Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-99227

Information and Communication Technology for industrial design using parallel CFD

M.E.S. Vogels

Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-99227

Information and Communication Technology for industrial design using parallel CFD

M.E.S. Vogels

This report is based on a presentation held at the 4th ECCOMAS CFD Congress, Athens, Greece, 7-11 September, 1998.

Division: Information and Communication Technology
Issued: June 1999
Classification of title: unclassified

Summary

The awareness of the role of Computational Fluid Dynamics (CFD) in the entire aircraft development process in industry has shifted the Information and Communication Technology (ICT) focus from "parallelising the compute intensive element in CFD analysis" to a wider view. The wider view includes

1. speeding up the entire CFD analysis, and not just the compute intensive part
2. performing trade-off studies across disciplines and
3. managing the industry's competence including the industry's competence on parallel CFD.

Based on various projects on Dutch and European scales, different aspects from the contribution of Information and Communication Technology are highlighted.

Contents

(17 pages in total)

1 Introduction

Since the potential of the impact of Computational Fluid Dynamics (CFD) on aircraft design has been established, CFD is considered to be part of an aircraft industry's business process. And as the industry continuously strives to improve its business process, and measures each local improvement on its result for the complete business process, the same applies to improving CFD and application of CFD. In this sense, parallel CFD is seen as a means to an end, not as an end in itself. As a result improving CFD tools is no longer restricted to making the CFD codes run faster

In the paper, examples of Information and Communication Technology contributions to (1) CFD analysis and (2) Multi-discipline analysis, design, and optimisation (MDO) are described from the user's point of view. The enterprise's point of view is highlighted in a separate section.

2 ICT for industrial design using parallel CFD

2.1 CFD analysis

2.1.1 Parallelisation

In the NICE initiative[1] an industrial multi-block Navier-Stokes flow solver is parallelised. The purpose of the parallelisation is not just to reduce the execution time, but also to reduce the use of computing resources (CPU-time, memory) expressed in execution cost (Ref. 1).

The parallelisation strategy is to distribute the blocks, as available in the grid, over the processors. The classification of the group of representative applications is given in Figure 1.

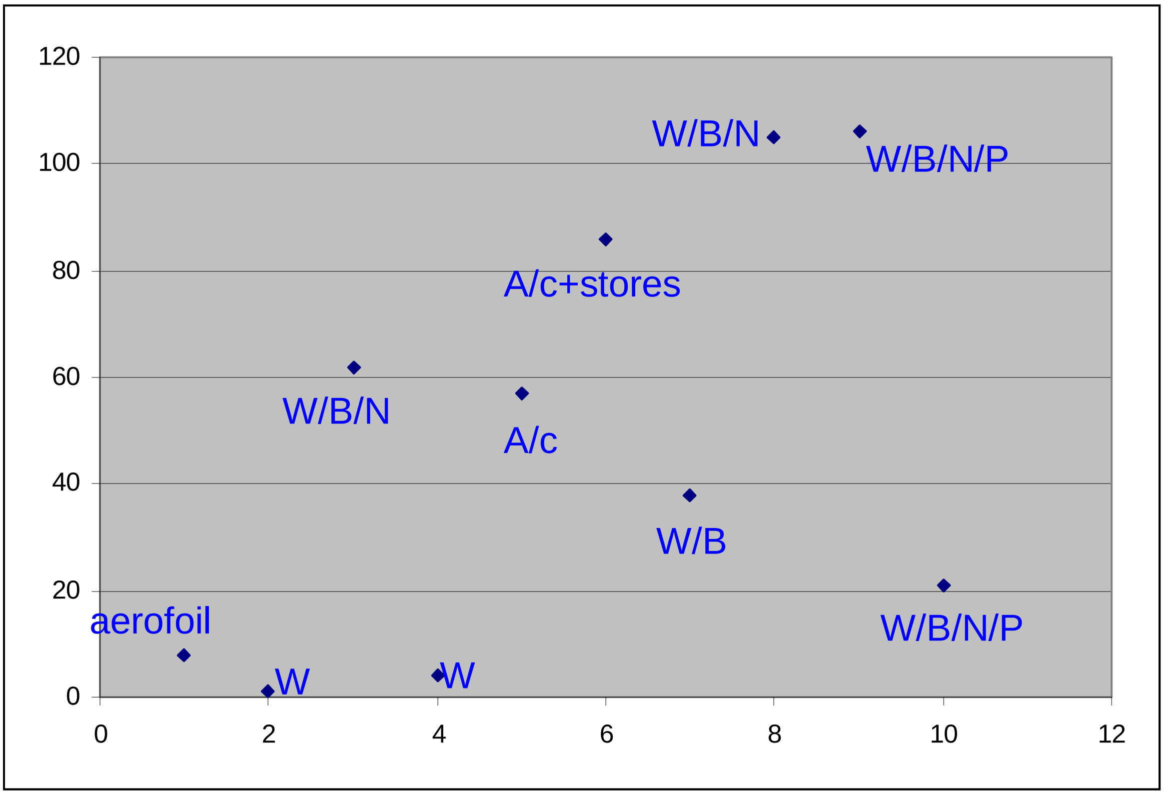

Fig. 1 Configurations and numbers of blocks for the representative applications (W=wing, W/B=wing-body, W/B/N=wing-body-nacelle, W/B/N/P=wing-body-nacelle-pylon

[1] The NICE (Netherlands Initiative on Computational Fluid Dynamics in Engineering with HPCN) is a collaboration between the National Aerospace Laboratory NLR, Delft Hydraulics, the Maritime Institute MARIN, and the research school for CFD J.M. Burgers' Centre. The project is managed by NLR, and is partly funded by the Dutch foundation HPCN.

The parallelisation on the shared memory parallel vector machine NEC SX-4 is for the moderate number of 8-16 processors with each processor a vector processor with 2 GFlop/s peak performance. The assignment of grid blocks to processors is based on the work load in the grid blocks measured during a first iteration.

It is clear that a parallel execution takes more CPU time and more memory compared to a sequential execution. Nevertheless, resource use for a parallel execution may be less than for a sequential execution because the larger memory is occupied for shorter time. In the practice of Reference 1, the resource use was reduced for all representative applications except for a 2D case and a single block case (see Figure 2). The achieved speed-ups on 8 processors are given in Figure 3.

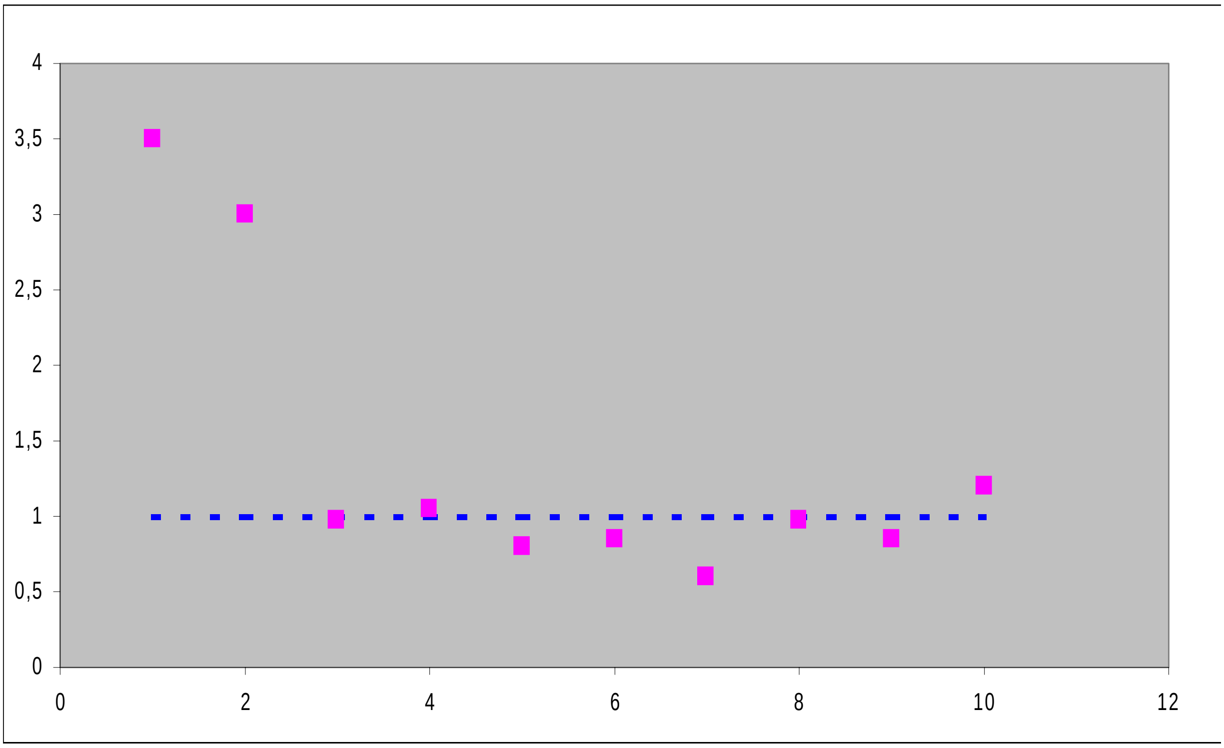

Fig. 2 Ratio of resource use during parallel execution on 8 processors and resource use of sequential execution, for the 10 representative applications. (The dashed line shows the equal use of resources.)

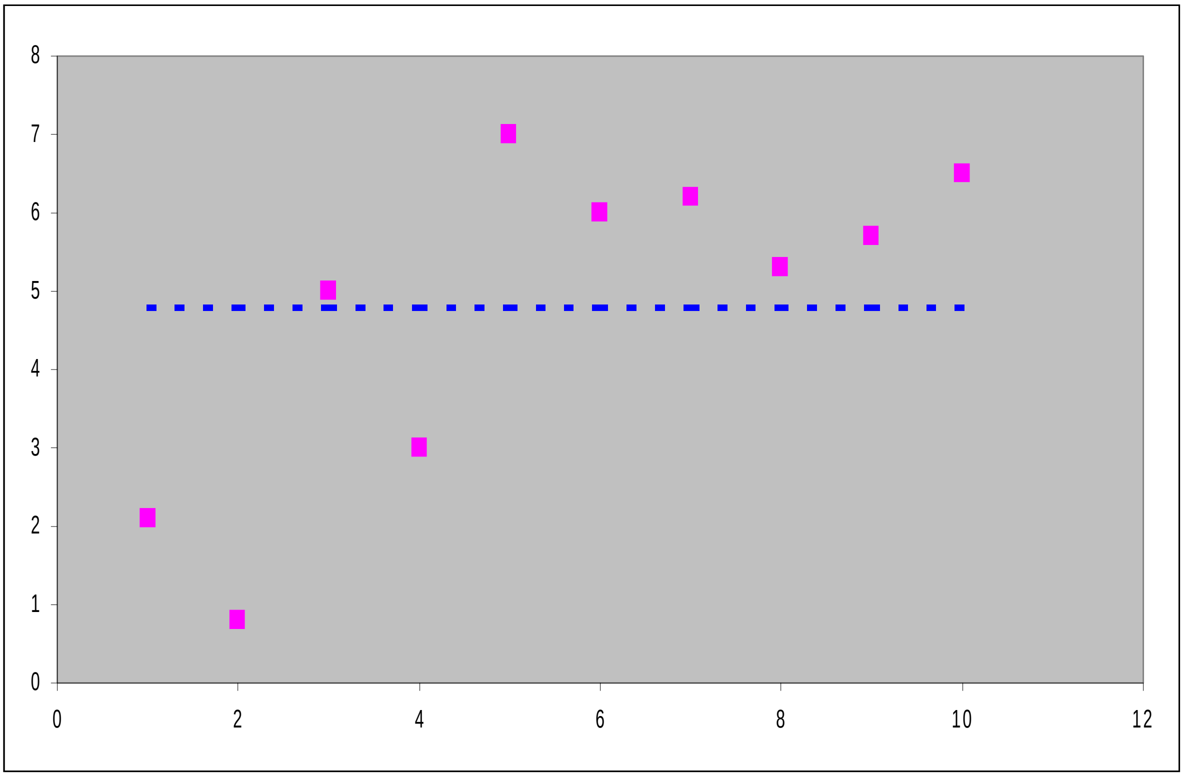

Fig. 3 Speed-ups for the 10 representative applications on 8 processors (The dashed line give the required overall speed-up.)

During the execution of the parallelisation work, it proved to be beneficial for the parallelisation specialist to run the benchmark cases with the parallel code integrated into a CFD system together with tools for task estimation, task allocation, and speed-up estimation. In the CFD system, built with the system integration tool SPINEware (Ref. 4), the heterogeneous computing network is presented to the user as a single computer. Distributed processes in the network are connected; operations in the system are invoked by simple drag-and-drop actions.
Now that the parallelisation work is finished, the same way of operating is available for other users of the multi-block Navier-Stokes flow solver.

The example shows that Information and Communication Technology today involves not only parallelising a CFD solver for shorter execution times, but also

- parallelising for lower resource use, and
- providing the parallel code in an integrated system for both the developer and the application user

2.1.2 Reducing complexity

In the FASTFLO project[2] it is acknowledged that the first requirement to be satisfied for CFD technology to have an impact on the design of an aircraft, is that the CFD problem turnaround time is of the order of a day to a week (Ref. 2). In the aerodynamic analysis, several geometric variants are studied at a significant number of flow conditions. The FASTFLO Project aims at a common, automated European CFD system. To provide results at the mentioned industrial time scale, the CFD system covers the trajectory from geometry to aerodynamic forces.

While the accuracy and computational performance issues are being addressed in the development of the FASTFLO system, the industrial partners in the Consortium are validating the FASTFLO system. This brings about the pure ICT issue of reducing the complexity of the FASTFLO system itself without affecting the components of the system. For instance, the grid-generator and flow solver consist of 9, resp. 6 separate programs. When running the analysis on a parallel computer, the number of files to produce the final grid file and final flow field, is

- 12 input files,
- $15+(A+1)*(2+D*L)+R$ intermediate files,
- to produce 2 result files.

where A is the number of grid adaptations, D is the number of domains, L is the number of multi-grid levels, and R is the number of restarts. For a practical case of 2 grid adaptations, 16 domains, 5 grid levels, and 5 restarts (Ref. 3), this gives 266 files.
Both the number of programs and the number of user files are too high for users in industry to apply the FASTFLO system; on the other hand, the current system of programs and files is a convenient set-up for developing the FASTFLO system further.

The ICT application of a system integration tool, SPINEware (Ref. 4), gives the opportunity to serve the industry users by reducing the complexity of the FASTFLO system, without disturbing the developers too much. In the system integration with SPINEware, the programs for grid-generation and flow solver are grouped into two groups, consisting of 3 resp. 4 compound programs. The number of compound programs is less than the number of programs because:

- some compound programs consist of chains of programs, which the developers have succeeded to make sufficiently robust,
- one compound program contains a switch between alternatives for geometry representations, and automatically calls the appropriate alternative.

[2] The FASTFLO Project aims at a common, automated European CFD system and is a collaboration between NLR, DLR, FFA, SAAB, DASA-LM, IBK. The project is funded by the CEC under the Brite Euram initiative (Project Ref. BRPR-CT96-0184).

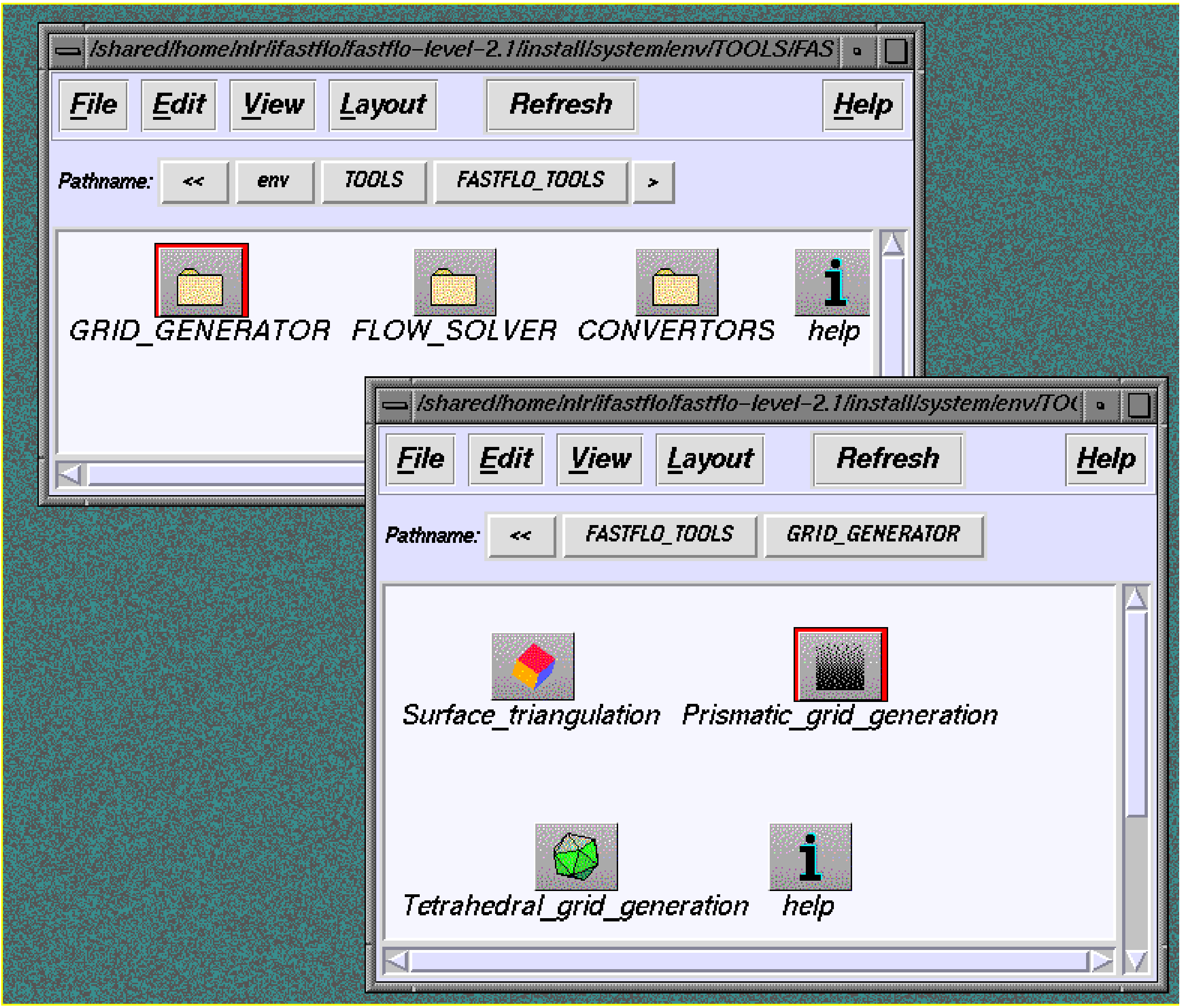

Fig. 4 View of the groups, and of the compount programs within the grid-generator group

This results in a more logical, and less implementation oriented, presentation of the FASTFLO system to the industrial user. Furthermore, in the icon representation of the group of compound programs (see Figure 4), the order of the icons reflects the order of the compound programs which helps the user in the path through the system. Finally, on-line help-information (see the `help` icons in Figure 4) is made available by the developers whereas the structure is defined by the system.

On the point of the large number of files, the solution has been sought in

– putting the geometry, grid, flow field, parameters in a top level directory
– separating out the intermediate files (directory `work`)

– separating out the log-information (directory `Log_files`)
– separating out the non-recoverable user input files (directory `inputs`)

resulting in

– input files, and one directory with input files,
– directories with intermediate files
– to produce 2 result files.

Note that these numbers are independent of, for example, the number of grid adaptations. Furthermore, the user encounters only the information on the top level directory, and sees maximum 9 entries, which is favourably low compared to the above practical example with 266 entries. The final view on the top level directory is shown in Figure 5.

The example shows that Information and Communication Technology involves

– reducing the complexity of the CFD tools, and
– supporting the industrial use of a code under development.

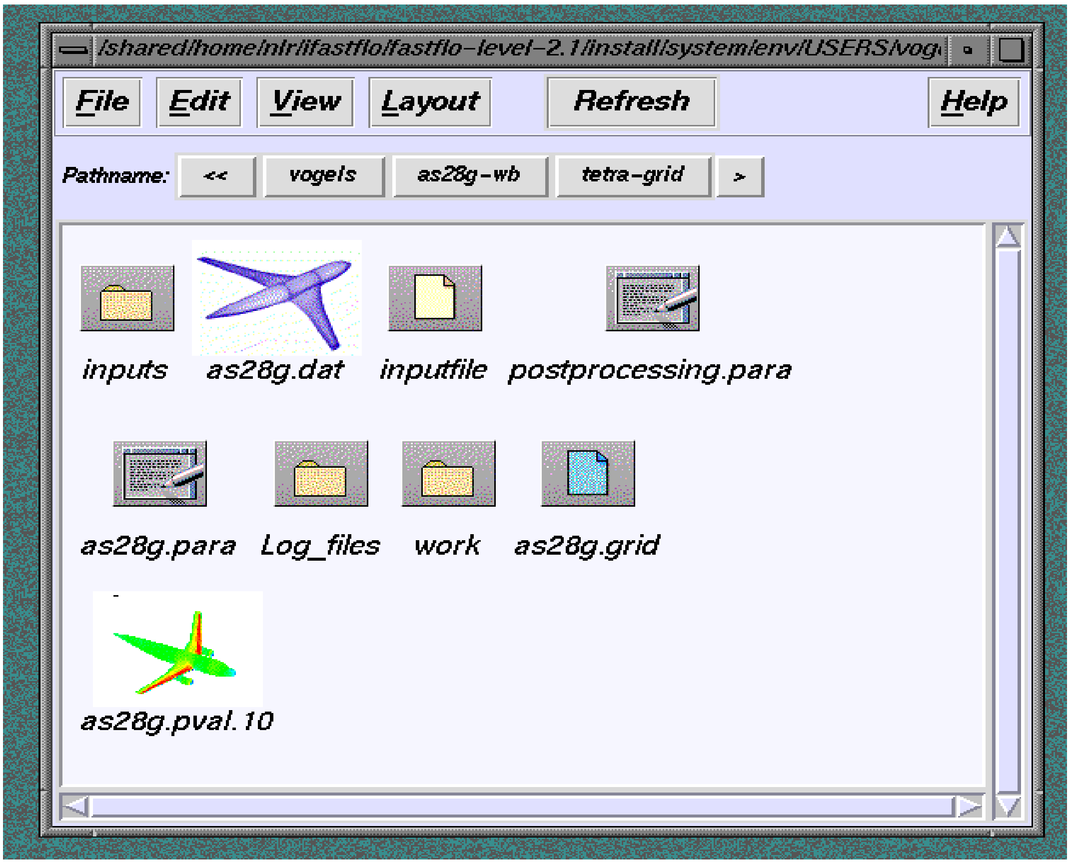

Fig. 5 Final view of data window

2.2 Multi-discipline analysis, design, and optimisation (MDO)

To effectively resolve cross discipline trade off's both to improve the product and reduce development time scales and costs, concurrent engineering principles are under investigation for the preliminary design stage. In Europe, the MDO Consortium[3] is addressing integration of design and analysis tool to create a Multi-Discipline Optimisation capability (Refs. 5, 6). One of the purposes is to validate the viability of MDO for simplified yet realistic aircraft preliminary design tasks concerning the design of a large civil aircraft where a non-trivial interaction is expected between the disciplines of aerodynamics and structures. The aircraft is shown in Figure 6.

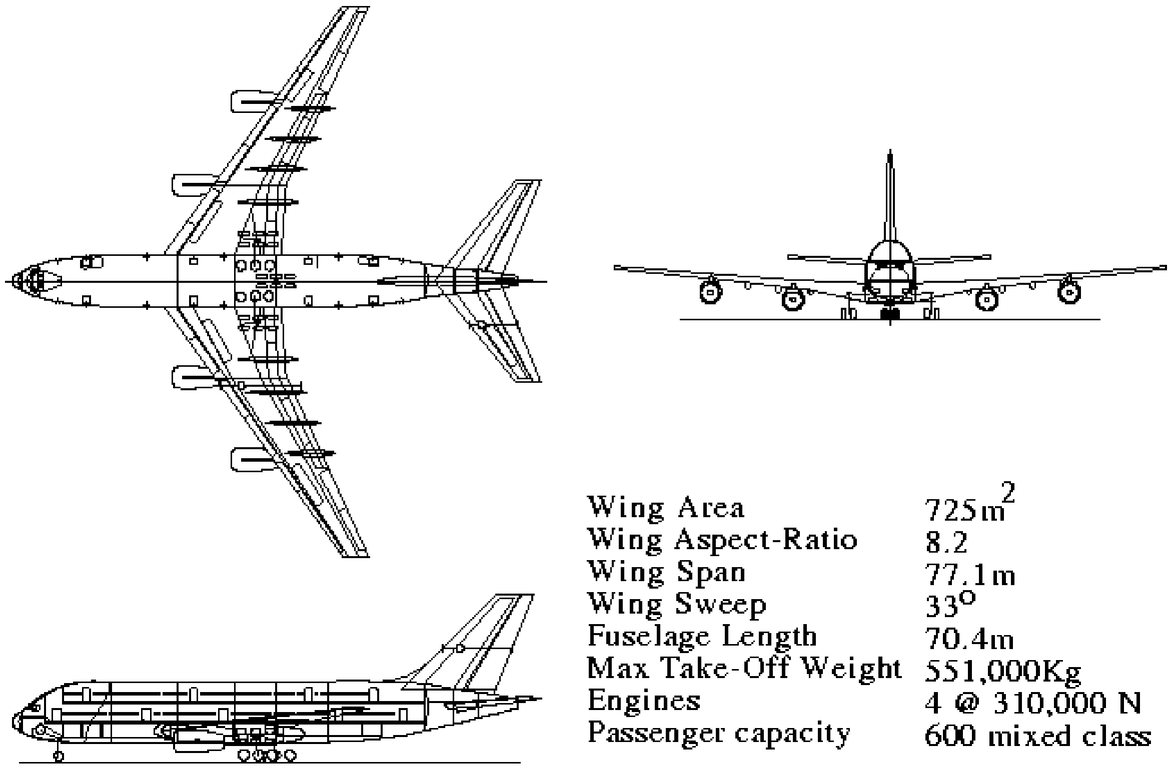

Fig. 6 MDO Reference Aircraft

[3] The MDO project (Multi-Disciplinary Design, Analysis and Optimisation of Aerospace Vehicles) is a collaboration between British Aerospace, Aerospatiale, DASA, Dassault, SAAB, CASA, Alenia, Aermacchi, HAI, NLR, DERA, ONERA, and the Universities of Delft and Cranfield. The project is managed by the British Aerospace and is funded by the CEC under the BRITE-EURAM initiative (Project Ref: BE95-2056).

At NLR, where the work has been carried out in a multi-disciplinary team with specialists from aerodynamics, structures, and information and communication technology, the ICT discipline is responsible for the coherence of activities on the ICT level (Ref. 7).

The analysis, design, and optimisation tools used by the multi-disciplinary team have been integrated using the system integration tool SPINEware (Ref. 4) to provide the team with access to all tools, while making the use of the computer network transparent.

Multi-Discipline Optimisation is an activity which requires both problem specification and problem solution (via process execution). Because of the nature of the MDO activity, the followed process in the sequence and the depth of the analysis steps should be flexible: the MDO process is characterised by continuously improving the process structure and the tools that are applied in the process (Ref. 8). Because the data can only be interpreted in the view of the process in which they were created, it is required that data and processes are integrated. The chosen pragmatic solution is to identify the data and the process in which they were created by the matching time stamps. The process is documented in N^2 diagrams as in Figure 7.

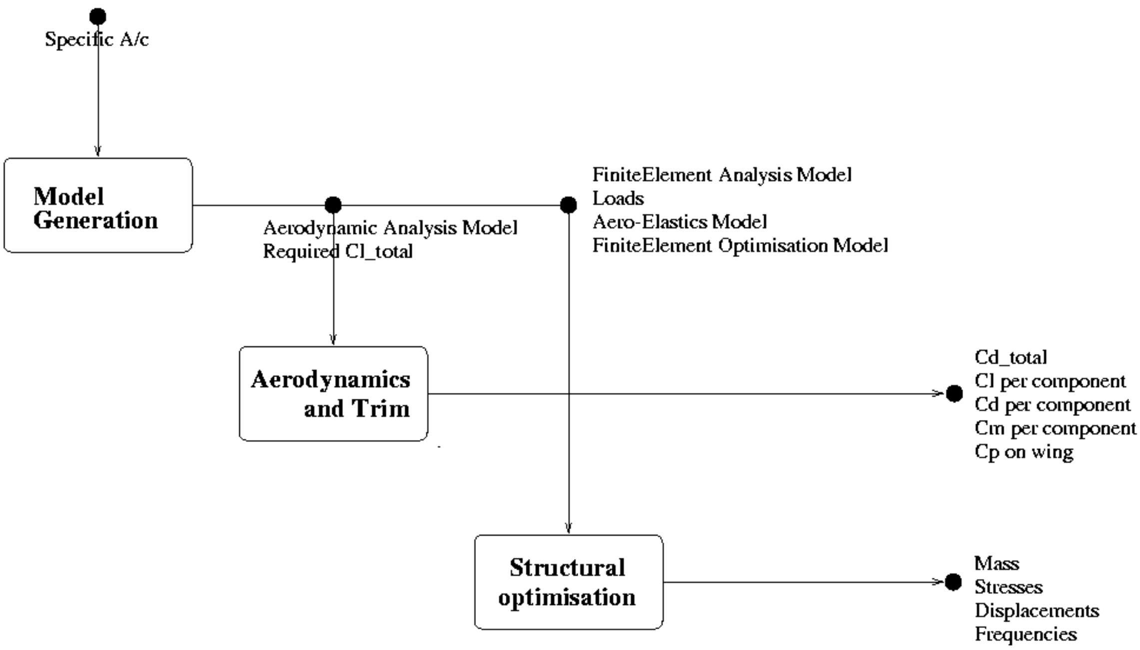

Fig. 7 Example of multi-discipline analysis process involving the Aerodynamics and Structures disciplines

Multi-Discipline Optimisation gives many opportunities for parallelisation. If the optimisation is driven by a gradient based algorithm, the objective function and its derivatives may be calculated by performing *k+1* parallel multi-discipline analyses for *k* design variables. The level of parallelisation is "embarrassing", t.i. the analyses can be carried in parallel, without any interaction between any two of the analyses. And within a multi-discipline analysis, the calculations for the separate disciplines may be carried out in parallel, as for example in the multi-discipline analysis process followed in Figure 7. In theory, these parallelisation dimensions can be combined at will with any parallelisation within a single run as described in Section 2.1.1. In the practice of the MDO project, however, this was not done.

Multi-Discipline Optimisation requires exploitation of parallelisation, because in practice computing times were found to be of the order of 1 hour CPU time per discipline, design variable, and optimisation cycle (Ref. 8) on today's fastest platforms. With typically 2 disciplines, 7 design variables, and 10 optimisation cycles, a full optimisation takes about one week if the calculations are performed in sequence. After such an optimisation, the work is not finished, but is followed by revisiting and improving the process structure and the applied tools, or even the MDO problem.

The evolution of the MDO process requires ICT to provide flexibility in the way tools are invoked. The developed mechanisms range from invoking a single tool, via forming a chain of tools (see Figure 8), to invoking fully automated iteration with a suite of analysis tools and an optimiser tool (Ref. 8). The first two mechanisms are provided by the system integration tool SPINEware; the latter mechanism is developed by BAe during the MDO project (Ref. 9).

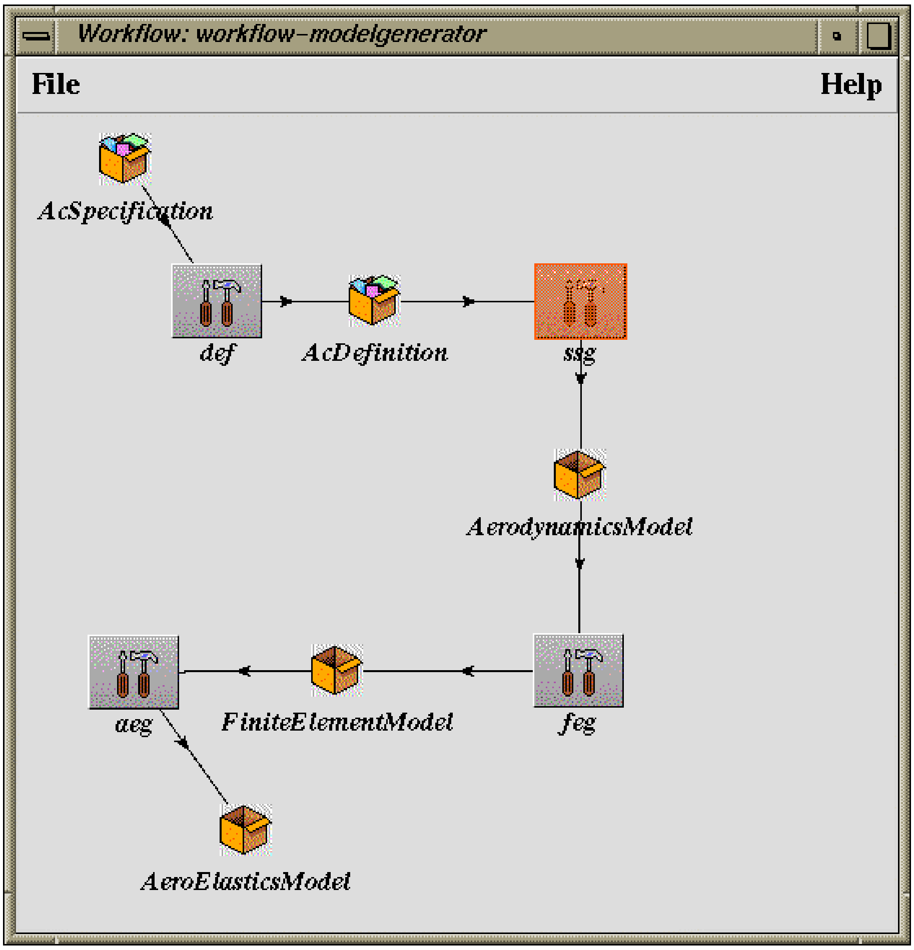

Fig. 8 Example of a chain of MDO tools (the program ssg is executing)

The example shows that Information and Communication Technology provides, next to the parallelisation,

- the followed process, and a description thereof
- a system to be used by the multi-disciplinary team

2.3 Enterprise competence

Enterprises are operating in a dynamic market, and try to survive by realising ever better satisfaction with the enterprises customers by:

- delivering the product in shorter time,
- producing at lower cost, and
- achieving better customer appreciation.

For this the enterprise tries to continuously increase the engineering know-how, the capability to apply the know-how in the business process and, on the metalevel, the know-why to ensure the enterprise capability to adapt to a changing market.

The combination of know-how, know-why and the capability to apply know-how is defined as *enterprise competence* (Ref. 10). Only proper management of competence enables an enterprise to cope with the need to lower cost when, at the same time, the speed of changes in the global market increases and additional requirements such as ecological aspects of the product life cycle, become more difficult to meet.

Enterprises have to react fast and in a flexible way to this changing environment and global competition. This means:

- attracting the right people
- educating and training the work force,
- and introducing and maintaining a competence oriented way of working and supporting tools in the enterprise.

Enterprises on the one hand need to continuously adapt the engineering work force to the work load to minimise enterprise cost. Engineers on the other hand need to be attractive for enterprises to be hired. As a consequence, enterprises are continuously searching for competent engineers whereas engineers are continuously searching for positions which increase their market value. The situation results in many cases in *a floating work force* in enterprises.
The position of an enterprise in the global competition is determined by the degree of *synergy* between the individual competence of engineers and the competence accumulated in the enterprise

The enterprise competence stems from the previous experiences in similar business processes. Where previously, the enterprise specific know-how and know-why was preserved in the work force, now, the enterprise has to take measures to preserve the know-how and know-why in other carriers, viz. software, data, and documents.

Whereas in the previous section the Information and Communication Technology was described from the view point of serving the engineers, it is clear that the enterprise is served as well. The system integration enables conservation, accumulation, management and application of the carriers of the enterprise specific know-how and know-why. For instance, by making the use of the computer network transparent, parallel CFD is not limited to a happy few but is a commodity (Ref. 10). And by providing a multi-disciplinary team with a friendly environment for the multi-discipline optimisation, the team's competence is automatically conserved and accumulated, and may be used at a later time or by a different team.

3 Conclusion

When discussing Information and Communication Technology in the context of industrial design with parallel CFD, numerous topics come to mind, in addition to the reduction of execution times of the compute intensive part of CFD analysis. Such topics include resource use, the user aspect, and enterprise competence.
Information and Communication Technology is about more than just programming tools.

Acknowlegdment
The contributions of all project partners in the NICE, FASTFLO, and the MDO project are gratefully acknowledged.

4 References

1. P. Wijnandts, M.E.S. Vogels. *Coarse-grain parallelisation of a multi-block Navier-Stokes solver on a shared memory parallel vector computer*. VECPAR'98. 3rd International Meeting on Vector and Parallel Processing. Porto, Portugal, June 21-23, 1998, NLR-TP-98380.

2. J.W. van der Burg, B. Oskam. *FASTFLO - Automatic CFD system for three-dimensional flow simulations.* Presented at the Conference on Industrial Technologies, Toulouse, France, October 1997, NLR-TP-97556 U.

3. J.W. van der Burg, NLR. May 1998. Private Communication.

4. W. Loeve, M.E.S. Vogels, H. van der Ven, E.H. Baalbergen. *Network middleware for enterprise enhanced operation.* Proceedings of the 4th International Conference on Computer Integrated Manufacturing ICCIM'97, Singapore, Springer 1997. ISBN 981-3083-68-9, NLR-TP-97224 U.

5. http://www.telecall.co.uk/~srcbae/mdo

6. S.E. Allwright. *Multi-Discipline Optimisation in Preliminary Design of Commercial Transport Aircraft.* ECCOMAS'96, Paris, France, September 1996.

7. M.E.S. Vogels, P. Arendsen, J.E. van Egmond, R. Krol, M. Laban, G.W. Pruis. *From a mono-disciplinary to a multi-disciplinary approach in aerospace: as seen from an Information and Communication Technology Perspective.* 21st ICAS Congress, Melbourne, Australia, September 13-18, 1998, NLR-TP-98226.

8. M.E.S. Vogels, S.E. Allwright, M. Stettner, Ph. Sims, P. Bartholomew. *The Information and Communication Technology's Contribution to the MDO Project.* Multidisciplinary Design and Optimisation Conference, London, United Kingdom, October 26-27, 1998, NLR-TP-99029.

9. Ph. Sims, M.E.S. Vogels, S.E. Allwright. *A Computational Framework to Support MDA and MDO.* 7th AIAA/USAF/NASA/ISSMO Symposium on MDO, St. Louis, Missouri, September 2-4, 1998.

10. W. Loeve and M.E.S. Vogels. *Competence Management in Engineering Environments with HPCN.* HPCN Europe '98. International Conference on High-Performance Computing and Networking. April 21-23, 1998, Amsterdam, the Netherlands, NLR-TP-98085.