**Nationaal Lucht- en Ruimtevaartlaboratorium**
National Aerospace Laboratory NLR

NLR TP 97256

# High performance computing in simulation that fulfils user-driven quality criteria

W. Loeve, E.H. Baalbergen, M.E.S. Vogels, H. van der Ven.

# DOCUMENT CONTROL SHEET

| | | | |
|---|---|---|---|
| | ORIGINATOR'S REF.<br>NLR TP 97256 U | | SECURITY CLASS.<br>Unclassified |

ORIGINATOR
National Aerospace Laboratory NLR, Amsterdam, The Netherlands

TITLE
High Performance computing in simulation that fulfils user-driven quality criteria.

PRESENTED AT
Conference DASIA 97 organised by Eurospace in Sevilla, Spain,
26-29 May 1997.

| | | | |
|---|---|---|---|
| AUTHORS<br>W. Loeve, E.H. Baalbergen, M.E.S. Vogels and H. van der Ven. | DATE<br>970507 | pp<br>10 | ref<br>8 |

DESCRIPTORS
Applications programs (computers)
Client server systems
Computer networks
Configuration management
Graphical user interface
Management information systems
Multidisciplinary design optimization
Navier-Stokes equation
Quality control
Software tools
Supercomputers
User requirements
World wide web

ABSTRACT
Simulation is part of engineering processes. Considerations in the present paper concern simulation as a stand-alone process, with (on application level) no interrupts from the simulation environment. Simulation environments consist of computer systems, software, data and supporting documents. Accessibility of the simulation environments for engineers and performance of the computers determine simulation efficiency. Engineers enter and leave enterprises. Continuity of simulation know-how and tools is a concern of enterprise management. Controllability of simulation is of interest for both management and engineers. To realise continuity and controllability, configuration management and control is required for simulation environments as well as for simulations. Ideas of engineers and management about how simulation shall be supported by simulation environments are quality criteria for construction of these environments. It is described in the paper how off-the-shelf products can be used for realisation of simulation environments fulfilling user-driven quality criteria.

# HIGH PERFORMANCE COMPUTING IN SIMULATION THAT FULFILS USER-DRIVEN QUALITY CRITERIA

W. Loeve, E.H. Baalbergen, M.E.S. Vogels, H. van der Ven.
National Aerospace Laboratory, The Netherlands
Anthony Fokkerweg 2, 1059 CM Amsterdam, The Netherlands,
phone: +31 (0)20 5113421, fax: +31 (0)20 5113210, E-mail: loeve@nlr.nl

## ABSTRACT

Simulation is part of engineering processes. Considerations in the present paper concern simulation as a stand-alone process, with (on application level) no interrupts from the simulation environment. Simulation environments consist of computer systems, software, data and supporting documents. Accessibility of the simulation environments for engineers and performance of the computers determine simulation efficiency. Engineers enter and leave enterprises. Continuity of simulation know-how and tools is a concern of enterprise management. Controllability of simulation is of interest for both management and engineers. To realise continuity and controllability, configuration management and control is required for simulation environments as well as for simulations. Ideas of engineers and management about how simulation shall be supported by simulation environments are quality criteria for construction of these environments. It is described in the paper how off-the-shelf products can be used for realisation of simulation environments fulfilling user-driven quality criteria.

## 1. INTRODUCTION

Simulation is applied in engineering processes in industry to generate information about products under development. In the present paper, the starting point is simulation based on mathematical models. Examples of models are aerodynamic models that describe flows around objects and models that describe the dynamic behaviour of manipulators. The models are discretised for numerical processing in the computer. The engineer who wants to execute a specific simulation defines the input for the simulation process such as for flow simulation the geometry, the discretised grid and the flow conditions. Sometimes also parameters for the numerical processing have to be defined by the user of the simulation.

Simulation not only requires software for numerical processing. Also software tools are required for generating the input for the simulation and for analysis of the results. These comprise tools for visualisation. Manuals are required for support of the user of the simulation software. It is preferred to have manuals available on-line in electronic form for easy access and easy updating. Workstations are required for the interaction of the engineer with the software tools. In many cases, such as for computational fluid dynamics, workstations do not offer the required performance [Borland 1994; Vogels 1995, 1996]. In those cases remote computers with more performance can be used. On-line storage of input data for simulations and output data is preferable to enable engineers to verify their own results.

The combined set of software, manuals, computers and data is called the simulation environment. Engineers have ideas about how access should be organised to the software, manuals, local workstation, remote computers and data in the simulation environment. For management it is possible to look at the generation of information by simulation as a production process. As a consequence the ISO 9001 industry standard can be used as a guideline for quality control of simulation. Quality control aims at improvement of simulation with respect to cost, reliability and required time.

Quality control of simulation means realising continuity of simulation expertise, the possibility to reconstruct results from previous simulations and registration of errors and changes in the simulation environment and problems that engineers have encountered in using the simulation environment. Managers would like also to support quality control by tools in the simulation environment.

The ideas of engineers and management about how they like simulation to be made possible by the simulation environment are quality criteria for the construction of this environment. Based on this the quality criteria for structuring and using simulation environments are discussed. It is described how off-the-shelf products can be applied and combined in such a way that they become useful for realisation of simulation environments that fulfil the quality criteria.

## 2. QUALITY MANAGEMENT

Introduction of quality management in the production of simulation results requires the definition of limits for time, cost and reliability for the production. Competition leads to the requirement that in course of time the limits can be set sharper. The way to do this is to improve the expertise of people involved, to improve the simulation environment, and to improve the way the simulation environment is used.
One way to improve the expertise of people involved in simulation is to document help information at different levels of knowledge about the concerning simulation principles and simulation environment. The concerning documents have to be updated regularly and have to be made easily accessible. This leads to the requirement to make documents part of the simulation environment {req. 1}.
Continuously improving the simulation environment leads to the requirement that the structure of simulation environments allows easy replacement of elements if new developments make improvement of the environment financially and technically possible {req. 2}.

The users of simulation environments have the tendency to implicitly use the applicability of a single personal computer with an advanced graphical user interface as reference for their judgement of the applicability of simulation environments. This quality criterion for simulation environments is difficult to meet in case visualisation, high performance computing, handling of large amounts of data and exchange of simulation related information between co-operating specialists or organisations are required. A single workstation or personal computer in those cases is not sufficient. In those cases various computing platforms have to be used. A quality requirement is that multi-platform simulation environments present itself to the user as one single computer {req. 3}.

Timeliness of simulation results leads to the need to define error reporting, testability and mean-time-to-repair for elements in the simulation environment and for the environment as a whole. As a consequence it is required that the original suppliers of software elements can be kept responsible for introducing error reporting in their software, and for designing their software for testability and maintainability {req. 4}. This is to safeguard quick corrective actions if required.

Timeliness ofsimulation results also leads to the need to be able to restore former situations in case of disturbances {req. 5}.

It is required that the simulation environment enables the user to verify his simulation by comparison with results obtained earlier, in order to ensure reliability of simulation results {req. 6}.

## 3. SIMULATION ENVIRONMENTS

In many simulations the performance of the available computers always has been a bottleneck. This is so for aerodynamics, strength of complex structures, combustion, chemistry and electromagnetics for instance. It is even more so for integrated design of complex products such as aircraft. This is the reason that in many organisations and in national and super national policy making, attention is paid to ensure access to high performance computing such as vector computing and parallel computing.

For simulation not only high performance compute servers are required. There is also a need for graphical presentation and processing with the help of workstations and for information management by means of specific servers. The first step towards integration of the resources into one virtual computer is to integrate the systems in a network. The capacity of the network has to be sufficient for the often large information flows in high performance computing and networking.

The dominating challenge in constructing simulation environments is {req. 3}: to make it possible that the multi-platform simulation environment presents itself to each user as one single computer for use of information in three forms: data, software and documents, and for execution of simulations on a heterogeneous collection of hardware and system software.

The following issues have to be addressed in simulation environments:

- Software heterogeneity. Programs come from different sources (e.g., public domain, commercial, developed in-house), and hence usually lack uniformity with respect to user interface and data interface.

- Programs run on different, possibly heterogeneous computers, where the heterogeneity may concern hardware as well as operating system. As a result, the user of the computer network has to be shielded from such issues as:
  - Finding out on which computer a specific program may run in the network.
  - Remote login and remote execution, facing the user with aspects such as authentication and accounting.
  - Operation of different operating systems or variants of the "same" operating system, commonly with a variety of possibly incompatible utilities.
  - Exchange of files between computers.
  - Hardware heterogeneity. Availability of different computers may give rise to incompatible data formats due to different approaches with respect to byte ordering and representation of real numbers. Consequently, the user may have to perform conversions when transferring binary information to another computer.
- Users typically get faced with organising, managing, and finding a way through an amount of online information that is much larger than that on a single computer.
- Realisation of one virtual computer in a heterogeneous computer network requires co-operation of all control facilities among the computers involved.
- Configuration management and configuration control have to be applied to software, data and documents in the co-operative work that is supported by the heterogeneous HPCN environment.
- Reliability of simulation results is possible via continuous verification of results. Part of this has to be done by comparison of results with other results and analysis of not well understood differences. To do this, the configurations of all versions of the simulation environment and input of all simulations have to be registered.

All of the above mentioned issues are dealt with in SPINE (a product of NLR and NEC). With SPINE, the network acts as one single computer, a so-called metacomputer. SPINE is described in chapter 4. Applications of SPINE are described in chapter 5

## 4. SPINE SIMULATION ENVIRONMENTS

SPINE is a system that supports the development and operational use of general as well as application-specific working environments in computer networks. A SPINE-based working environment provides its end user with access to and efficient use of the resources - computing power, data storage, I/O facilities, software - available from the network, as if the resources are located on one single, "virtual" computer, a metacomputer (cf. req. 3). The user is confronted with the single computer model, and not with a computer network that reveals its particularities. Details emerging from the use of multiple systems, networking, and distributed computing are dealt with in SPINE, and are transparent to the user.

SPINE supports the development of metacomputers in that it provides facilities to realise application-specific working environments, which are open (in that existing software can easily be integrated; see req. 1), extendible, portable, and easy to customise to the specific needs of a user or group of users.

Support for the operational use comprises technical solutions for the key elements of a metacomputer: an easy-to-use desk top, facilities for the management of information, and middleware.

The desk top provides the end user of a SPINE-based working environment with a GUI, a graphical integrated shell that facilitates operation of the metacomputer in a user and applications oriented fashion. Most of the common data management and processing operations are available in terms of point-and-click and drag-and-drop operations on windows and icons. This mode of user interaction has become an important aspect of user-oriented computing, as can be seen from the popularity of the Windows operating systems for PCs. In addition to the Windows look-and-feel, the SPINE desk top offers capabilities for data-flow and work-flow driven computing.

Management of information is an important aspect in today's computing systems. On the one hand users may drown in the enormous amount of (mostly unrelated but sometimes interrelated) information. On the other hand, since today's (meta)computers become more and more used by multi-disciplinary development teams, access to information needs to be carefully controlled. SPINE supports these aspects of information management by providing tools for managing storage, access to, and modifications of data, documents (cf. req. 2), software and any information related to this (cf. req. 4, 5, 6), and for the exchange of information among users.

The SPINE middleware serves to “glue” the computing and information resources available from the network together into one coherent system that plays the role of operating system for the metacomputer. Its major task is to manage the available resources in order to accomplish the single computer look-and-feel, and to exploit the potentials of the underlying computer network.

SPINE consists of the following subsytems:

- The *User Shell,* which provides all user interface facilities
- The *Environment Management System,* which provides means for integration of available information (software, documents and data) into the working environment, for management of working environments, and for processing and network services.
- The *Information Management System,* which services for management (including storage, access, version control, conversions, exchange) of all information available from the network.
- The *Common Tool Set*, which provides a set of general-purpose tools, and interfaces to commercial packages, that will be available for the working environments. The Common Tool Set is considered to be part of the Information Management system.

In figure 1 the set up of a SPINE simulation working environment is illustrated.

To maintain portability, flexibility, and vendor-independence, and to avoid costs for third-party licenses, one of the major constraints applicable to the development of SPINE has been to utilise as many standard facilities and public-domain (in particular, freeware) as possible, whenever applicable. Examples are:

- rsh (remote shell), NFS (Network File System), and FTP (File Transfer Protocol), for networking;
- RCS (Revision Control System) for version control of files;
- Tcl/Tk for its own graphical user interface (desk top);
- Mosaic and HTTPD for Internet and Web applications;
- Ghostview for browsing documents prepared using text processing packages that yet do not support HTML

Native (commercial) packages may be used (i.e., plugged in) instead, when desired. For example, Netscape is often used as Web browser instead of Mosaic.

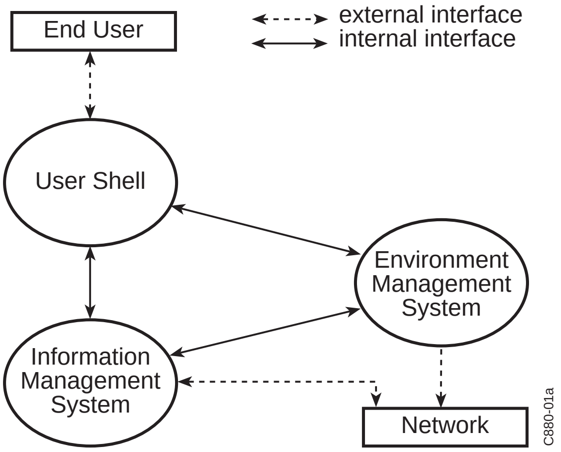

*Figure 1. A SPINE simulation environment and external and internal interfaces involved*

NEC has planned to market SPINE in the near future. SPINE has been used since 1993 to develop several working environments. It is currently being used - and planned to be used - in several national and international projects.

At the moment, several systems exist that support realization of, mostly only part of the aspects of, metacomputing. For example, the Parallel Simulation User Environment (PSUE), aims to provide a software platform that can be customised for particular simulations. However, PSUE provides a particular desk-top model, which leaves hardly any freedom with respect to customisation, and which enforces a tool-oriented mode of working, whereas the SPINE desk- top model supports data-oriented, tool-oriented, as well as work-flow driven modes. Another example is SiFrame, an open working environment that aims at supporting design-process management in a multi-user environment. SiFrame, however, concentrates mainly on the work-flow driven way of system interaction.

Most developments in metacomputing are primarily driven by research in computer science, which has resulted in a gap between scientifically well- founded metacomputing solutions and industrial needs. This gap often leads to products that do not meet the demands from engineers. SPINE, although including results of scientific research, has its roots in a practical engineering environment, ISNaS [Vogels 1989], has proved to be the solution that meets the demands in modern engineering.

## 5. SPECIFIC SIMULATION ENVIRONMENTS

Since computers were introduced at NLR in the late fifties this institute has applied a centralised IT approach. The main reasons were economy of scale and the need to be able to procure the computing power that is required for the most demanding applications [Loeve 1976].

When we look at the most powerful compute servers installed in the world [Dongarra, 1996] we see, at the top of the list, massively parallel computers with many scalar processors and distributed memory and vector computers with several processors and shared memory. The vector computers are easier to use than the distributed memory computers, especially with computational mechanics software that is frequently used in engineering [Vogels 1995].

The most advanced industrially applicable CFD software for aerodynamic analysis in aircraft engineering at the moment is based on the Reynolds-Averaged Navier-Stokes equations [Rubbert, 1994]. One not too detailed computation for a full aircraft configuration requires about 5 hours in case of a sustained computer performance of about 500MFLOP/s. One of the interesting applications of these computations for the near future is support of aerodynamic optimisation [Borland, 1994]. Optimisation requires more computer performance than analysis, because for optimisation for instance about 40 times solution of the flow equations in combination with an optimisation algorithm is required. With a NEC SX-4/16 (a vector computer with 16 processors and shared memory) it is possible to realise for CFD a computational performance on 14 processors of 8,500MFLOP/s. This is realised with relatively little effort: only DO-loop parallelisation (Ven 1997). One aerodynamic optimisation of 40 analysis runs then takes about 10 hours. Based on this reasoning NLR decided to procure a NEC SX-4/16 for R&D and for support of engineering in The Netherlands [Loeve 1996, Hameetman 1997].

Since the first supercomputer was installed at NLR in 1987 it was decided also to make use of a tape robot integrated with a data management server for management of data, software and electronic documents. It was considered to be essential to realise similar access to the computer facilities for all engineers and scientists in both NLR establishments. The NLR establishments are at a distance of 100 km from each other.

For both NLR establishments it was made possible to integrate the supercomputer and information server with local workstations in such a way that the combination was presented to the user as one single virtual computer. The first application of the integration was realised for CFD. The resulting working environment was called Information System for flow simulation based on the Navier-Stokes equations (ISNaS). ISNaS contained the SPINE features described above. Together with the architecture and construction principles for CFD software [Vogels 1997], ISNaS was the basis for quality management of development and application of CFD software for industry by the Informatics Division of NLR.

*ISNaS*

At NLR the working environment ISNaS is being used to support the development and use of a simulation environment for fluid flows. Concurrent development of computational fluid dynamics (CFD) software is supported by the software repository tool of SPINE and complemented by a parallel software development toolpack tuned to the NEC SX-4/16. This toolpack contains tools for compilation, execution, analysis, optimisation, and debugging. Maintenance of the software is supported by automatic generation of makefiles and obligatory regression tests to ensure preservation of past functionality of the code. This working environment part is only accessible to developers.

A forms based set of tools is used to registrate submitting and processing of error reports, change proposals and problem reports concerning the various aspects of the simulation environment.

Use of ISNaS is supported by the same working environment. The complete simulation process from grid generation up to visualisation of the flow results is integrated.

For ISNaS, development and production are concurrent processes, so it is obligatory that the functionality of the simulation environment does not degrade. This requirement is fulfilled by the working environment. ISNaS is used in a national project in the Netherlands to make a national science network to become the computer for CFD.

*MDO*

Demanding engineering activities in which simulation plays an important role are Multi-disciplinary Design and Optimisation (MDO) processes of products with great added value such as aeroplanes. In addition to simulation in separate mono-disciplines awareness of each discipline is required that interactions with other disciplines have to be realised in such a way that the team can steer the work. Especially in MDO applications the easy and simultaneous processing of distributed information is required as well as a reliable seamless flow of information through the successive steps of simulation processes. The complexity of the heterogeneous HPCN computer networks that are applied in MDO of products with great added value and of the collections of software systems implemented on the networks increases continuously. In the MDO project partly financed by the EU [MDO Consortium], an IT framework is developed based on SPINE.

For optimal use of the network as if it is one computer for MDO, it is desirable to structure the information in accordance with the MDO process. Structuring the MDO process has to be such that each discipline recognises their own contribution to the process and the interaction via information exchange with other disciplines. So called $N^2$ diagrams can be used to make visible what simulation processes are executed, from what process input is required and for what process the output has to serve as input. Also iteration loops in the process can be made visible via this way of process modelling. It appears that even more than in a mono-disciplinary simulation application, the structuring of the information (software, data and documents) in the "virtual" computer has to be a reflection of the simulation process. It is a prerequisite for access to the system for each contributing specialist based on look-and-feel.

## 6. CONCLUSION

The present paper describes how the intranet of an organisation or a national network can become the computer for engineers and scientists.

## 7. REFERENCES

Borland, C.J., Benton, J.R., Frank, P.D., Kao, T.J., Mastro, R.A. and Barthelemy, J.F.M. (1994) Multidisciplinary design optimization of a Commercial Aircraft Wing - An Exploratory Study. 5th Symposium on Multidisciplinary Analysis and Optimization, Panama City Beach, Sept. 7-9, 1994. AIAA-94-4305-CP, pp 505-519.

Dongarra, J.J., Meuer, H.W. and Strohmaier, E (1996) TOP500 Supercomputer Sites November 1996; retrievable from internet by anonymous ftp to www.netlib.org/benchmark/top500.ps.

Hameetman, G.J., Loeve, W., Poppinga, G., Potma, K. (1997) Early bench mark results on the NEC SX-4 supercomputer; in *Parallel Computational Fluid Dynamics (Algorithms and results using advanced computers)* Elsevier.

Loeve, W. (1976) A hierarchical network linking two research laboratories; Computer Networks 1 North-Holland Publishing Company.

Loeve, W. (1996) From parallel CFD to a tool for computer aided engineering, in *Parallel Computational Fluid dynamics* Elsevier.

MDO Consortium (1996) http://www.telecall.co.uk/~srcbae/mdo

Rubbert, P.E. (1994) CFD and the Changing World of Airplane Design; AIAA Wright Brothers Lecture, Anaheim California; September.

Ven, van der H and Vegt, van der J.J.W. (1997) Experience with advanced CFD algorithms on NEC SX-4, in *VECPAR '96 selected papers from the 2nd International Meeting on Vector and Parallel Processing* Springer.

Vogels, M.E.S. and Loeve, W. (1989) Development of ISNaS: An Information System for Flow Simulation in Design, in *Computer Applications in Production and engineering* North Holland.

Vogels, M.E.S., Ven van der H., Hameetman, G.J. (1995) The need for supercomputers in aerospace research and industry, in *Proceedings HPCN Europe '95*

Vogels, M.E.S. (1997) Architecture and construction principles for computational fluid dynamics software for engineering in industry; Thesis, ISBN 90-9010295; M.E.S. Vogels, NLR