From the list of service providers, the best partners are selected for assembly sequence and path planning, simulation / analysis (such as virtual prototyping, determination of gripping forces, etc), and physical assembly. Match making algorithms are used to match the service provider descriptions with design specification, manipulation specification and simulation specification to find the best vendors. After identifying the VE partners, the user agent formulates a detailed plan of tasks to be achieved. Each VE partner in this collaborative framework is represented by their own software agent which has knowledge about the respective specifications and standards. In the proposed collaborative framework, the software agents are developed using 3APL.

An Abstract Agent Programming Language (3APL) is a new agent oriented programming language developed at; Universiteit Utrecht for developing agents with cognitive capability is given in [15]. The language comes with programming constructs that allows developing agents with complex mental states. A 3APL agent developed using this language is a tuple which is represented as:

3APL Agent = <B, G, P, A>,

where B is Belief Base, G is Goal Base, P is a set of Practical reasoning rules and A is an Action base. A Belief Base is a set of beliefs that an agent possesses about the tasks it has to perform and the environment in which it's going to act defines the belief base of an agent. In 3APL, a subset of first order predicate language is used to represent the beliefs in the form of formulas. The Goal Base defines the set of goals that an agent wants to achieve. Goals in 3APL language are of procedural type, which means that a goal can be considered as an imperative program defining a plan of action for an agent to execute. The language also can define simple and complex goals. A test goal, for example, defines criteria to an agent to evaluate its own beliefs about an external environment. Test goals not only define belief evaluation criteria to an agent but also bind values to any free variables available in the test goal. Complex goals (also called composite goals) are composed from basic or simple goals. Operators such as ';' and '+' are used to create complex goals. For example, "goal_1; goal_2" defines a sequence of goals and "goal_1+goal_2" defines disjunctive goals. A 3APL agent can manipulate its goals by using a set of practical reasoning rules. These reasoning rules define a plan of action for an agent to execute its goals. Using these rules, an agent can monitor as well as revise its goals in the goal base. Additional descriptions on the adoption of this 3APL architecture can be found in [10].

A. Design and Implementation

In this section, the design and implementation of collaborative framework is explained in the context of creating a virtual enterprise for distributed accomplishment of micro assembly tasks using robotic manipulation and other resources. The collaborative framework consists of a core module and ancillary modules from the web. The Core module is the foundation module of the framework which processes a user request by choosing the best partners and forms a virtual enterprise. It consists of two sub modules which include a User Agent Module and a Service Directory Agent Module

On obtaining the input and preference constraint specifications, the User Agent searches the service directory for respective service providers. The Service Directory is maintained by the Service Directory Agent which returns the list of service providers to the User Agent. The list of available service providers is then evaluated by assembly, manipulation and other criteria to choose the relevant partners. After selecting the relevant partners, the User Agent develops a collaboration specification for the selected partners and a Virtual Enterprise is formed.

In general, all the VE partners are identified by the designated 3APL agents called "Service Agents". The term 'Service Agent' is a generic term for any type of services. Each agent is identified depending upon the type of service they provide. For instance, a Service Agent called Design Agent of a design company provides the services related to manufacturing design such as developing conceptual design, detailed design and performing design validation. All the processes and transactions related to any services are processed through these Service Agents. Service agents have the capability to understand their own company's internal specifications and standards.

The semantic interoperability between the agents representing physical systems and other VE resources is addressed by developing ontologies. OWL, Ontology Web Language, is used to develop these ontologies. As MDA is an emerging domain, this is a very crucial step in the design of the overall approach and framework. Ontology defines the common words and concepts used to describe and represent an area of knowledge. An ontology is a specification of the conceptualization of a term. An ontology models the domain(s) of interest in a computer interpretable form. In this research, OWL is chosen to create ontology to describe the VE resources, etc. A better understanding of the relationships and attributes involving the resources and capabilities of the partner organizations were obtained used this approach. The domain of micro assembly was first divided into the various life cycle activities of interest: planning, simulation, analysis, assembly sequence generation, path planning and assembly. A structured set of relationships were defined along with introduction of new terms as needed which were relevant for the creation and use of the proposed approach. For example, physical assembly was described in terms of micro gripping, movement generation, etc.

The Service Directory agent maintained the description of all available services in the VE (for the MDA domain). OWL-S (Ontology Web Language for Services) is used to describe the services provided by the agents. Originally, the Oracle UDDI registry (which is part of the Oracle Application Server) was used as the service directory. However, this was found to be inefficient; subsequently, the Apache Xindice server was used for supporting this aspect of the implementation.

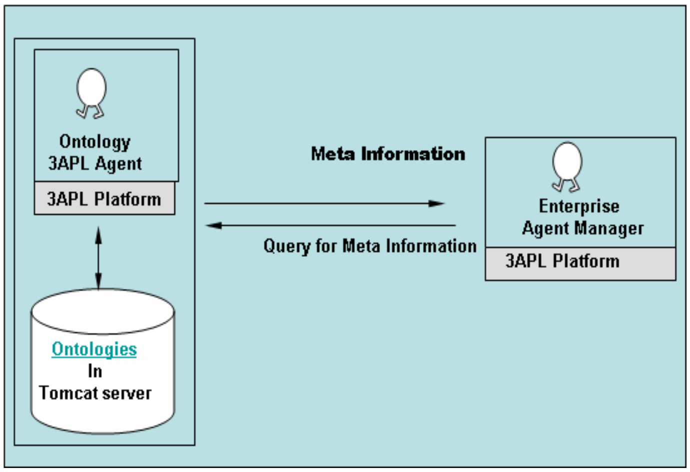

Fig. 1 a: Interaction between the EA Manager and Ontology Agent

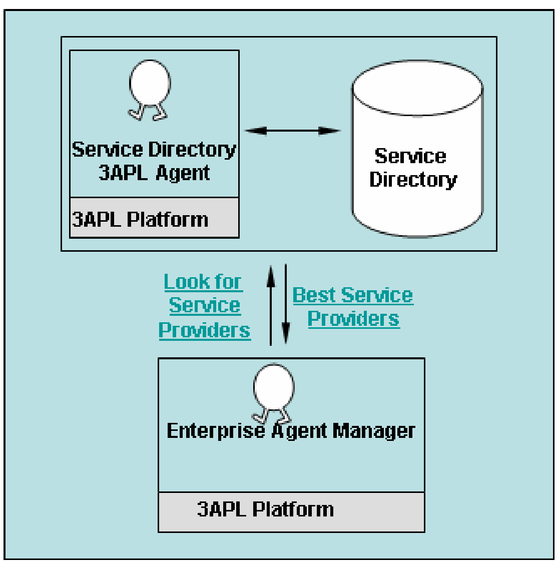

Fig. 1b: The Service Directory agent and the EA Manager

The services provided by VE partners were described using:

- Service Profile
- Service Process Model
- Service Grounding

The service profile describes capabilities of service providers. The service process model describes the required inputs, preconditions, resulting output and effects for a service and also describes the control flow during the execution of the service. The service grounding describes how to access the service and specifies the communication protocols, message formats, and port numbers.

The Enterprise Agent (EA) Manager (sometimes referred to as the VE Agent) coordinates the various activities in the collaborative framework, receives the input from the user agent, queries the ontology agent for meta information about the input, and requests the service directory agent for a list of best available service providers (figures 1 a and b). The Ontology agent provides the necessary meta information for the VE agent to progress on processing the input from the user agent; sample ontologies were created using Stanford's Protégé editor and were deployed on a Tomcat web server; modifications to existing ontologies can be completed through the ontology plug-in of this agent.

Ancillary modules are service provider modules. In the implemented framework, agents for assembly sequence and path planning, simulation / analysis and physical assembly play an important role in the collaborative process. Match making algorithms are used to match the service provider descriptions with design specification, manipulation specification and simulation specification of design; path planning and simulation partners form ancillary modules. These agents are capable of publishing their capabilities in the form of services into a service directory.

An overview of the capabilities and knowledge base of 3APL agents used follows. The user agent is the service agent for the user module, whose capabilities include understanding the company's request specification, generating the collaboration specification, identifying the relevant VE partners, providing self status and collaborating with other identified VE partner's agents. The associated knowledge base of this agent contains rules pertaining to business process specifications and standards, collaboration specification, criteria for selecting the best candidate VE partners and location information of various service directories.

The Service Directory Agent is the service agent associated with the service directory, where service providers in the MDA VE will publish their service descriptions. The associated knowledge base of this agent contains rules pertaining to service description specification and security specification. Other agents include the planning resource, the analysis / simulation resources and physical assembly resources.

B. *Description of Resources used in VE for MDA*

To demonstrate the feasibility of the framework and supporting concepts, a set of resources have been designed and implemented to mimic the functioning of a VE. The software resources are capable of performing top level collaborative planning, assembly sequence generation, 3 D path planning, virtual prototyping and analysis; the physical resources such as micro assembly cells are also part of this VE created to support the assembly of micro devices.

A brief description of some of these resources is provided. Two micro assembly cells have been considered. A description of Assembly cell 1 and 2 follow. Figure 2 provides an image of cell 1; this cell is capable of assembling pins and cams in the size range of 100 – 200 microns (diameter) and a few millimeters in length. Work cell 2 has a more dexterous manipulator and can manipulate parts in the order of 30 to 80 microns.

In micro assembly handling of micro components is very critical and mainly depends on the size and shape of the components. In cell 1, micro grippers from Zyvex Corporation are currently being used at VEEL Lab. Various grippers can grasp objects in the range of 160 to 500 microns. Micro positioners, cameras and other sensors are part of the physical micro assembly resources (see figure 2). A description of a physical resource and another software resource is provided to enable a better understanding of the VE resources considered in the implemented MDA context.

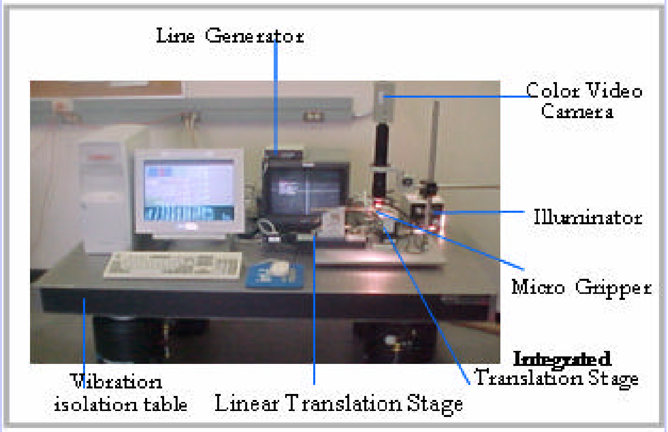

Fig. 2: View of a physical resource used in the VE implementation

Both the micro positioners are controlled from a computer and allow the user to control the position of stages, velocity, and acceleration of the micro positioner and receive feedback from the motors rotational encoder. The integrated micro positioner is composed of three separate linear micro positioners that are mounted orthogonal to each other and contains 3 DOF positioning stages. The images of the target environment and parts are used to guide the assembly operations. A Sony video camera is mounted vertically above the workspace plane and connected to a video microscope.

Apart from the work cells, virtual prototyping environments have been developed which form part of the VE resources. Figure 3 provides a view of one of these virtual reality based environments, which can be used to study alternates assembly and path plans, etc. This virtual environment was built using Open Inventor ™ and C++.

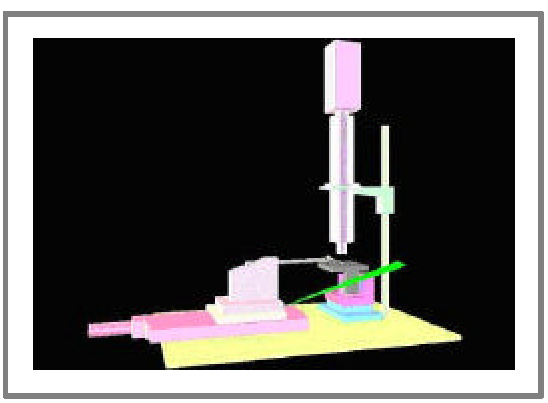

Fig. 3 A virtual assembly environment (one of the resources considered in the VE implementation)

The implementation activities sought to mimic diverse software and physical resources. Two virtual assembly tools were considered; the first uses a genetic algorithm based sequence generator and allows users to 'immerse' themselves (fig. 3) while the second virtual assembly tool uses approach based on Djekstra's algorithm to quickly build path plans for pick and place operations.

Some of the software resources within the collaborative framework include micro assembly sequence generators as well as 3 D path planners. One of the micro assembly generators used Genetic Algorithms to determine the sequence of assembling a target set of micro parts (this is detailed in [Cecil 04]). The 3 D path planners work in conjunction with the assembly sequence generation task and determine paths around fixtures and other obstacles on the WSP. For purposes of brevity, only an overview of one of the path planning resources (or software module) is provided. In this approach, the positions and dimensions of all fixtures, part feeders and obstacles as well as the start (source) / destination positions of the micro assembly task are provided as an input.

A summary of the key steps used by one of the path planning resources follows. First, enclosing boxes are created around each fixture, part feeders and other objects, which may be potential obstacles to be considered in a given sequence. Then a primary cell (PC) is built by creating an enclosing box around the gripper and the target micro object to be manipulated. A 3D (source to destination) region is constructed which is divided into reference cells whose size is equal to those of the primary cell. A test is performed to see if any fixture or obstacle coordinates lie within any of the reference cells. If an obstacle coordinate lies within the reference cells, then that cell is marked as 'F' (for Filled); else, that cell is marked as 'E' (for Empty). The position where a target part is picked up by the micro-gripper is known as the source position. Beginning from the centroid of the primary cell (at the source position) and ending at the destination position of the target object, a vector is constructed. This vector is referred to a Direct Path Vector – DPV. An equation of every vertical plane parallel to Y-Z plane is constructed; subsequently, the point of intersection of the vector DPV and the Y-Z plane is determined. Once the point of intersection is obtained, the reference cell in which this point lies is identified. A test is performed to see if the reference cell is marked as 'E' or 'F'. If the reference cell is marked as 'E', then the primary cell's new intermediary position is this empty reference cell's position. If the reference cell is marked as 'F', then a test is performed to determine whether the current position of the primary cell is greater in the X or Z direction. If the distance to be moved in X direction is greater than the distance to be moved in Z direction, there are 2 options: (1) if the reference cell in X direction is marked as 'E', then the object is moved to the next reference cell in X direction. (2) If the reference cell is marked as 'F', the feasibility of moving the object in the Z direction is evaluated in a similar manner and the primary cell's next intermediary position is determined. Once the primary cell's new (intermediary) position is determined, this position becomes the new start position. Subsequently, steps 4 through 6 are repeated until the target position is reached. This approach is followed to determine a feasible path plan needed to complete the assembly of each part from its source position to its destination or final position. Based on this path plan, the physical traveling distance is calculated for a given candidate sequence.

Based on the assessment of feasibility, the best path plan is selected by the Enterprise Agent Manager; subsequently, the detailed motion instructions are generated, which are then transmitted to the appropriate physical agent or resource, where the target parts are assembled.

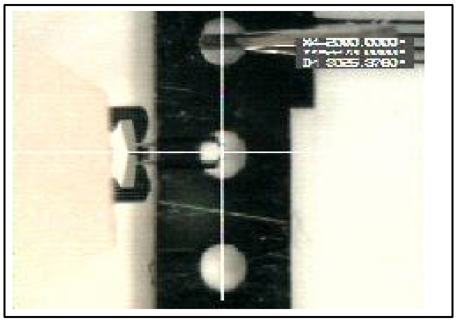

Fig. 4: A gripper inserting a pin into a comb spring

IV. CONCLUSION

To demonstrate the feasibility of the collaborative framework, several micro assembly tasks were completed using the framework discussed in this paper. Figure 4 shows one of these tasks completed, which involves the insertion of micron sized pins in holes on a comb spring.

In this paper, the design of an Internet framework (to facilitate the realization of a virtual enterprise based partnership for micro assembly) is discussed. With the help of ontologies, the semantic interoperability issues are address; an overall Enterprise Agent Manager works with other agents including an ontology agent, a user agent, service and other agents; software and physical resources are considered in a VE implementation relating to the field of MDA. Some of the problems encountered in adopting the 3APL Platform related to poor network communication and an inefficient directory facilitator (for the service directory; an Oracle UDDI registry was used initially; subsequently the Apache Xindice server was used.

Future research involves extending the capabilities of the VE for MDA as well as providing better evaluation criteria for comparing competing VE resources for MDA sequence generation, manipulation and simulation.

ACKNOWLEDGEMENT

Funding for the research activities outlined in this paper was received through a grant from the National Science Foundation (NSF DMI 0423907). Their assistance is gratefully acknowledged.

REFERENCES

[1] Antonio Puliafito, Salvatore Riccobene, Marco Scarpa: Which paradigm should I use? An analytical comparison of the client-server, remote evaluation and mobile agent paradigms. Concurrency and Computation: Practice and Experience 13(1): 71-94 (2001).

[2] Brewington, B., Gray, R., Moizumi, K., Kotz, D., Cybenko, G., & Rus, D. (1999). Mobile agents in distributed information retrieval. *Intelligent Information Agents*, 355-395.

[3] Cecil, J. A. (2003). *Virtual Enterprises*. In Internet Encyclopedia (editor: Hossein Bidgoli). New York: John Wiley & Sons, Vol.3, pp.567-578

[4] Cecil, J., et al, Gripping and Manipulation techniques for micro assembly applications. International Journal of Production Research, 2005, Vol. 43, No. 4, p.819-828.

[5] Cheng, F. T., Shen, E., Deng, J.-Y., & Nguyen, K. (1999). Development of a system framework for the computer integrated manufacturing execution system: A distributed object-oriented approach. *International Journal of Computer Integrated Manufacturing, 12,* 384-402.

[6] Cooper, S., & Taleb-Bendiab, A. (1998). CONCENSUS: multi-party negotiation support for conflict resolution in concurrent engineering design. *Journal of Intelligent Manufacturing, 9,* 155-159.

[7] Deshmukh, A., Krothapalli, A., Middlekoop, T., & Smith, C. A. (1999). Emergent Aerospace Designs Using Negotiating Autonomous Agents. Technical Report. Laboratory for Fundamental and Applied Research in Multi-agent Systems Report. Mechanical and Industrial Engineering, University of Massachusetts, Amherst, MA.

[8] Dornfeld, D., Wright, P. K., Wang, F. C., Sheng, P., Stori, J., Sundararajan, V., Krishnan, N., & Chu, C. H. (1999). Multi-agent process planning for a networked machining service. *Society of Manufacturing Engineers.* MS. MS99-175, MS99-175-1 – MS99-175-6.

[9] Foo, W,., Jie, W.,A reliable and flexible architecture based on CORBA and mobile agents for e-commerce,Info-tech and Info-net, 2001. Proceedings. ICII 2001 - Beijing. 2001 International Conferences on , Volume: 5 , 29 Oct.-1 Nov. 2001,Pages:211 - 216 vol.5

[10] Gobinath, N., Cecil, J. and Son, T. A Collaborative framework to realiz VEs using 3APL. Proceedings of the 2006 DALT Workshop (May 8, 2006), Hakodate, Japan.

[11] Heikkila, T., Kollingbaum, M., Valckenaers, P., & Bluemink, G. J. (2001). An agent architecture for manufacturing control: ManAge. *Computers in Industry, 46,* 315-331.

[12] Jain, A., Aparcio, M., & Singh, M. P. (1999). Agents for Process Coherence in Virtual Enterprises. *Communications of the ACM, 42,* 62-69.

[13] Jordan Jr., J. A., & Michel, F. J. (2000). *Next Generation Manufacturing: Methods and Techniques.* New York: John Wiley & Sons.

[14] Kadar, B., Monostori, L., & Szelke, E. (1998). An object oriented framework for developing distributed manufacturing architectures. *Journal of Intelligent Manufacturing, 9,* 173-179.

[15] Koen V. Hindriks, Frank S. De Boer, Wiebe Van Der Hoek, John-Jules Ch. Meyer, Agent Programming in 3APL, Autonomous Agents and Multi-Agent Systems, ACM, Volume 2 , Issue 4 (November 1999) Pages: 357 - 401

[16] Kouiss, K., Pierreval, H., & Mebarki, N. (1997). Using multi-agent architecture in FMS for dynamic scheduling. *Journal of Intelligent Manufacturing, 8,* 41-47.

[17] Lee, W. B., & Lau, H. C. W. (1999). Multi-agent modeling of dispersed manufacturing networks. *Expert Systems with applications, 16,* 297-306.

[18] Madejski, J. (2000). Agents as building blocks of responsibility-based manufacturing systems. *Journal of Materials Processing Technology, 106*, 219-222.

[19] Miller, L., Yang, J., Honavar, V., & Wong, J. (1998). Intelligent mobile agents for information retrieval and knowledge discovery from distributed data and knowledge sources. *Proceedings of the IEEE Information Technology Conference*.

[20] Miyashita, K. (1998). CAMPS: a constraint-based architecture for multiagent planning and scheduling. *Journal of Intelligent Manufacturing, 9*, 147-154.

[21] NIIIP website. http://www.niiip.org . 2002 (January 2003).

[22] Rus, D., Gray, R., & Kotz, D. (1996). Autonomous and adaptative agents that gather information. *Proceedings of the AAAI '96 International Workshop on Intelligent Adaptative Agents.*

[23] Song, L., & Nagi, R. (1997). Design and Implementation of a Virtual Information System for Agile Manufacturing. *Proceedings of the IIE Transactions on Design and Manufacturing, special issue on Agile Manufacturing, 29,* 839-857.

[24] Sun, J., Zhang, Y. F., & Nee, A. Y. C. (2000). Agent-based product design and planning for distributed concurrent engineering. *Proceedings of the IEEE International Conference on Robotics and Automation,* San Francisco, CA, 3101-3106.

[25] Zhou, L., & Nagi R. (2002). Design of distributed information systems for agile manufacturing virtual enterprises using CORBA and STEP standards. *Journal of Manufacturing Systems, 21*, 14-31.

[26] McIlraith, S., Son, T. C., and Zeng, H. Semantic Web Services, *IEEE Intelligent Systems*, vol. 16, no. 2, pp. 46-53, March/April, 2001.

[27] Cecil, J., and Gobinath, N., A Semantic Web based Framework for Micro Devices Assembly. Proceedings of the 2005 Flexible Automation and Intelligent Manufacturing (FAIM) Conference, July 18-20, Bilbao, Spain.