

# COUPLING CAD AND CFD CODES WITHIN A VIRTUAL INTEGRATION PLATFORM

**S. Gatchell**, Hamburgische Schiffbau-Versuchsanstalt, Germany  
**RI Whitfield** and **AHB Duffy**, University of Strathclyde, UK  
**C. Abt**, FRIENDSHIP SYSTEMS, Germany

## SUMMARY

The Virtual Integration Platform (VIP) is an essential component of the VIRTUE project. It provides a system for combining disparate numerical analysis methods into a simulation environment. The platform allows for defining process chains, allocating of which tools to be used, and assigning users to perform the individual tasks. The platform also manages the data that are imported into or generated within a process, so that a version history of input and output can be evaluated.

Within the VIP, a re-usable template for a given process chain can be created. A process chain is composed of one or more smaller tasks. For each of these tasks, a selection of available tools can be allocated. The advanced scripting methods in the VIP use wrappers for managing the individual tools. A wrapper allows communication between the platform and the tool, and passes input and output data as necessary, in most cases without modifying the tool in any way. In this way, third-party tools may also be used without the need for access to source code or special modifications.

The included case study demonstrates several advantages of using the integration platform. A parametric propeller design process couples CAD and CFD codes to adapt the propeller to given operating constraints. The VIP template helped eliminate common user errors, and captured enough expert knowledge so that the casual user could perform the given tasks with minimal guidance. Areas of improvements to in-house codes and to the overall process were identified while using the integration platform. Additionally, the process chain was designed to facilitate formal optimisation methods.

## 1. INTRODUCTION

Ship hydrodynamic design today is often still a sequential approach. Sequentially dealing with separate hydrodynamic design aspects precludes fully taking into account the mutual influences and interactions between those aspects. As the tools used for the different aspects, and even for the different levels of detail within a single aspect, are often poorly integrated, the consistent use of hydrodynamic prediction tools in design is not always efficient and therefore often not fully adopted.

Most tools are typically coupled via external data exchange which is realised by file transfer utilising either standard formats or bilateral formats. Data exchange not only between different institutions, but also at a single site among the various divisions, is often time consuming and labour intensive, due to pre- and post-processing of data and additional checks on validity and conformity.

The interaction with the customer (e.g. a shipyard representative) is important to define the available optimisation space and design constraints. This interaction needs to fit into the short timeframe typically available in ship design.

Therefore, it is desired to significantly improve the level of integration while maintaining an open and flexible environment. Consistently sharing and distributing information at all levels needs to be supported to a much higher degree so as to increase productivity and to consider the various hydrodynamic design aspects

holistically rather than sequentially. Improved design methods and practices are necessary to meet these demands.

The objective of VIRTUE Work Package 5, integration platform development, is to provide a system that will enable various distributed CFD and design applications developed within VIRTUE to be integrated so that they may operate in a unified and holistic manner. The distributed nature however imposes a number of integration challenges to ensure that the data is consistent between the applications involved, that changes and interactions between the applications are correctly propagated and that the simulations are undertaken in an organised manner.

## 2. INTEGRATION PLATFORM

The Virtual Integration Platform (VIP) developed within the VIRTUE project aims to provide support for the integration of existing design and CFD tools into a virtual environment, that allows distributed designers to collaborate within the CFD design process. These tools have been organised within four “virtual basins” relating to resistance (towing tank), sea-keeping, manoeuvring and cavitation [1]. One of the aims of the VIP is to provide data consistency management such that a geometric representation of a hullform for example could be created and modified by a designer and subsequently be made available to a CFD expert for analysis within a different organisational or geographical location. Since the VIP operates in a distributed sense, the management

of data consistency is further confounded by the fact that there may be multiple, different versions of the geometry being operated on by multiple, different designers, which may in turn be used as input to different CFD tools representing the four virtual basins.

A novel approach has been implemented within the VIP to manage the distributed access to the data that is generated within a VIP project. The approach relies upon a centralised database that contains XML based meta-data defining the distributed locations of the design and CFD data. The database is structured such that it keeps a constantly updated Uniform Resource Locator (URL) for each version of each file within a project, defining where these files may be distributed on an organisation's network or on the Internet.

Another key component of the data management approach is a series of distributed File Transfer Protocol (FTP) servers that store and manage the access to the data generated within a project. The URLs within the centralised database essentially point to a particular version of a file stored within one of the distributed FTP servers. One of the main reasons for developing this approach to data management within a distributed CFD environment was due to the size of the files created by both the design and CFD codes which could typically be in the region of 100MB. The implemented approach attempted to minimise the amount of data that needed to be communicated across a network, whilst utilising the most robust technology for transferring files. The components within the data management approach can be seen within Figure 1.

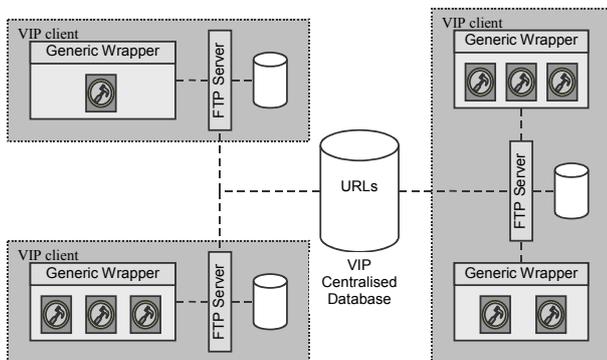


Figure 1 VIP components for distributed data management

The controlling element for managing the transfer of data between designers and CFD experts is the generic wrapper, which is also responsible for managing the integration of any type of CFD tools, irrespective of their underlying programming language or operating system, into the virtual environment. If designers choose to modify the hullform geometry for example, the generic wrapper firstly establishes which version of the hullform (within the current project) the designer requires. The generic wrapper then queries the URL database to establish which of the FTP servers is responsible for

storing the required hullform, and once a URL has been identified, the generic wrapper communicates directly with that FTP server to download the data. The designer may then modify the hullform using whatever design tool that they have wrapped into the VIP, and once completed, the wrapper then uploads the modified hullform back to either the original or their own FTP server, modifying the meta-data in the database to show an update, as well as information relating to when the data was modified and by whom. A CFD expert would undertake a similar process (automated by the generic wrapper) to retrieve the modified hullform file in order to perform their analysis.

The FTP servers contain only data that has been associated with a project, either that which has been initially provided, or derived through use of the platform; hence VIP users have restricted access to the data that is otherwise available on each computer within the network. It can also be seen from Figure 1, that the VIP architecture is capable of supporting multiple FTP servers, either used individually, or shared between multiple designers and CFD experts. It has been observed through the use of the VIP, that managing the data in this way provides a coherent approach to consistency that can result in a significant reduction in effort expended in locating, modifying and transferring data between designers and CFD experts. In addition to managing file-based data, the VIP can also manage parametric data which may automatically be inserted into or extracted from the files that are associated with a VIP project. All parametric data is stored within the centralised database.

There is however an additional need to co-ordinate the design and analysis activities that are undertaken through the use of the VIP due to its distributed nature – managing the transferral of data between VIP users is insufficient to ensure that a project is undertaken successfully, and design and analysis is undertaken in a logical manner. To satisfy this need the VIP has been designed around a project and process concept, where tasks within a process are mapped by the VIP users to tools wrapped within the VIP. A project may be defined by a user having a project management role, and be created with data relating to project identifiers, start and end dates, client information, contract details, as well as a list of associated processes that will be undertaken in order to complete the project. These processes may for example be related to specific design and analysis aspects of cavitation around a propeller, i.e. within a single virtual basin; or may have a broader context considering towing, sea-keeping and manoeuvring and subsequently require effort across multiple virtual basins. The VIP manages information relating to the users of the platform, and during project and process creation, this user information is used to associate “expertise” to the individual tasks within a process. This allows a project manager to distinguish which users are considered to be capable of undertaking the associated task.

Once the creator of the project has associated users to the tasks within a process, the user will see the project details within their VIP client when they log onto the platform. The user subsequently has the ability to open the project and participate in a distributed collaborative manner. The tools wrapped by a user are automatically associated with the appropriate tasks within the process. The tasks within a process must however follow a logical sequence of enactment, dependent upon the prior completion of preceding tasks. Providing this structure aims to ensure that the tasks within a process are undertaken for the right reason at the right time, with the beneficial side-effect that the data consistency is automatically maintained.

The processes may contain any degree of complexity-containing a simple design/analysis loop or consisting of multiple design tasks iterating with multiple analysis tasks, with the whole project being co-ordinated across multiple designers and CFD experts. Optimisation is currently being implemented within the VIP to allow either entire processes, or sub-processes to be enacted, automatically guided by parametric concepts generated by the optimisation tool (modeFrontier).

### 3. CASE STUDY

During the development of the integration platform, several case studies were designed and executed by the end-user partners. These case studies helped to validate and evaluate the functionality and applicability of all of the components within the integration platform, as well as the tools integrated.

#### 3.1 PROPELLER DESIGN APPLICATION

The following is a case study developed and tested using the VIP. The subject of the case study was a propulsion prediction exercise, to compute and adapt the propeller thrust and torque for a given operational condition, namely, behind a ship at a given operating speed and load condition.

#### 3.2 SHIP AND PROPELLER PARTICULARS

The container ship, Hamburg Test Case (HTC), was selected for numerical analysis. This ship is widely published and has been used in other CFD projects [2], and was selected as one of the standard test subjects for the entire VIRTUE project. More details for the HTC are given in [3].

The propeller blade sections were based on a generic NACA 66 profile (meanline  $A=0.8$ ). The main data of the propeller were fixed except for the mean value of the blade pitch.

The main particulars of the vessel and propeller are given below:

#### Ship: HTC

Lpp	153.70 [m]
T, fore	10.30 [m]
T, aft	10.30 [m]
Block Coeff.	0.651
Vs	18 knots
Froude No.	0.238

#### Propeller

Dp	6.10 [m]
Rot. speed	1.652 [rev/sec] 100 [rpm]
Adv. Coeff., J	0.92

### 3.3 BACKGROUND

Propulsion prediction evaluations are necessary to ensure that the propeller performance matches the predicted engine power or the required thrust at the right RPM. The non-uniform flow of the wake distribution affects the propeller performance. One way to address this is to adjust the propeller geometry, taking this non-uniform flow into account. Constructing a propeller and even a propeller model for each variation would be too expensive; computer simulations of the propeller performance of many variations is a cheaper alternative. Panel codes offer a significantly shorter computing time than full-RANSE computations, while still offering a fairly good prediction.

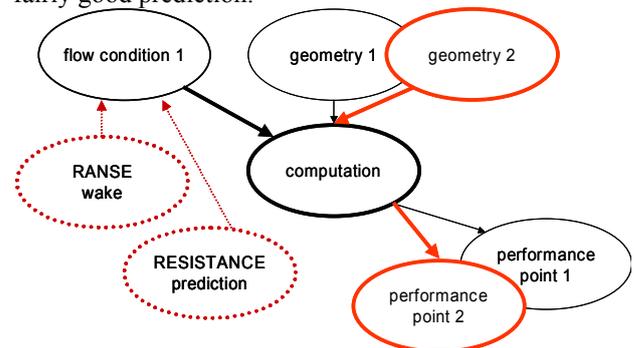


Figure 2 General process workflow

The figure above illustrates the general workflow for a propeller design exercise. The initial flow conditions, “flow condition 1”, contains the relevant data for the operating conditions, given by “RANSE wake” and “RESISTANCE prediction”. The “RESISTANCE prediction” (including thrust deduction) is the target thrust value that the propeller should produce (within reasonable margin). In this example as in the case study, only one set of flow conditions is considered; the changes are made to the geometry. A baseline geometry is established, indicated by “geometry 1”, and the performance of this geometry is computed in “computation” to produce “performance point 1”. This performance point is compared with the target thrust value. Any modifications to the geometry are defined in the subsequent “geometry 2”, and the associated “performance point 2” is compared to the target value.

In a typical propeller optimisation, balancing the geometry modifications with the performance criteria is an iterative process. At each iteration, a geometry is defined, the performance is evaluated, and the designer must decide what changes may be necessary. This process repeats until all requirements have been fulfilled.

The propeller designer has many parameters to consider when modifying a propeller design. The pitch distribution affects thrust and propeller efficiency. Changing the camber and thickness distributions influences the cavitation, flow separation, and chordwise load distribution. Changing the skew is a common measure to reduce the hull pressure fluctuations, and is helpful in preventing erosion due to cavitation. Parameter variation continues until all the performance criteria are met. At that point an acceptable design has been reached.

### 3.4 PROCESS CHAIN

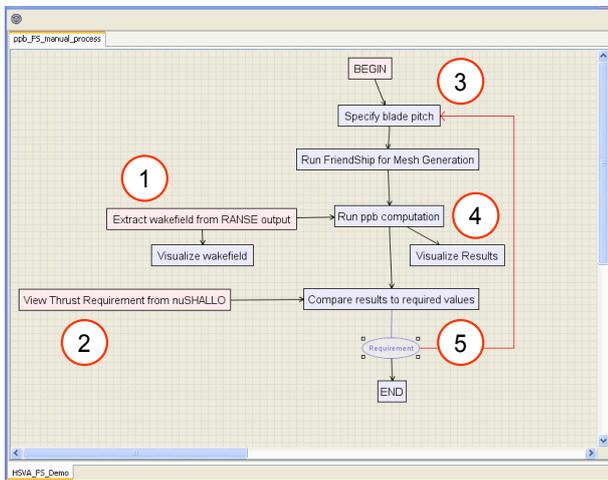


Figure 3 Design process within the VIP

The figure above shows the process chain within the integration platform. Each box represents a task to be performed by a specified user. For each task, one or more tools have been allocated; the user has the option which tool to use, depending on availability and preference. The process chain for this case study consists of five (5) main tasks:

1. Extract wake distribution from RANSE output
2. Compute thrust requirements in **nuShallo**
3. Create geometry and mesh in **FRIENDSHIP**

#### Framework

4. Compute performance in **ppb**
5. Evaluate results

Steps 1 and 2 provide the flow conditions and the target thrust value, respectively. Steps 3-5 are the iterative loop, repeated as many times as necessary in order to fulfil the design requirement. The five tasks and their associated tools are described in more detail in the section below. The three tools listed- nuShallo, FRIENDSHIP Framework, and ppb- were the three main components within this case study.

### Task 1 Extract wake distribution from RANSE output

A double-body RANSE computation was performed to produce the wake distribution, as input to the ppb computation. This computation was performed outside the platform since the tool had not yet been integrated. However, the output from the RANSE computation could still be used. A data translation module was executed from within the platform and extracted the wake distribution information from the RANSE output. The wake distribution information was saved to the set of “derived data” in the common data model. The platform data management system provided the wake distribution to the ppb code as needed.

The additional sub-task, “visualize wakefield” was appended to the RANSE computation. Although it is not essential to the work flow, it provides a very useful graphical representation of the wake distribution. The visualization software Amira, developed by ZIB, is used throughout the VIRTUE project as the preferred visualization tool. The figure below shows the normalized axial component of the velocity at the propeller plane, as depicted in Amira:

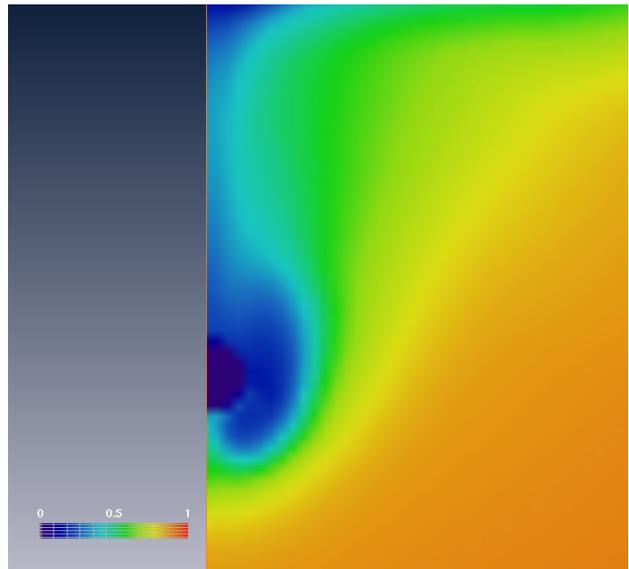


Figure 4 Axial velocity components at propeller plane

### Task 2 Compute thrust requirements in nuShallo

The panel code nuShallo was fully integrated and executed within the platform. This tool is an in-house, commercially available panel code, developed by HSVA. A panel mesh and a text control file are used as input; several options are available for the output. For this case study, a single value for the total resistance was parsed from the nuShallo output text file. This value was stored in the platform common data model as “derived data”, and could be recalled as needed at any time during the process.

### Task 3 Create geometry and mesh in FRIENDSHIP Framework

This case study applied parametric principles to propeller design. The potential of this approach was investigated in the context of the VIRTUE platform. An open and flexible set of parameters was specified using the core of FRIENDSHIP's modelling system. A form parameter-oriented approach was implemented that allowed the efficient generation and effective variation of the propeller geometry.

In order to keep the case study simple, the design changes were limited to one parameter: controlling the pitch distribution. A baseline propeller was modelled in FRIENDSHIP-Framework. It was constructed so that the pitch distribution curve could be shifted by a single control value. The FRIENDSHIP-Framework generated a panel mesh, which was then evaluated using HSVA's propeller panel-based (ppb) code. The figure below shows the propeller geometry in the FRIENDSHIP-Framework fully parametrically defined by a set of radial functions for chord, pitch, camber etc., see lower left window, the inset showing the completed panel mesh:

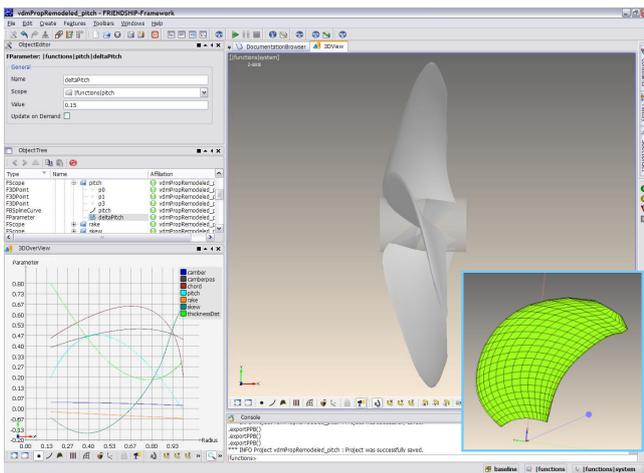


Figure 5 Propeller in FRIENDSHIP-Framework, inset: panel mesh

The FRIENDSHIP-Framework was configured to run in batch mode, where the integration platform passed the mean pitch value to the tool on execution. The FRIENDSHIP-Framework then used this value to generate a new propeller geometry and the corresponding panel mesh, fully automated. This batch mode execution is necessary for later implementation of formal optimisation methods. The user still has the option of running the Framework in interactive, GUI mode, and would have full control over the entire geometry model.

### Task 4 Compute performance in ppb

HSVA's in-house propeller panel code, ppb, was used as the third main component in this case study. A panel

mesh file, wake distribution file, and some text configuration files are used as input to the computation. As output, ppb creates a text file listing the numerical results, along with several PostScript files and a Tecplot file for visualization. The pressure distribution over the blade is shown in the Amira visualisation below:

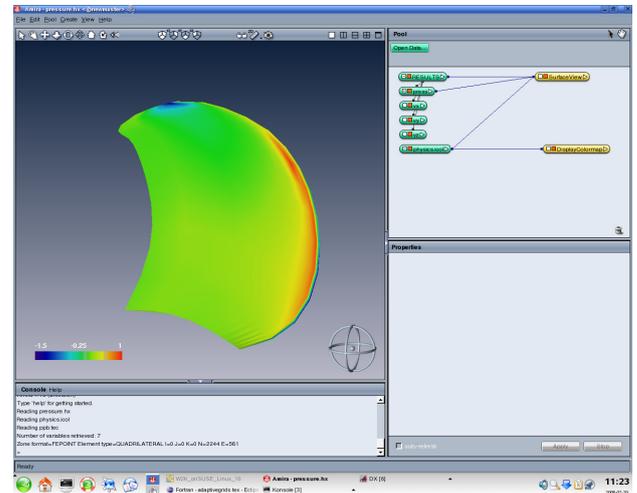


Figure 6 Pressure on blade pressure side

For the case study, only one piece of information from the numerical results was needed, namely the thrust value. This value was extracted from the text results, and uploaded to the common data model as "derived data". The data management system in the VIP tracks each iteration of the design process, for evaluation in Task 5.

### Task 5 Evaluate results

A simple text editor served as evaluation tool in this task. The tool wrapper was configured to create a text file, inserting the input and output values and the requirements, where necessary. Although this is a simplistic example, the usefulness of this task should not be underestimated. Complex scripts, macros, or formatting tags may be included in the tool configuration, a step toward automatic report generation.

For the case study, the thrust value from ppb in task 4 was then compared to the nuShallo thrust prediction, from task 2. The design was considered successful when these two values were within 10% (sea margin). If not, another iteration, tasks 3-5, was necessary. The table below shows the results of four (4) iterations to satisfy the convergence criterion:

Iteration	DeltaPitch	Computed thrust, kN	Compare to required 648.23 kN
1	0.00	410.95	0.634
2	0.15	1235.03	1.905
3	0.11	1006.23	1.552
4	0.05	647.66	0.999

## 4. WORKFLOW IMPROVEMENTS

Over the course of the case study, several areas of improvement were identified in individual tools, as well as the overall workflow. In-house codes were expanded to include more options for input and output formats, in some cases for more adherence to a common data format. Additional improvements were made to facilitate future implementation of formal optimisation methods. The workflow was improved in two ways: the workflow was simplified for non-expert users, and faster turn-around time per iteration.

### 4.1 TOOL IMPROVEMENTS

Some areas of improvement were identified in the ppb code as a direct result of wrapping the tool and integrating it into the platform. Command-line arguments were included in the executable, thereby eliminating the need for the user to manually input certain values. This serves two purposes. First, the wrapper could be configured to provide these values automatically, reducing possible errors. Second, this allows for automated execution, essential for formal optimisation methods.

Another improvement to the ppb code was through additional visualisation outputs.

### 4.2 SIMPLIFIED PROCESS

The workflow for this propeller design exercise was also improved through the use of the integration platform. All of the configuration files for a ppb computation were included in the tool wrapper. These files could be left unchanged, and the user was not overwhelmed by the many details they included. This reduced the amount of direct, manual interaction with the configuration files.

### 4.3 TIME SAVINGS

Usually, when a process is simplified, this leads to time savings. By automating the data management, the entire workflow was even faster. Additionally, less time was required from the “expert user”, especially for the in-between steps, where little decision-making was required. Only in key moments, was the expert required. For the simple one-parameter optimisation demonstrated in this case study, the expert was only needed at the beginning for set-up, and at the end, to verify that the requirements had been met.

Based on the experience with the case study above, it is estimated that with the introduction of formal optimisation methods, the time savings for this type of process could be even greater. Allowing the process to run, autonomously, overnight could evaluate hundreds of design variations over a multiple parameter search field.

## 5. CONCLUSIONS

Addressing the need to integrate design tools so that they may operate in a unified and holistic manner, the VIRTUE project develops the Virtual Integration Platform. The distributed nature however imposes a number of integration challenges to ensure that the data is consistent between the applications involved, that changes and interactions between the applications are correctly propagated and that the simulations are undertaken in an organised manner.

Distributed resources and data are managed from a central database. The mapping of processes into the integration platform improves the overall workflow in several ways, by identifying areas of improvement to individual tools, and automating data handling, which reduces human errors as well as increases time savings.

Work on the VIP continues, as improvements to current functions and implementation of new functions are in constant development.

## 6. ACKNOWLEDGEMENTS

The Virtual project is funded by the European Commission’s 6<sup>th</sup> Framework Programme – Sustainable Surface Transport, under the grant TIP5-CT-2005-516201.

The authors would like to acknowledge the efforts of all the WP5 partners. Special thanks to Christoph Petz at ZIB, for the additional technical support provided throughout the platform testing, and also thanks to Stefan Wunderlich and Heinrich Streckwall for implementing the necessary translators for coupling the codes.

## 7. REFERENCES

1. J. Marzi, ‘VIRTUE – A European Approach to Developing the Numerical Towing Tank’, *RINA CFD*, 2008
2. K. Chao, et. al, ‘Database of full scale and model flow measurements’, *EFFORT 5<sup>th</sup> Framework Programme*, 2003
3. Qiuxin Gao, Dracos Vassalos, ‘Computational Hydrodynamic Derivatives by Numerical PMM’, *RINA CFD*, 2008

## 8. AUTHORS’ BIOGRAPHIES

**Scott Gatchell** holds the current position of Senior Researcher at the Hamburg Ship Model Basin. He is responsible for mesh generation developments in the CFD department during daily operation, and for some WP5 management in the context of the VIRTUE project.

**Robert Ian Whitfield** is a Lecturer within the department of Design Manufacture and Engineering Management at the University of Strathclyde. He is currently co-ordinating and undertaking research within a number of European Union FP6 and EPSRC research projects.

**Alex Duffy** holds the current position of Professor and Director of the CAD Centre within the department of Design Manufacture and Engineering Management at the University of Strathclyde. He lectures in CAED Systems, Design Management, Information Management, Knowledge Engineering and Management, Product Development, and Research Studies

**Claus Abt** holds the current position of Managing Director at FRIENDSHIP SYSTEMS. He is responsible for product development and CAD to CFD integration projects for customers in the marine and aeronautical industry.