Proceedings of the 2007 Winter Simulation Conference
S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, eds.

PANEL: DISTRIBUTED SIMULATION IN INDUSTRY - A REAL-WORLD NECESSITY OR IVORY TOWER FANCY?

Peter Lendermann

D-SIMLAB Technologies Ptd. Ltd.
30 Biopolis Street, Matrix #03-11
Singapore 138671, SINGAPORE

Matthias U. Heinicke

Unigraphics Solutions GmbH
Ingersheimer Strasse 12
70499 Stuttgart, GERMANY.

Leon F. McGinnis

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205, U.S.A

Charles McLean

Manufacturing Systems Integration Division
National Institute of Standards & Technology
Gaithersburg, ML 20899-8260, U.S.A.

Steffen Straßburger

Technical University of Ilmenau
Helmholtzplatz 3
98693 Ilmenau, GERMANY

Simon J.E. Taylor

Centre for Applied Simulation Modelling
Brunel University
Uxbridge, Middx UB8 3PH, UNITED KINGDOM

ABSTRACT

Distributed simulation has a long history at the Winter Simulation Conference. Although successful in the military domain it appears, however, that the idea of applying distributed simulation in other fields for modeling and analysis of large-scale, heterogeneous systems such as communication networks or supply chains has still not reached the stage of commercial use until today. This panel attempts to identify reasons for this phenomenon by debating whether distributed simulation is actually a real-world necessity or should rather be considered ivory tower fancy.

1 INTRODUCTION

Distributed (or "federated") simulation refers to technologies that enable a simulation program to execute on a computing system containing multiple processors that are interconnected by a communication network in the hope of either decreasing the time taken to run the simulation and/or to support model reuse by linking together separately developed simulations to form a larger one (Fujimoto 2000). It was originally motivated by needs in the military domain for more effective means to train personnel in distributed virtual environments that mimic actual combat situations (Fujimoto 1998). Subsequently, the availability of synchronization middleware such as the Runtime Infrastructure of the High Level Architecture (Kuhl et al. 1999) has also inspired research looking at potential application of distributed simulation for modeling and analysis of large-scale, heterogeneous systems such as communication networks or supply chains.

The fundamental question of applicability of distributed simulation in these areas has already been previously addressed at the Winter Simulation Conference, for example in Lendermann (2006). In this paper it was concluded that the number of application scenarios for distributed simulation to resolve real-world manufacturing and logistics challenges is actually quite limited. In particular, even though it has been discussed most (with regard to technical feasibility) in the literature, the across-echelon supply chain management scenario was found to be the most unrealistic one due to some inherent limitations associated with it. Only in few domains such as semiconductor manufacturing promising realistic application scenarios can actually be thought of. But before reasonable simulation execution times become feasible additional research issues are yet to be resolved.

In this setting, it appears that the simulation community as a whole is also still far from a unanimous viewpoint with regard to the applicability of distributed simulation. This panel attempts to highlight some of the relevant issues by debating conflicting viewpoints as follows:

Three panelists are making the case in favor of distributed simulation by explaining for what kind of real-world applications this technology is indispensable, why this is the case, and what has to be done to make it happen. In turn, two panelist are presenting their doubts with regard to having distributed simulation applications that add real value and therefore make decision-makers in industry pay for it.

2 POSITION STATEMENT BY STEFFEN STRASSBURGER

Parallel and distributed simulation has been a research topic within the simulation community for many years. It should be assumed that it had achieved a wide-spread practical application by now. However, this is certainly not the case. The traditional objective of parallel simulation was to gain speed-up – an objective which can only be reached in certain constellations and with quite a bit of effort in the modeling process.

Therefore the author argues that this traditional objective is nowadays outdated, simply because of a lack of major applications which would justify the effort of parallelizing the simulation model. Some exceptions may apply for some rather rare high performance simulations like climate and weather prognostics.

The new objective for distributed simulation is to provide and facilitate interoperability between and reusability of heterogeneous simulation systems. This objective is supported by the advent of the High Level Architecture. HLA provides for the first time a real industry standard which provides interoperability for a wide range of simulation systems and applications.

The range of applications which can benefit from this standard is quite large – nevertheless HLA is far from ever becoming a mainstream technology in the civilian simulation community (Straßburger 2006).

A general prerequisite for a successful and economically reasonable application of distributed simulation is

1. the existence of complex simulation models,
2. the existence of a problem or question which can only be solved if the models are combined.

There may be other cases where the application of HLA may be reasonable, but the best business case can be drawn from the possibility of combining existing models to investigate their interactions.

An important application scenario where this is useful comes from the digital factory area. Here, often separate simulation models of different sections of the production process exist (e.g. assembly, body-in-white, paint shop,…). The interdependencies between these separate sections of the production facility are often neglected, although they are of major importance for the functionality of the overall system.

In order to harmonize the production across all sections and to investigate existing dependencies (e.g. in the treatment of rework, buffer sizes, transport strategies, scheduling strategies, shift regimes etc.) distributed simulation is the suggested tool of choice and the perfect setting to demonstrate the advantages of a standard like HLA.

Several pilot projects to investigate the benefits have been conducted by DaimlerChrysler (Straßburger et al. 2003) and other producers of commercial vehicles. They provide promising results in terms of economical benefits drawn from the experiments as well as acceptable performance.

The subject of one investigation was the dimensioning of buffer sizes between the paint shop and several assembly lines prior and after the paint shop. A distributed simulation of the existing models was able to show the effects of using the correct buffer size and as well as implications of using harmonized shift regimes in the different sections of the factory. Traditional and individual stand-alone simulations of the sections of the factory could not have provided these results.

A further important future application of distributed simulation is in the area of worker training. By combining production simulations and interactive virtual environments it will become possible to insert a worker into a virtual factory model and train him towards future work responsibilities (Straßburger et al. 2005).

The major requirements to enable distributed simulation based on HLA to become a commodity technology which is simply used when needed are the following:

- Standardization of use patterns for the usage of HLA in COTS simulation packages (Taylor et al. 2006)
- Integration of HLA support into COTS simulation packages by the tool vendors (HLA has to become a standard interface just like today's ODBC or Socket Interface)
- Efficient synchronization algorithms which provide a good performance while not burdening the modeler with any extra-effort

In an ideal world, i.e. when all these requirements are fulfilled, it will be possible to combine any desired simulation package with one another to solve a certain problem. Only then it may be economically worthwhile for today's standard user to use distributed simulation in day-to-day business.

3 POSITION STATEMENT BY LEON McGINNIS

I have focused my comments on "federated" rather than "distributed" to rule out those situations in which multiple copies of the same simulation model are run simultaneously on different CPUs. This latter application is quite viable, and not that uncommon today. I would rather address the "other" meaning of "distributed simulation," in which a single model is modularized, and the modules executed simultaneously on different CPUs.

The notion of federated manufacturing simulation, with federates running on different computers, synchronized in some way to achieve temporally correct interactions, is appealing because it offers the promise of faster simulations, as well as the potential benefits from modularization. A number of researchers have recognized that appeal, and Google Scholar returns over 32,000 hits for the search "manufacturing.and.simulation.and.(parallel.or.distributed.or.federated" excluding publications prior to 1999. Many of these hits reference papers at the Winter Simulation Conference. Closer investigation seems to indicate a limited number of sustained efforts, however, as might be indicated by a series of publications over an extended time. Moreover, it is difficult to uncover successful examples of distributed simulation actually deployed and used in any ongoing fashion.

My own dalliance with distributed simulation has been focused primarily on semiconductor manufacturing and supply chains, and has extended over a period of roughly a decade. My group has attempted to develop a start-from-scratch distributed simulation implemented in Java using "plain vanilla" HLA (Park et al. 2001); we have developed some capability with AutoMod/ASAP as a benchmark (McGinnis 2004), we have looked at alternative federating approaches (McGinnis et al. 2005), and we have explored various mechanisms to speed up distributed fab simulations (Wang et al. 2005), and (Xu and McGinnis 2006). We have collaborated with others (Lendermann et al 2001), (Lendermann et al. 2003), (Lendermann et al. 2004). In short, we have considerable experience with distributed simulation, at least in the context of semiconductor manufacturing.

Based on that experience, it is my considered opinion that it is extremely unlikely that federated simulation in manufacturing will be a commercially viable activity within the foreseeable future.

The reasons seem pretty obvious:

1. There is no theoretical basis for knowing a priori that a particular problem is even amenable to effective (run time reducing) federated simulation. Our experience with manufacturing is admittedly limited to highly automated manufacturing, where a complex automated material handling system implements material flow, and must be coordinated with production operations to achieve desired schedule performance. Our intuition is that the large disparity in "event density" between the AMHS and production operations reduces straightforward synchronization via HLA to a sequential computation performed on multiple computers rather than a single computer. A moment's reflection indicates that this is probably not a good idea.
2. For a candidate application, achieving any parallelism in simulation computation requires carefully designed and implemented federates, and the methodological foundation for this is not yet fully articulated. Again, based on our experience, the only way to achieve true parallelism in the simulation is employ modeling and computational "tricks" to ameliorate the problems caused by differences in event densities. This also seems to argue for a diminished potential for federating "legacy" simulations.
3. The critical hurdles are modeling, rather than software per se, so it is difficult to identify a value proposition for the sellers of simulation software. Given a system as complex as a semiconductor wafer fab, there are a number of "obvious" ways to partition the model into federates, some straightforward and easy, and some rather subtle. It's not clear at the outset which of these approaches will perform best, and so at this point, it appears that experience and judgment are critical in developing good federated models. Given the limited number of application examples in the literature, this is not a "leverage point" for federated manufacturing simulations.
4. There are no "existence proofs" for the value of federated simulation in manufacturing, i.e., no published case study where a user can point to the payoff from using federated simulations; hence it is very difficult to pose a convincing value proposition for the prospective users of federated simulations in manufacturing. My, hopefully informed, opinion is that neither prospective users of federated manufacturing simulation, nor commercial simulation software vendors are much interested in funding research in this area, because both communities perceive that the cost and risk are substantial and the value is arguable, due to the continuing rapid evolution of compute power.

To summarize: We do not have a solid theoretical foundation for federated simulation in manufacturing; to make it work—at least in systems with automated material handling—seems to require specific modeling expertise; that expertise appears to be extremely limited; and there is not a compelling value proposition for either the suppliers or the users of manufacturing simulation software. The conclusion then, is that the necessary theory, methods, and software will never be developed, at least not in a commercially available form.

4 POSITION STATEMENT BY CHARLES McLEAN

For what kind of applications is distributed simulation indispensable? Simulation-based training systems for emergency responders is one such application area. Emergency response organizations need to be better prepared to deal with both man-made and natural disasters. The responses to the attacks on the World Trade Center and Hurricane Katrina are strong evidence of this need. Effective emergency response presents a number of challenges to the respective authorities. First responders and incident management personnel need better planning and training resources to prepare for future incidents. One major challenge is the lack of time and opportunities to train the emergency responders and decision makers to handle emergencies. Another challenge is the variety of different types of disaster scenarios that must be dealt with. Yet another is the complexity of organizations and systems affected by and involved in responding to disasters, see the National Incident Management System (NIMS 2004) and the National Response Plan (NRP 2006).

Live training exercises while valuable are often very expensive to organize and conduct. The limitations of live exercises could be overcome through the use of integrated modular simulations that model the major phenomena and incident response operations associated with a disaster. Planning and training systems that are based upon simulation technology could help to prepare for a more diverse range of scenarios than live exercises. These systems could also support individual, team, or multi-organizational planning and training needs at lower cost. Distributed simulation will be an absolute must to support preparations for disaster incident management and emergency response operations.

Distributed simulation could help address many of the challenges that we face today. Why build distributed rather than monolithic simulations? A distributed approach could enable the integration of modules created by different developers and enhance overall functionality of planning and training systems. For example, distributed simulations could be used to:

- enable parallel, modular development of specialized simulation system components by independent software developers with different areas of expertise,
- allow the configuration of integrated simulations that meet specific regional or scenario-based needs,
- model multiple organizations where some of the information about the inner workings of each organization may be hidden from other participants for reasons such as security or proprietary issues,
- simulate multiple levels of organizations and systems at different degrees of resolution such that lower level simulations generate information that feeds into higher levels,
- model multiple systems with different simulation requirements where an individual simulation-vendor's products does not provide the capabilities to model all areas of interest,
- allow software developers to hide the internal workings of a simulation system through the creation of run-time simulators with limited functionality,
- create an array of low-cost, run-time, simulation models that can be integrated into larger models,
- take advantage of additional computing power, specific operating systems, or peripheral devices (e.g., virtual reality interfaces) afforded by distributing across multiple computer processors,
- provide simultaneous access to executing simulation models for users in different locations (collaborative work environments),
- offer different types and numbers of software licenses for different functions supporting simulation activities (model building, visualization, execution, analysis).

The behavioral phenomena, and response organization simulations may be large, complex, and expensive to model. For example, some of the different types of simulators that may be included in emergency response planning and training simulations are:

- *Social behavior* – models the collective social behavior of multiple individuals including crowds, traffic, epidemics, and consumer behavior.
- *Physical* – models the physical phenomena involved in the creation and growth of the emergency incident including earthquakes, explosions, fires, chemical, biological, or radiological plumes.
- *Environmental* - models the environmental phenomena that may affect the growth or containment of the emergency incident, its impact on the population or on the efforts by responding agencies including weather, watershed systems, indoor climate, and ecology.
- *Organizational* – models the actions of the organizations involved in any aspect associated with the incident including fire departments, law enforcement, health care, other government agencies, and even terrorist organizations.
- *Infrastructure systems* - models the behavior of the infrastructure systems following the occurrence of an emergency incident including the propagation of the impact of damage throughout systems such as power distribution, water and food supply, computer and communications networks.

A survey (Jain and McLean 2003) indicates that a number of modeling and simulation applications for analyzing various disaster events already exist.

What has to be done to make it happen? A coordinated effort by developers, government agencies, and standards organizations will be required to achieve the vision of interoperable simulation-based planning and training systems. More effort will need to be put into the following areas:

- Development of technically correct models, behaviors, and data for various phenomena that affect training, mission planning, and operational support.
- Use of gaming technology to provide an immersive, engaging graphical, audio, and haptic (force feedback) environment that offers high quality realistic experiences.
- Enhancement of distributed simulation mechanisms that enable time synchronization, data sharing, checkpointing, time warp, rollback, replay, and logging functions between various simulation and gaming applications.
- Implementation of mechanisms that allow for centralized distribution and management of updates to software and data sets.
- Establishment of security features that prevent unauthorized access to, or modification of, computer systems, software, and data.

Effective, technically sound, and commercially-available data standards will also be needed. Examples of data types that will need to be standardized:

- *Incident management structure:* organizations, roles, responsibilities, policies, plans, procedures, actions, records, resource allocations, checklists,
- *Response resources*: organizations, equipment, systems, vehicles, people, evacuation centers, supplies, contact points, data, capabilities, resource capacity, status,
- *Infrastructure systems*: transportation (roads, trains, buses, trucks), telecommunications, power, gas, water, food, healthcare, sewage, alerting systems, status, sensitive targets,
- *Spatial data*: maps, terrain, regions, areas, and building layouts and models,
- *Hazard effects:* chemical, biological, nuclear, fire, severe weather, other natural and man-made disasters, plume models, flooding, health,
- *Incident events:* chronologies, timing, descriptions, victims, damage assessments, and other status data,
- *Responder computing and communications:* radio and other equipment, channel assignments, switching systems, transmission towers, areas of coverage, message formats,
- *Population and demographics:* location, age, sex, other attributes by time of day,
- *Weather and environmental* – wind speed, air temperature, precipitation,
- *Financial:* cost of operations, consumables, leased equipment, labor.

Although a number of standards development efforts are underway, much more needs to be done. Standards will need to be harmonized. Validation, verification, and testing capabilities will need to be established as well to ensure correctness and interoperability of simulations.

5 POSITION STATEMENT BY MATTHIAS HEINICKE

Over the past ten years, there have been efforts aimed at distributed simulation of manufacturing systems, but these efforts have had minimal vendor involvement. This stands in strong contrast to the situation with other simulation related technologies, such as 3D, or SQL integration or ODBC integration or SAP integration, where there has been a very strong "customer pull" for innovation. In general manufacturing customers are not clamoring for a distributed simulation capability. The vendor perspective is that customers might think distributed simulation would be a nice feature, provided somebody else would pay for and develop the capability.

There have been a number of university based projects that have examined distributed simulation for manufacturing applications, in Japan, Germany, Singapore, and the US. However, none of these projects has yet led to a commercially available distributed simulation capability.

There has been important progress in establishing standards for distributed simulation infrastructure, particularly the High-Level Architecture and the variants that have appeared in recent years. This progress is driven largely by defense industry applications, not by manufacturing applications, which remain the largest market segment for discrete event simulation software vendors.

There are three distinct "flavors" of distributed simulation:

1. Distribution of one simulation over different computers,
2. Integration of multiple vendor packages using distributed simulation infrastructure (e.g., discrete event simulation and PLC),
3. Cloning simulation on multiple machines for DoE.

For Category 3, there is at least some potential market, and providing this kind of simulation capability does not require significant changes to existing simulation software. Rather, achieving type 1 distributed simulation is mostly a matter of selecting an appropriate operating system, and finding the software that will manage a grid-based computation.

Category 2 distributed simulation, the integration of multiple simulation and perhaps control system applications, is interesting, but remains a significant technical challenge. To date, there do not appear to be many customers willing to underwrite the development of these solutions.

Category 1 distributed simulation has yet to demonstrate a compelling value proposition for manufacturing oriented users. In other words, the promise of reduced run times from distributed simulation of a single simulation simply has not been convincingly demonstrated. Thus, there is little justification for investing the resources necessary to develop the required theory, methods, and tools required to make this a commercially viable software offering.

If we look at the manufacturing oriented distributed simulation projects that have been described in the literature, we find that they are

- small scale,
- specialized or requiring custom code versus integration of legacy simulations,
- developed to explore data structures.

These efforts collectively do not provide the necessary intellectual foundation upon which to build a successful commercial offering.

If we look at what has been accomplished so far in the manufacturing-oriented distributed simulations described in the literature, we find that:

- The level of effort required is high.
- Payoff for distributed simulations of type (1) is low; for (2) is not achieved; for (3) is high.

For simulation vendors, this reinforces the impression that any efforts devoted to distributed simulation should be focused on grid computing applications, where a single simulation model is replicated many different times on different computers to support an experimental design.

Given these observations, from the perspective of simulation software vendors, we can answer the question, "Will we see commercial distributed simulation offerings?"

- For type (1) and (2) applications, no.
- For (3) yes, in fact, the capability is already available.

One last observation that seems important is that continuing advancement of computing speed and memory space also appears to reduce the justification for distributed simulation in manufacturing.

6 POSITION STATEMENT BY SIMON TAYLOR

For what kind of applications is distributed simulation indispensable? Why this is the case? What has to be done to make it happen? To answer these questions let us take an alternative look at distributed simulation. In 2003 I "challenged" Professor Stewart Robinson, a well-known simulation modelling researcher and practitioner, to give his views on distributed simulation. He presented these as the keynote speaker at the 2003 DS-RT Symposium in Delft and then published them in the paper *Distributed Simulation and Simulation Practice* (Robinson 2005). In this article he identifies three modes of simulation practice. Mode 1 *Simulation as Software Engineering*, Mode 2 *Simulation as a Process of Organisational Change* and Mode 3 *Simulation as Facilitation*. Briefly, Mode 1 concerns the development of large-scale models that are used and reused over a long period of time, such as those found in military and public policy sectors, Mode 2 represents relatively small scale models used over a months or weeks while a problem is being investigated and are usually found in industry and business, and Mode 3 again is typically used in industry and business and represents models that usually exist only to "prove a point", i.e. a model is used to bring together different stakeholders to establish a common view (or at least some form of understanding). Widening the scope of distributed simulation as defined above, he then considers the nature of potential applications and classifies them as being demand-led or technology-led. Demand-led (D) indicates that, in Robinson's opinion, there is a clear user demand for the application while technology-led (T) indicates that the application is primarily the focus of technology researchers (i.e. a solution looking for a problem). In the original paper blanks were used to indicate no clear application; in this panel paper "N" will be used to indicate this. Table 1 shows Robinson's table.

Table 1: Demand-led vs Technology-led Applications of Distributed Simulation (after Robinson 2005)

Category	Application	Simulation as		
		Software Engineering	Process of Organizational Change	Facilitation
Model execution	Distributing model execution	D	T (D)	N
	Linking separate models	D	T (D)	N
Data management	Linking to databases or other software	D	D	N
	Linking to real-time systems	D	T (D)	N
Experimentation	Gaming	D	T (D)	T (D)
	Distributing multiple replications	D	D	N
	Distributing multiple scenarios	D	D	N
Project Processes	Sharing models	D	D	D
	Application sharing	D	D	D
	Virtual meetings	D	D	D
	Searching for model components	D	D	D

Although this is largely based on opinion rather than a literature review or survey, the table does serve to illustrate two interesting observations on distributed simulation (and reflects, to some extent the simulation research priorities surveyed in Taylor and Robinson, 2006). The first is *breadth*. It is sometimes refreshing for those active in a field to see how others see their efforts. Researchers in distributed simulation often restrict their own field to the first category in the table, *model execution*, with the applications *distributing model execution* and *linking separate models*. However, it is clear from the table that a broader definition of distributed simulation as being a mixture of simulation and distributed computing techniques reveals three other categories and eight applications. The second observation is *demand*. As indicated above, in each Mode of simulation practice our applications are either *demand-led*, *technology-led* or no clear application. In Mode 1, all the applications of distributed simulation are considered to be demand-led. In Mode 2, all are demand-led apart from *distributed model execution, linking separate models*, *linking to real-time systems* and *gaming*. In Mode 3 all *project processes* applications are demand-led, *gaming* is technology-led and other applications are considered to be not really applicable.

However, relatively recent experience might question some of Robinson's "T" classifications. Work done by researchers involved in the COTS Simulation Package Interoperability Product Development Group (CSPI PDG) (Taylor et al. 2006) have identified several applications where *model execution* could be demand-led. For example, on-going work in automobile manufacturing, semiconductor manufacturing and heath care have identified demand for both *distributed model execution* due to the size of a model and *linking separate models* in cases where models have been previously developed. Outside of the CSPI PDG there have been instances where simulations have been developed to link to real-time systems to prototype new control algorithms. For both Mode 2 and 3, there is also evidence for *Gaming* to be demand-led in the sense of training scenarios deployed over the world wide web. These "updates" to Robinson's table are indicated by D in brackets. To answer the first question, "for what kind of applications distributed simulation is indispensable?" one might therefore take the broad view of many!

Let us now consider why this is the case. In Model 1 simulations, there is a demand for all categories of distributed simulation. The reason for this is the size and scope of the simulations that typically for into this type of simulation. For example, in military simulations there is evidence of a demand for simulations that are large, that are reused, that link to different data or real-time sources, that demand large amounts of experimentation and involve many people. The evidence for this can be found in the many papers of the various Simulation Interoperability Workshops of the Simulation Interoperability Standards Organization. Mode 2 simulations tend to be smaller and involve less people. However, there is a small demand for model execution and real-time linking, and then larger demand for data management, experimentation and support for project processes. For Mode 3, the picture is more restrictive as there is a small demand for gaming but a larger demand for project process support. Evidence for these observations can be found in papers from the Winter Simulation Conferences.

What has to be done to make it happen? This is a very interesting question, especially in the light of my expanded definition of distributed simulation. Given the existence of standards such as the HLA, in Mode 1 simulations one might take the view that for the most part demand is being satisfied. These simulations tend to be directly programmed using conventional software engineering techniques and tools. These simulations also tend to have relatively simple time management needs as they are either real-time or time-stepped. Effective solutions to distributed model execution can be therefore more easily developed. Direct access to code means that data management is relatively easily handled as is the development of distributed support for experimentation. Software engineers tend to be well versed in groupware needed for the support of project processes. An important point is that the developers of Mode 1 simulations tend to have the skills available to make our list of distributed simulation applications possible.

Mode 2 simulations present a different problem. Developers of Mode 2 simulations tend to be operational researchers trained in operational research and not software engineering. Such developers also tend to use tools that are familiar to the participants of the Winter Simulation Conference. These *COTS Simulation Packages* (CSP) (Arena, Simul8, Witness, etc.) are productivity tools that distance operational researchers from low-level coding needs. Mode 2 simulations tend to be smaller than Mode 1 simulations but, arguably, require more replications and experimentation. Another difference to Mode 1 simulations is that any Mode 2 distributed simulation applications that require solutions to be developed (usually) needs significant involvement from the CSP vendor. CSP vendors, in turn, need a strong reason to invest in the development of a solution and there is a clear relationship between the complexity of solution versus its realisation. With this in mind, let us consider each application category. The small demand of *Model execution* and the complexity of realisation (primarily due to time management requirements) mean that without a generalised "off-the-shelf" solution, it continues to be unlikely for such applications to become mainstream. The exception to this is the development of standards to facilitate distributed model execution solutions (Taylor et al. 2006). However, there is still much work to be done. Data management is relatively straightforward and many CSPs contain the functionality to link to different data sources. Experimentation, however, is unusual. In my experience there is a clear need for distributed experimentation and replication for some Mode 2 simulations. However, with the exceptions of systems such as WINGRID (Mustafee et al. 2006) and some limited support claimed by few packages, there are few practical solutions to this clear need. Finally, there is also clear need for support for project processes which can be easily fulfilled by contemporary groupware. A similar comment can be made for Mode 3 simulations.

So, to sum up my viewpoint, distributed simulation as a field will benefit from a wider definition of the applications that distributed simulation techniques can give. There is clear demand for this scope of application that is currently being satisfied in large scale simulations. Where simulation is used as a facilitation technique, there is also demand for distributed simulation support for project processes which is also being fulfilled. The largest need/solution gap, however, is in the area where simulation is used to investigate organisational change. In this area, although there is a clear need for distributed simulation applications, a combination of an understandable skills gap of simulation developers, the need for software vendors to widely invest in new features and the potential complexity of solution means that progress in distributed model execution and experimentation is understandably slow. To make significant progress in this area, in partnership with industry, relevant research action is needed that takes care to understand the *actual* problems encountered in Mode 2 simulations. With this, researchers in this area can provide a vital "bridging" function between developers and vendors. Unfortunately, with certain exceptions, current research in this area fails to understand key real-world problems. This problem urgently needs to be addressed for solutions to this demand to be created.

7 SUMMARY AND CONCLUSIONS

The views presented in this panel illustrate the complexity associated with the applicability of distributed simulation.

As suggested by Simon Taylor, distributed simulation as a field can benefit from a wider definition of the applications that distributed simulation techniques can give rather than just using it for investigation of organizational and operational changes. And the High Level Architecture as an interoperability standard helps fulfill one major prerequisite for successful realization of distributed simulation as pointed out by Steffen Straßburger.

But the concerns raised by Matthias Heinicke and Leon McGinnis carry heavy weight: A really compelling, commercially sound proof-of-concept, not to mention a solution for federated simulation is still not available, even though substantial R&D work has been carried out also with companies such as Chartered Semiconductor Manufacturing (refer again to Lendermann et al. 2004) and Daimler Chrysler (refer again to Straßburger et al. 2005). New application scenarios such as the one outlined by Charles McLean could possibly make a contribution towards resolving this challenge.

REFERENCES

Fujimoto, R.M. 1998. Parallel and Distributed Simulation. In *Handbook of Simulation: Principles, Methodology, Advance, and Practice*. Ed. J. Banks. John Wiley and Sons Inc. New York NY, USA. pp. 429-464

Fujimoto, R.M. 2000. *Parallel and Distributed Simulation Systems*. John Wiley and Sons Inc. New York NY, USA.

Jain, S. and C.R. McLean. 2003. Modeling and Simulation of Emergency Response: Workshop Report, Relevant Standards and Tools. *National Institute of Standards and Technology Internal Report*, NISTIR-7071. Available via <http://www.nist.gov/msidlibrary/doc/nistir7071.pdf>

Kuhl, F., R. Weatherly, and J. Dahmann. 1999. *Creating Computer Simulation Systems: An Introduction to the High Level Architecture.* Prentice Hall PTR.

Lendermann, P., B.P. Gan, and L.F. McGinnis. 2001. Distributed Simulation with Incorporated APS Procedures for High-Fidelity Supply Chain Optimization. *Proceedings of the 2001 Winter Simulation Conference*. Washington DC, USA. December 9-12, 2001. pp. 1138-1145.

Lendermann, P., N. Julka, B.P. Gan, D. Chen, L.F. McGinnis, and J.P. McGinnis. 2003. Distributed Supply Chain Simulation as a Decision Support Tool for the Semiconductor Industry. *SIMULATION: Transactions of the Society for Modeling and Simulation International.* 79(3). pp.126-138.

Lendermann, P., B.P. Gan, Y.L. Loh, H.K. Tan, S.K. Lieu, L.F. McGinnis, and John W. Fowler. 2004. Analysis of a Borderless Fab Scenario in a Distributed Simulation Testbed. *Proceedings of the 2004 Winter Simulation Conference*. Washington DC, USA. December 5-8, 2004. pp.1896-1901.

Lendermann, P. 2006. About the Need for Distributed Simulation Technology for the Resolution of Real-World Manufacturing and Logistics Problems. *Proceedings of the 2006 Winter Simulation Conference*. Monterey CA, USA. December 3-6, 2006. pp.1119-1128.

McGinnis, L. F. (2004). Distributing a Large-Scale, Complex Fab Simulation Using HLA and Java: Issues and Lessons. *Proceedings of the 2004 Winter Simulation Conference*. Washington DC, USA. December 5-8, 2004. pp.1891-1895.

McGinnis, L. F., S. Xu and K. Wang (2005). Evaluate Simulation Design Alternatives for Large Scale Manufacturing Systems. *Proceedings of the 2005*

Winter Simulation Conference. Orlando FL, USA. December 4-7, 2005. pp.344-347.

Mustafee, N., Alstad, A., Larsen, B., Taylor, S.J.E. and Ladbrook, J. (2006). Grid-Enabling FIRST: Speeding Up Simulation Applications Using WinGrid. *Proceedings of the Tenth IEEE International Symposium on Distributed Simulation and Real-Time Applications*. IEEE Computer Society. pp. 157-164.

National Incident Management System (NIMS 2004). http://www.fema.gov/emergency/nims/index.shtm

National Response Plan (NRP 2006). http://www.dhs.gov/xprepresp/committees/editorial_0566.shtm

Park, J., S. A. Reveliotis, D. A. Bodner, C. Zhou, J. Wu and L. F. McGinnis. 2001. High-Fidelity Rapid Prototyping of 300mm Fabs through Discrete Event System Modeling. *Computers in Industry* 45(1). pp.79-98.

Robinson, S. (2005). Distributed Simulation and Simulation Practice. *SIMULATION: Transactions of the Society for Modeling and Simulation International.* 81(1). pp.5-13.

Straßburger, S. 2006. Overview about the High Level Architecture for Modelling and Simulation and Recent Developments. *Simulation News Europe*. Vol. 16/2. pp. 5-14.

Straßburger, S., G. Schmidgall and S. Haasis. 2003. Distributed Manufacturing Simulation as an Enabling Technology for the Digital Factory. *Journal of Advanced Manufacturing Systems (JAMS)*. Vol. 2/1. pp.111-126.

Straßburger, S., T. Schulze, M. Lemessi, and G. Rehn. 2005. Temporally Parallel Coupling of Discrete Simulation Systems with Virtual Reality Systems. *Proceedings of the 2005 Winter Simulation Conference*. Orlando, USA. December 4-7, 2005. pp. 1949-1957.

Taylor, S., S. Turner, M. Low, X. Wang, S. Straßburger, J. Ladbrook. 2006. Developing Interoperability Standards for Distributed Simulation and COTS Simulation Packages with the CSPI PDG. *Proceedings of the 2006 Winter Simulation Conference*. Monterey, CA, USA, December 3-6, 2006. pp. 1001-1110.

Taylor, S.J.E. and Robinson, S. 2006. So where to next? A survey of the future for discrete-event simulation. *Journal of Simulation*, 1, 1. pp. 1-6.

Xu, S., and L. F. McGinnis. 2006. Optimistic-Conservative Synchronization in Distributed Factory Simulation. *Proceedings of the 2006 Winter Simulation Conference*. Monterey CA, USA. December 3-6, 2006. pp.1069-1074.

Wang, K., S. Xu, and L.F. McGinnis. 2005. Time Management in Distributed Factory Simulation, a Case Study Using HLA. *Proceedings of the 2005 Winter Simulation Conference*. Orlando FL, USA. December 4-7, 2005. pp.394-401.

AUTHOR BIOGRAPHIES

PETER LENDERMANN is the Co-Founder and CEO of D-SIMLAB Technologies, a Singapore-based company providing simulation-based decision support solutions and services to Aerospace, Semiconductor Manufacturing and other asset-intensive industries. He has been engaged in the simulation community since the early 1990's when he worked in a multinational research collaboration at the European Laboratory for Particle Physics CERN (Geneva, Switzerland) and Nagoya University (Japan). In 1996 he joined a German consulting firm where he was responsible for business process re-engineering projects with numerous process manufacturing, aviation and automotive clients in Europe, Canada and China. In 2000 he joined the Singapore Institute of Manufacturing Technology where he led the simulation-related research activities until spinning them off into D-SIMLAB Technologies. Peter holds a PhD in Applied High-Energy Physics from Humboldt-University in Berlin (Germany) and an MBA in International Economics and Management from SDA Bocconi in Milan (Italy). He is also an Adjunct Associate Professor at the Department of Industrial and Systems Engineering at the National University of Singapore. His email address is <peter@d-simlab.com>.

MATTHIAS U. HEINICKE works in the Marketing Manufacturing Business Group of UGS in Stuttgart. He received a B.S. degree in economic engineering from the University of Kaiserslautern. He has over ten years of experience in discrete event simulation using SIMPLE++, eM-Plant and Plant Simulation. During his career he has trained more then 100 engineers in creating efficient and accurate material flow simulation models. He has also lectured as an assistant professor in the advanced technical college of Esslingen (Germany). His email address is <matthias.heinicke@siemens.com>.

LEON F. McGINNIS is Eugene C. Gwaltney Professor of Manufacturing Systems at Georgia Institute of Technology, where he also serves as Associate Director of the Manufacturing Research Center founding Director of the Keck Virtual Factory Lab and Director of the Product and Systems Lifecycle Management Center. His research focuses on the application of operations research and computer science to solve decision problems arising in the design and operation of industrial logistics systems. His email address is <leon.mcginnis@isye.gatech.edu>.

CHARLES McLEAN is is a computer scientist and Group Leader of the Manufacturing Simulation and Modeling Group at the National Institute of Standards and Technology. He has managed research programs in manufacturing simulation, engineering tool integration, product data standards, and manufacturing automation at NIST since 1982. He has authored more than 50 technical papers on topics in these areas. He is on the Executive Board of the Winter Simulation Conference and the Editorial Board of the International Journal of Production, Planning, and Control. He is formerly the Vice Chairman of the International Federation of Information Processing (IFIP) Working Group on Production Management Systems (WG 5.7). He holds an M.S. in Information Engineering from University of Illinois at Chicago and a B.A. from Cornell University. His e-mail address is <mclean@cme.nist.gov>.

STEFFEN STRASSBURGER is a professor at the Technical University of Ilmenau in the School of Economic Sciences. In previous positions he was working as head of the "Virtual Development" department at the Fraunhofer Institute in Magdeburg, Germany and as researcher at the DaimlerChrysler Research Center in Ulm, Germany. He holds a Ph.D. and a Master's degree in Computer Science from the University of Magdeburg, Germany. His international experience includes a one-year stay at the University of Wisconsin, Stevens Point and a stay at the Georgia Institute of Technology, Atlanta. His research interests include the topics simulation and distributed simulation as well as general interoperability topics within the digital factory context. He is also the Vice Chair of SISO's COTS Simulation Package Interoperability Product Development Group. His email address is <Steffen.Strassburger@tu-ilmenau.de>.

SIMON J.E. TAYLOR is a Senior Lecturer in the Department of Information Systems and Computing and is a member of the Centre for Applied Simulation Modeling, both at Brunel University, UK. He is also a Visiting Associate Professor at the Parallel and Distributed Computing Centre at Nanyang Technological University in Singapore. He is also Information Director of ACM SIGSIM, ACM SIGSIM PADS Liaison Officer and Chair of the Simulation Study Group of the UK Operational Research Society. He is a steering committee member of PADS and general co-chair of the UK Simulation Workshop Series. His main research interests are distributed simulation and applications of ICT to simulation modeling. His email address is <simon.taylor@brunel.ac.uk>.