SAND97-2454C
SAND--97-2454C

Manufacturing in The World of Internet Collaboration

CONF-980514--

Michael J. McDonald and Raymond W. Harrigan

Sandia National Laboratories, Intelligent Systems & Robotics Center

RECEIVED
OCT 15 1997
OSTI

ABSTRACT

The internet and the applications it supports are revolutionizing the way people work together. This paper presents four case studies in engineering collaboration that new internet technologies have made possible. These cases include assembly design and analysis, simulation, intelligent machine system control, and systems integration. From these cases, general themes emerge that can guide the way people will work together in the coming decade.

KEYWORDS: Collaboration, Virtual Collaborative Environments, internet, intelligent system control, CAE, simulation.

INTRODUCTION

In every country, leaders are addressing how to make the internet improve their governmental and industrial effectiveness. Many deriving great benefit have focused their efforts on using the internet to foster new ways of working together. Indeed, billions of dollars are being invested in collaboration technologies ranging from Multi-User Dimensions to Distributed Interactive Simulations to Computer Supported Collaborative Work to Virtual Collaborative Environments. Researchers and practitioners are coming together as never before. The world is becoming a smaller place.

Four years ago, robotics researchers began building on the idea that since many machines are controlled through a computer-based interface, physical proximity is unnecessary. While many early remote experiments based on this idea made for great demonstrations, they failed to change the way our research was done. In a sense, the early work made use of network technology to control the machine, but failed to use that same technology to team the researchers.

At Sandia, where we specialize in developing intelligent robot system technologies, we've traditionally focused our efforts on putting new capabilities and tools into the hands of design and production engineers. In this tradition, researchers produce concepts and early prototype technologies, developers integrate these concepts and technologies into prototype systems, and application engineers move these technologies through maturation phases that yield commercial hardware and software. Each step includes long "shake-out" phases where users learn to apply new capabilities, report bugs, and gradually integrate the new approaches into their work processes.

However, the world is getting to be a very impatient place. Long delays between innovation and application are becoming less and less acceptable. Some customers

*Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheeed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

simply can't afford to wait. The solution would seem to be to find ways to deliver new tools faster, but that's the wrong mindset. The real problem is how to use new capabilities to produce value; right away. Our recent experiments with internet-enhanced collaboration have focused on producing value rather than merely technology tools.

This paper describes four case studies in collaboration. These include assembly design and analysis, simulation, operational control, and system integration. Each case study explores a unique approach to collaboration. Together, the cases show that new forms of collaboration allow teams to break problems into small, manageable parts, solve the sub-problems separately, bring the pieces together, and, in a collaborative manner, resolve the overlapping issues.

CASE STUDY 1: ASSEMBLY DESIGN AND ANALYSIS

Sandia researchers recently developed an automated assembly planning software tool that can script complex assembly sequences using ordinary CAD data [1 , 2]. Designers at a commercial partner's institution wanted to use the prototype software to check a new product concept for assembly in a big hurry. One solution, shipping the research-grade software over the net to allow the product designers to try it out, wasn't realistic. Having the Sandia researcher attempt to solve the product designer's problem was equally unrealistic. The best solution required bringing the best minds from each domain together to work synergistically.

In this case, the commercial product engineer sent a CAD file of their product to Sandia researchers, who quickly adapted their prototype geometry conversion software to convert the CAD data into the geometry representations needed by the assembly planner. The researcher made several on-the-fly software modifications to generate a first assembly sequence and transmitted it back over the internet to the product designers in about a day's time. That was only the beginning.

The product engineers viewed the video sequence and quickly concluded that the assembly method was promising but unworkable. In short, software alone couldn't solve the problem; insight from real people was needed. And here is where the real value of internet-based teaming was realized: once the product designers communicated their added constraints, the Sandia team was able to modify their software planner further to generate new assembly sequences that yielded a solution in a matter of hours. The combined intellects of researchers and product developers were thus merged into a single problem-solving team unbound by the large distances between them.

CASE STUDY 2: VIRTUAL COLLABORATIVE ENGINEERING

In developing controllers and intelligent machine systems, National Institute of Standards and Technology (NIST) engineers in Gaithersburg, MD, and Sandia engineers in Albuquerque, NM, often informally review one another's work. The engineers at both institutions normally use Deneb's Envision product for simulation. Their reviews often include sharing one another's machine models and controller software. Here, engineers electronically ship models across the internet, execute the simulations on their own workstations, and discuss the evaluation through phone and email conversations.

In a project with Sandia, Deneb modified Envision to provide a Virtual Collaborative Engineering (VCE) capability [3]. This capability links Deneb simulators across internet networks in a way that allows multiple users at diverse locations to have their simulator interactions transmitted and shown to all connected users. Any VCE user can assume control of a model or simulation to make changes, or view changes made by others. This capability creates a virtual conference room and allows teams to evaluate product design concepts, make product design versus manufacturing process trade-off decisions, and design products for easy maintenance.

The first practical use of the VCE software was in June of 1996. Here, NIST shipped Sandia a new model of a hexapod machine tool that the Sandia engineer had never seen. Sandia and NIST engineers, communicating by phone, loaded copies of the model into simulators on their respective computers and at the same time connected their simulators to a *VCE hub* collaboration server. At this point, the NIST engineer simply clicked a button to establish control of both simulators and ran the simulation through the machine's normal cycle. While the simulation ran, both engineers saw each simulation step at exactly the same time and from the same viewpoints. Because the simulations were synchronized, the engineers were able to discuss the reasons for particular motions without having to wonder what the other was seeing. Once the simulation was completed, the Sandia engineer took control of the distributed system to test whether the cutting tool could reach all sides of a particular part. When the linked simulations showed both engineers that the part was not fully reachable, the NIST engineer took control and moved the part onto a fixture that better centered it within the machine's reach to demonstrate that the part could, in fact, be machined on the hexapod.

CASE STUDY 3: VIRTUAL COLLABORATIVE CONTROL

In 1990, Sandia began developing Graphical Programming to enable robots to be used in semi-structured and unstructured environments that are typical in nuclear waste and nuclear contaminated facility cleanup. In these environments, detailed planning activities must be closely synchronized with machine operations. Developing plans and programs for these tasks requires detail environmental and geometric knowledge that is often, as in the case of excavation, dependent on completion of prior tasks. The result is the need for rapid problem solving in a dynamic, changing environment. Graphical Programming meets this need.

In Sandia's Graphical Programming systems, a wide variety of task plan prototypes are pre-programmed for efficient detail planning, simulation, and execution [4]. However, some tasks lack sufficient structure for general solutions and, as a result, considerable time and expertise are required to perform these tasks. In response, Virtual Collaborative Control (VCC) technologies were developed to let the operator continue working on easily solved problems while letting specially trained collaborators solve the time-consuming problems.

In VCC systems, as diagramed in Figure 1, collaborators are remotely located and only one, the key operator, need be near the machine control console. Other collaborators might coordinate operations, monitor other's planning and machine use, or even run the machine. In Sandia's VCC systems, Graphical Programming is used to control the robots. Visualization [5] and computer video transmission tools allow any collaborator to monitor planning, testing, motion previewing, and eventual robotic motion.

In July 1996, Sandia, with support from the U. S. Department of Energy Robotics Technology Development Program, completed an experimental VCC testbed [6]. This testbed allowed a team of robotics researchers from Sandia, the Pennsylvania State

University, Case Western Reserve University, and the University of New Mexico to develop and test new ideas in collaborative control.

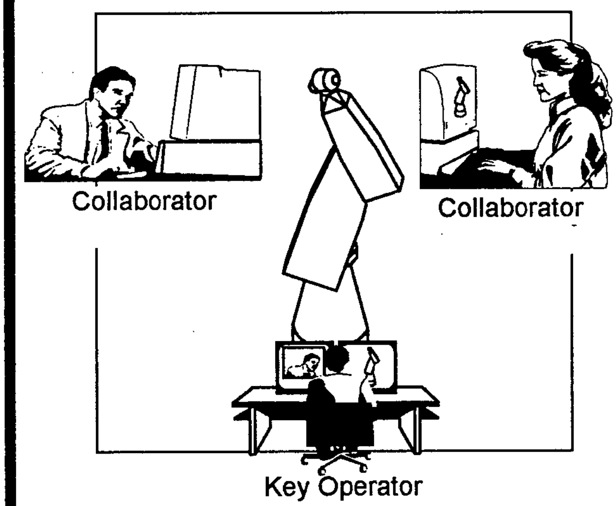

Figure 1: Collaborators in VCC System.

Three key experiments were performed [7]. Each followed a task framework shown in Figure 2. A robot system was put in operation. New information was uncovered and requirements for a new task were given to a collaborator. While the key operator continued to work on other tasks, the collaborator developed a plan for the newly dispatched task and, on completion, transmitted a high-level solution back to the key operator. At the next opportunity, the key operator used the graphical programming system to automatically generate and test a detailed robot program and, when appropriate, run it on the robot. In the first set of experiments, a collaborator used Envision to generate general excavation tasks for large debris-free areas. In the second, Penn. State's Virtual Tools software [8] with a point-and-direct interface [9] was used for large debris excavation and grasping tasks. In the third, structured lighting and video data were used for controlled depth excavation.

The first experiment set had the following steps: (1) A minimal model (without buried debris) of the tank environment was read into a collaborator's copy of Envision. (2) The collaborator developed a collection of generalized excavation tasks and transmitted them to the key operator who (3) chose, adjusted (*e.g.,* set excavation depth), tested and approved excavation tasks as needed.

In one run of these first experiments, 4 plans were developed and forty excavation operations were performed. Here, it took approximately 1 hour to develop each general plan and 8 minutes to use a plan to generate and execute detailed robot programs. Equipment time totaled 5.3 hours and the machine was kept moving (i.e., utilized) 40% of that time. Had collaboration not been applied, utilization would have been 10-20%.

The second experiment set, with Virtual Tools, had the following steps: (1) Using two computer-controlled cameras, the collaborator captured video image pairs of partially buried debris objects. (2) Utilizing the Virtual Tools point-and-direct interface, the collaborator accurately specified the task and parameters including the excavation tool's path or gripper's grasp point and then (3) generated and transmitted a parameterized task to the key operator. (4) The key operator then planned, tested, and executed detailed robot motion plans to accomplish the excavation or grasping task.

A series of 36 Virtual Tools experiments with 4 novice test subjects was performed. Collaboration increased robot utilization, on average, to 33%, a factor of 3 over comparable non-VCC rates.

The structured lighting programming experiments had the following steps: (1) A collaborator commanded a structured lighting system to scan and build a Envision model

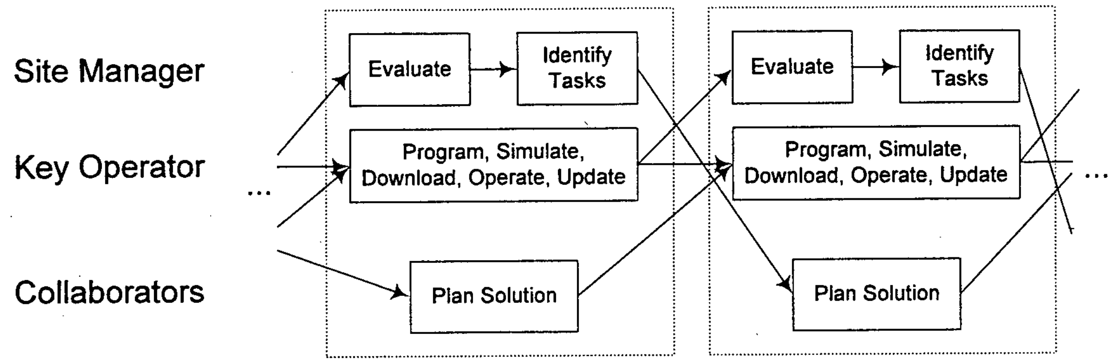

Figure 2) Work flow in a Virtual Collaborative Control system.

of the waste near the debris and commanded a calibrated video system to digitize a single image of the scene. (2) The collaborator then generated a basic path by drawing (with a mouse) a desired path on the video image, (3) used custom software to project the 2D drawing onto a 3D path on the polygonal model, (4) manually edited and tested the path in Envision for optimal motions and orientations, and (5) submitted the plan to the key operator. (6) The key operator planned, tested, and executed detailed motion plans as above.

Collaboration improved operational consistency and utilization rates when structured lighting was needed. Due to surface complexity variations and manual editing complexity, path creation and testing took between 2 and 11 minutes. When collaboration was not used, time variability and difficulty limited the technology's usefulness and hence the technical viability of controlled depth excavation. Here, collaboration remains the only practical means of controlled depth excavation.

CASE STUDY 4: ROBOTIC SYSTEM COMMUNICATIONS DEVELOPMENT

In developing intelligent machine systems, Sandia engineers in Albuquerque have developed considerable expertise in internet-compatible intelligent machine command and control architectures [10]. Recent availability of high-quality communications tools, like the Common Object Request Broker Architecture (CORBA), RTI's Network Data Delivery Service (NDDS), and Microsoft's ActiveX control, have heightened the interest in networking robotic systems.

While these commercial tools provide significantly higher functionality and ease of use than low-level implementations, subsystem developers must still agree on the command sets the applications will use. Furthermore, because tools like CORBA primarily support database-type applications, additional protocol issues must be addressed to meet machine control requirements. For example, the protocols must let the supervisory control software stop or modify a machine motion before the machine completes another motion.

In 1996, Sandia ordered two robots with Sandia-specified CORBA-based communications interfaces from PaR Systems Inc., of Shoreview, MN. Driven by schedule constraints, a new working relationship was established for this project. In earlier efforts, suppliers delivered complete subsystems that Sandia later integrated with their supervisory software. In this effort, Sandia and PaR began integration two months prior to hardware shipment. Here, the internet was used to connect Sandia's supervisory code to PaR's controller to let the engineers test and debug the interfaces prior to system shipment. This small change transformed the Sandia and PaR relationship from simply customers and suppliers into a problem-solving team.

The result was dramatic. Because the engineers began testing two months before product shipment, key integration milestones were completed before the robot subsystems were fully installed at Sandia. (Due to this fact, the integrated software system was often used to test hardware subsystems as they came on line. This testing, then, eliminated the need to develop additional custom test software that would otherwise be required.) Moreover, because problems were discovered earlier, time spent fixing bugs was further reduced. Finally, whereas the traditional approach would have required extensive travel, controller development-related travel was reduced to one kick-off meeting and a final training and evaluation meeting. The result was that for this effort, software integration time was not an issue.

CONCLUSIONS

This paper presented four case studies in internet collaboration and outlined two new areas of development that have resulted from performing the experiments.

The unique aspect of Case Study 1 is the way the problem, developing an appropriate assembly process, was solved. In contrast to a conventional focus on products, in this case the focus was on knowledge, problem solving, and teaming. The lead time between technology development and application was minimized. The problem owner remained focused on the problem while the technology provider remained focused on providing the analysis.

Prior to development of VCE, engineers would have great difficulty discussing problems like those described in Case 2. Conversely, the VCE feature allowed Sandia and NIST engineers to rapidly identify, discuss, and correct problems that had previously been very difficult to resolve. The result was more effective reviews producing higher quality results.

In the Case 3 VCC testbed experiments, three key issues were demonstrated. First, because all collaborator task plans were automatically tested before being run on the robots, system safety and reliability were retained. Second, because the key operator could operate the robot while others were performing the time-consuming planning tasks, utilization rates were improved. Third, because the collaborators were not responsible for all system operations, training was simplified and specialization was possible.

In Case 4, it was shown that geographic and institutional separation of customers and suppliers need not restrict the creation of teaming environments. By working together, the engineers were able to test one-another's software at each early critical juncture and thereby achieve rapid progress toward their goals.

Moving beyond these cases, we see that the solution of specific problems is less important than the development of new methods for working across traditionally separate engineering disciplines. It is becoming clear in many of the applications we deal with that a key role of technology is in enabling collaboration. As problems become more complex, a variety of domain area experts are needed, and software that facilitates communication across disciplines (such as research and product design) serves to rapidly raise the level of interaction to the expert level in each domain. Put bluntly, except for *shrink wrapped* software, using the net as merely a software delivery vehicle is insufficient. People need to team to solve problems. The net connects the people to make the teams work.

There are many serious implications of such *borderless* teaming of research and commercial experts. First, there is real money at stake. Reducing the time it takes to develop product innovations provides a significant competitive edge, which will likely create huge pressures to accelerate the move to technology-based collaboration. But whither the traditional research institution? What is the continued purpose of brick and mortar facilities when a thin fiber connects experts anywhere in the world? How do you protect intellectual property in a borderless collaborative environment? All of these questions will be brought into sharp focus by the growth of internet-based teaming. We suspect that the world is not really ready for such borderless teaming, but given the large potential economic impact, it is going to happen; and probably faster than most people anticipate. It should make for interesting times.

BIBLIOGRAPHY

[1] Jones, Wilson and Calton, *Constraint-based Interactive Assembly Planning*, **1997 IEEE International Conference on Robotics and Automation**, Albuquerque, NM, April 1997.

[2] Kaufman, Wilson, Jones, Calton, and Ames, *The Archimedes 2 Mechanical Assembly Planning System*, **IEEE Intl. Conf. on Robotics and Automation**, 1996, pp. 3361-8.

[3] **New Computer Technology Dissolves Limitations to Traditional Design and Engineering Methods**, Sandia National Laboratories News Release, Chris Miller (cmiller@sandia.gov), September 6, 1996 (also at http://www.sandia.gov/media/vce.htm).

[4] Small, D. E., & M. J. McDonald *Graphical Programming of Telerobotic Tasks*, **ANS 7th Topical Meeting on Robotics and Remote Systems**, Augusta GA, April 27 - May 1, 1997.

[5] McDonald, M. J. & L. G. Ice, *Rocinante: A Virtual Collaborative Visualizer*, **ANS 7th Topical Meeting on Robotics and Remote Systems**, Augusta GA, April 27 - May 1, 1997.

[6] Demonstrated at the 7th **Department of Energy Forum on Robotics for Environmental Remediation**, Albuquerque, NM, July 25, 1996.

[7] McDonald, M. J., D. E. Small, C. C. Graves, D. J. Cannon, *Virtual Collaborative Control to Improve Intelligent Robotic System Efficiency and Quality,* **1997 IEEE International Conference on Robotics and Automation**, Albuquerque, NM, April 1997.

[8] Wang, C. and D. Cannon. 1996. *Virtual Reality Based Point-and-Direct Telerobotic Inspection.* Submitted to **IEEE Transactions on Robotics and Automation Journal** 12 (4): 516-531.

[9] Cannon, D.J. **Point-And -Direct Telerobotics: Object Level Strategic Supervisory Control in Unstructured Interactive Human-Machine System Environments,** A dissertation of the Stanford Unviersity in Mechanical Engineering, June, 1992.

[10] Griesmeyer, J. M., M. J. McDonald, R. W. Harrigan, P. L. Butler and J. B. Rigdon, **Generic Intelligent System Control (GISC)**, Sandia Internal Report, SAND9202159, 10/92.

M98000390

Report Number (14) SAND--97-2454C
CONF-980514--

Publ. Date (11) 199803
Sponsor Code (18) DOE/CR, XF
JC Category (19) UC-900, DOE/ER

DOE