EMPOWERING ENGINE ENGINEERS

ADVANCING THE STATE-OF-THE-ART IN COLLABORATIVE MULTI-NATIONAL MULTIDISCIPLINARY ENGINE DESIGN

E. Kesseler, P. Arendsen, M.H. van Houten {kesseler, arends, houten}@nlr.nl+
R. Parchem, B. Meissner {Roland.Parchem, Bernd.Meissner}@Rolls-Royce.com°
M. Nagel, J. Barner {Marc.Nagel, Johannes.Barner}@muc.mtu.de†,
H. Wenzel holger.wenzel@engineous.com‡
+ National Aerospace Laboratory NLR, P.O. Box 90502, Amsterdam, The Netherlands
° Rolls Royce-Deutschland, Dahlewitz, Germany
† MTU, München, Germany
‡ Engineous, Germany

ABSTRACT

Economic realities result in ever increasing demands on aeronautical engines as well as on their development programmes becoming virtual organisations. Design evolution requires close collaboration between the engineers involved, across disciplines, geographic sites and organisations. Typically each organisation will have its own proprietary tool suite. This paper presents the realisation of a virtual enterprise allowing flexible addition or modification of partners and/or accommodation of tool suites. The approach demonstrates efficient support of evolving engine design for a realistic design problem in a realistic organisational setting thereby alluding to its potential in similar high-tech design collaborations.

1. INTRODUCTION AND BACKGROUND

Aircraft engines have evolved into complex high-tech systems. Complying with ever increasing environmental and economical requirements requires continuous design improvements. Engine design is correspondingly technically challenging and costly. This implies that the integrator has to work closely with its first tier suppliers, also during the engine design phase. To obtain an optimum design, technical information needs to be shared between the collaborating partners. As these partners are risk-sharing, they will use their own, partly proprietary, tool suite. However this design tool suite is the core asset of each partner and different engine collaborations comprise of different partners, so the intellectual property of each partner needs protection. Consequently a partner needs access to other partners' tool suites within a specific engine collaboration framework, but can not be allowed a copy of it. During the course of the engine design, the design objectives will inevitably evolve, which the collaboration needs to accommodate efficiently. The result is a need for a multi-site multi-company collaboration, also referred to as virtual enterprise[1].

2. VIRTUAL ENTERPRISE FOR ENGINE DESIGN

Within the framework of a large European research initiative, nine organisations from four European countries have realised a prototype implementation of a virtual enterprise for collaborative engine design. The organisations and their roles are representative of actual engine development comprising a system integrator, various engine manufacturers acting as first tier suppliers, an IT service provider, a research establishment and academia. Technical information covering the full collaboration is provided elsewhere[2]. This paper elaborates the detailed design phase.

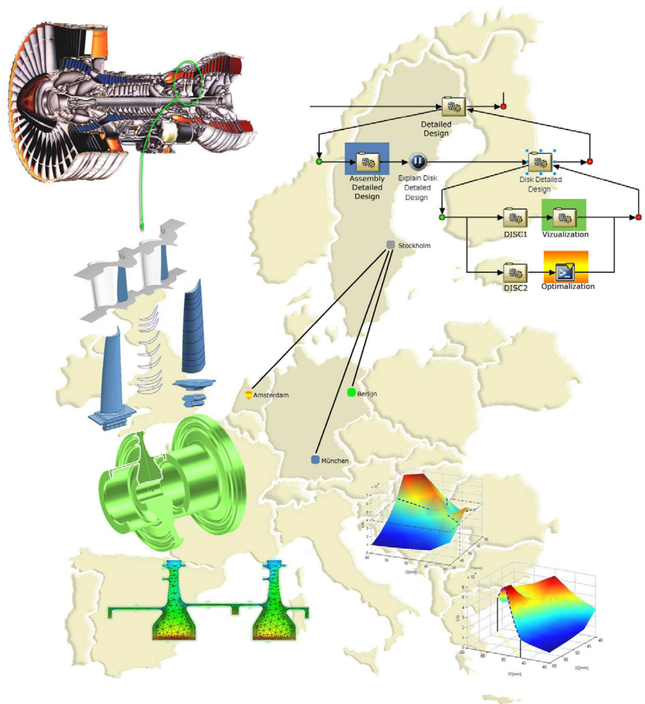

FIG 1. Multi-partner multi-site virtual enterprise collaboration for an engine multidisciplinary design analysis process.

Figure 1 provides a top-level overview of the implemented collaborative virtual enterprise engine design processes. The integrator provides the flight mission data. A two stage high pressure turbine is selected as it presents a demanding engine assembly. In the early design phases like the feasibility and conceptual phases the integrator uses whole engine simulation tools[8, 9]. During the preliminary design the requirements for the detailed design phase are derived, which is elaborated in this paper. Where needed the integrator role will be elaborated. Chapter 3 elaborates the airfoil design of a two-stage turbine, consisting of two rotors and two stators. Included is a description of the way this capability is provided to the partners of the virtual enterprise. A second partner provides a solution to share results within the collaboration. Continuous design evolution requires all data exchange to be automatic, i.e. no manual data conversion is required. The third partner uses the results to perform a disc design based on the provided design variables, as described in chapter 4. The resulting detailed technical information is translated by yet another partner into a form suitable for the last partner in the design chain who performs a similar detailed design of the second disc, but with a different tool suite, of which the optimisation part is described in chapter 5. This multi-site multi-company collaboration is based on web-services, allowing each partner to access the design assets of the other partners, without violating their intellectual property rights and is elaborated in chapter 6. The conclusions are provided in section 7.

3. DETAILED AIRFOIL DESIGN

The design of a two stage high pressure turbine is demonstrated in the virtual collaborative engine design loop example. The virtual design loop with all partners is done twice. A preliminary design of the engine component is done in the first loop. The detail design of the engine part takes place in the second loop. The first design step in the loop is the aerodynamic design of the rotor and stator blades at one partner of the virtual enterprise. The details of the step of the aerodynamic design are presented in this section of the paper. This section is split into two parts. A short introduction of the aerodynamic optimisation system is given in the first part. In the second part the technical connection of this process to the integrator and other partners is presented.

3.1. Aerodynamic optimisation

The inputs for the aerodynamic design are the requirements and constraints of the engine component. The integrator defines this flight mission data and sends them to the first partner. The computational effort for the 3D detailed aerodynamic design based on this input is high. On one hand this is caused by the high dimensional search problem for the 3D design, on the other hand, because a time-consuming 3D Navier-Stokes flow solver must be used.

The proprietary optimisation system *opus* (optimisation utility system) is used for this optimisation. Many processes for geometry modifications e.g. modification of the blade shape, pre-processing e.g. mesh generators, solvers and post processors are integrated in opus in a fully automated way. The system allows the airfoil designer to compose the process chain he wants to run during the design optimisation out of these implemented processes. The system has a simple file interface to optimisation techniques. Actually it hosts a huge variety of optimisation algorithms, e.g. gradient based, genetic and simulated annealing algorithms are integrated, as well as an interface to the algorithms of the commercial optimisation system[10]. The handling of the design variables and the result values, the objective function and the constraints and log files is done by the central system of *opus*. The system uses every possible, even nested, parallelisation for performance reasons. Parameter vectors during gradient search, or populations during genetic search are run in parallel. Parallelisation is integrated in a way that every discipline may run in parallel to each other, combined with unlimited further parallelisation within each part. Time-consuming calculations are processed on Linux Clusters. The designer interacts with the system through a graphical user interface. The effort for the problem definition, conditioning, strategic constraints and targets is reduced by pre-settings.

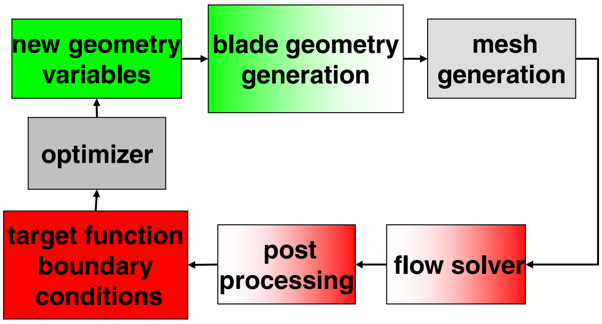

FIG 2. Process chart of the 3D blade shape optimisation.

In the case presented the processes for 3D blade optimisation are used (Figure 2). The 2D airfoil shape was parameterised by a set of 4th grade polynomial splines for the suction side, the leading edge, the pressure side and the trailing edge. The parameters are non-dimensional values, i.e. not depending on the blade dimensions. The parameters of the 2D sections are linked together by an additional radial parameterisation to ensure a radially smooth blade [5]. In the next step a CAD-model based on the parametric defined blade sections is build up using Commercial Off The Shelf (COTS)[11, 12] Tools. The flow channel is discretised by a structured multi-block mesh directly based on the Computer Aided Design (CAD) model. The aerodynamics solution is solved with the 3D-RANS TRACE on multiple processors on a Linux cluster. The boundary conditions of flow angles and pressures for example result from the operating point defined by the integrator. The objective of this optimisation is the reduction of aerodynamic losses.

At the end of the optimisation the CAD-models of the optimised rotor blades are completed with platforms and the fir-tree defined during the preliminary design loop (Figure 3).

FIG 3. Optimised blade completed with platform and fir tree.

The CAD-model of the optimised blade is sent to the integrator.

3.2. Connection to the partners/integrator

The distributed development process in the virtual enterprise demands a security infrastructure satisfying the commonly agreed virtual enterprise hub security requirements as well as the individual partners' security policies.

3.2.1. Prototype implementation

A prototype infrastructure was implemented to demonstrate the realisation of the virtual enterprise partner's minimal company security requirements based on a internet connection depicted by figure 4. The aerodynamic optimisation is offered as a web service to the virtual enterprise by publishing the service definition in a Web Service Definition Language (WSDL) file. To utilise the service a SOAP message (Simple Object Access Protocol) according to the service definition is sent to vivace.mtu.de by the requesting partner (the integrator). The message contains the input data and is packed into a HTML request. After authorization by the providing partner´s Lightweight Directory Access Protocol (LDAP) server the request is forwarded to the internal SOAP server and verified. The server unpacks the input data and starts the optimisation on a Computer Aided Engineering (CAE) compute server. The results are again packed into a SOAP data structure and sent back to the integrator.

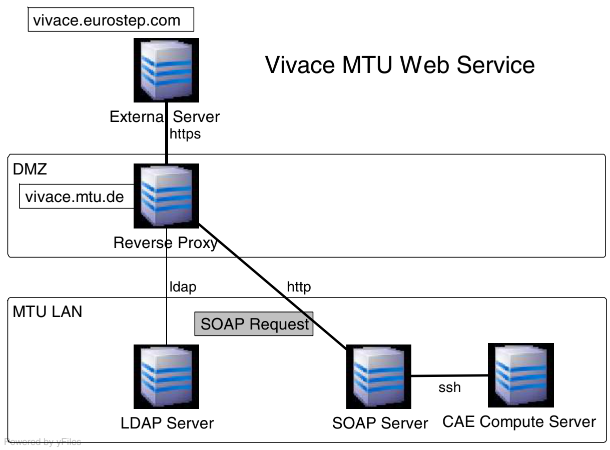

FIG 4. Prototype Security Infrastructure

The following security requirements are considered:

- Content security against unauthorised access, Content security outside the providing partner is achieved by HTTPS encryption, inside the partner by access control on the reverse proxy;
- Authorisation.
  The authorisation data is checked by the LDAP server containing the allowed service users. The prototype also shows the ease of implementation and maintenance, an additional important requirement beside security.

To gain full security on the partners side the following requirements have also to be considered:

- Content security against malicious content;
- Authentication.

All mentioned requirements (except authentication) are taken into account for a new security infrastructure concept. For example a single sign-on infrastructure has to be agreed by all partners so it cannot be implemented independently.

3.2.2. Security Concept

Since the input/output data in SOAP messages is usually encoded a conventional HTTP scanner will not detect malicious content hidden in eXtensible Markup Language (XML) data. It is necessary to use a scanner which understands XML structures in a generic way and sends the raw element contents to a virus scanner. The known data formats (eg. .gz .zip, .exe, base64, etc.) can now be unpacked if necessary and scanned. In case of virus suspicion the SOAP request is rejected, otherwise the message is forwarded unaltered to the SOAP server.

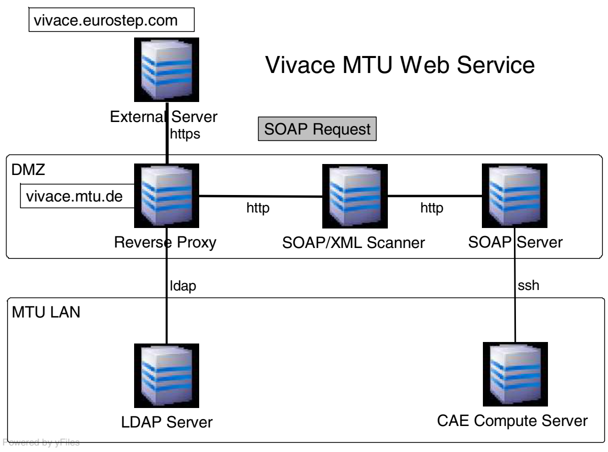

FIG 5. Security Infrastructure Concept

The SOAP server is now installed in the DeMilitarised Zone (DMZ) to reduce the vulnerability of the external accessible systems in the LAN. Special care has to be taken for the encapsulation of the compute server to prevent access of the LAN.

4. DETAILED DESIGN SINGLE DISC

For clarity of presentation, the Disc 2 detailed design process is described before Disc 1. The virtual enterprise has agreed that the detailed design process of the two High Pressure Turbine (HPT) discs is shared between two partners. This section describes the Disc 2 design analysis process with regard to collaborative engineering aspects. Based on the results from the airfoil design of the two stage turbine (turbine blades data), the basic input data for the two disc design analysis processes are provided on common data storage in Sweden (Figure 1), accessible for each collaborative partner in the virtual enterprise (Figure 6).

The basic flow of the detailed Disc 2 design process is:

1. Automated start of the local design process at partners company (controlled by the main flow),

2. Automated access to common data storage and download of design analysis input data (e.g. blade mass, number of blades, outer diameter of the disc, turbine speed),

3. Start of optimisation process with regard to the objectives functions (life cycles and disc weight),

4. Automated upload of results onto the common data storage,

5. Automated stop of local design process (signal to main flow).

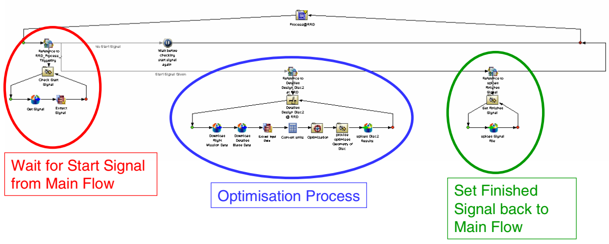

FIG 6. Partner´s design framework for multi disciplinary optimisation and robust design of Disc 2. Trigger-components are used for starting and finishing the local design process

The local design process for the collaborative engineering is shown in Figure 6. The automated start of the local process is achieved by implementing special trigger-components into the workflow. These components are linked to an internet connection with the common data storage in Sweden (Figure 1). The integrator can indicate the start of the Disc 2 detailed design process via the main flow once it is required. This is achieved by storing a start-indicator onto the common data storage in Sweden. Only a standard internet connection is used which requires no direct access of the main flow to the internal IT system of the collaborative partners company. Therefore, no proprietary rights are violated.

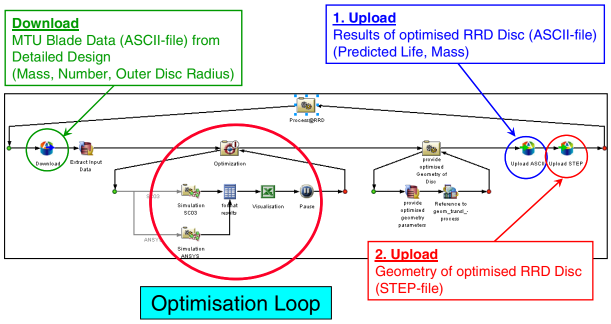

FIG 7. Local optimisation process at Disc 2 design partner. up- and download of data is automated.

In more details, the inner loop of the design process is shown in Figure 7. Once the local process is triggered, the automated download of the blade data relevant for the disc design analysis process starts.

Based on the flight mission data all other data required for the disc analysis, for example thermal- and mechanical loads as well as boundary conditions for the disc, are provided.

The final multidisciplinary optimisation process is driven by an internal loop (Figure 7). The basic work flow is shown in Figure 8. A fully parametric model of the disc will be modified every design iteration step by a CAD tool (Figure 9).

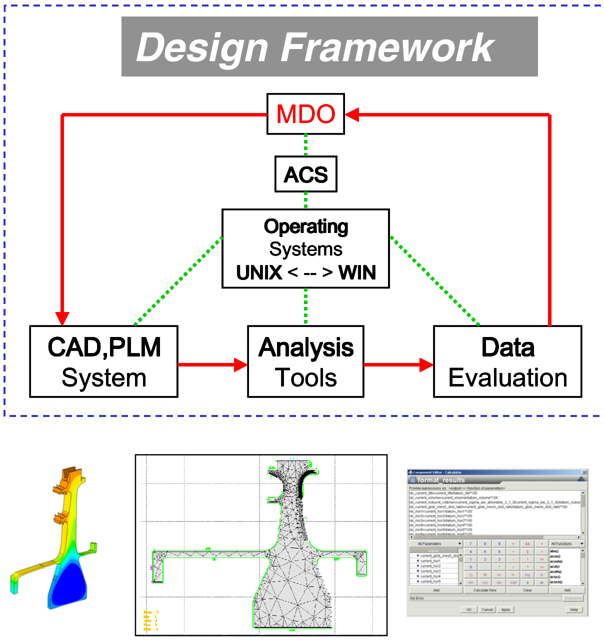

FIG 8. Design framework for multi disciplinary optimisation and robust design of Disc 2.

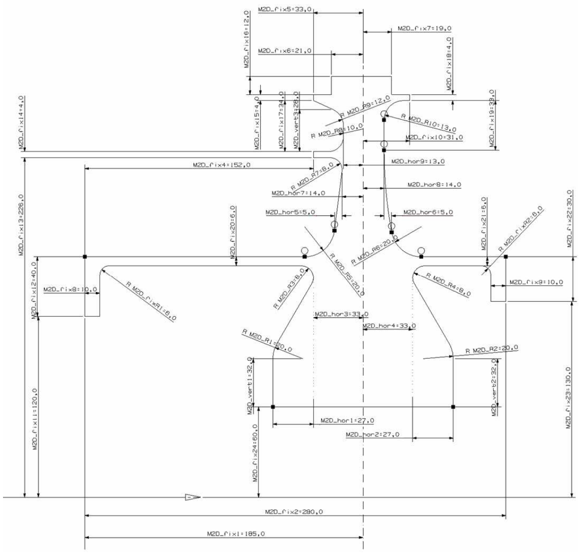

FIG 9. Parameterised two dimensional model of Disc 2

At each iteration step, the modified geometry is provided to the Finite Element (FE) analysis tool. The geometry is automatically meshed, all boundary conditions applied and the thermo-mechanical analysis is started. In the final step, the data evaluation of the results provides the objective functions data as the life cycles and the weight of the disc.

Optimisation strategies as gradient methods and genetic algorithms as well as robust design methods (Monte Carlo Simulations) are implemented into this process. In addition, surrogate modelling techniques such as the response surface method allow performing a faster analysis of the disc. Figure 10a, 10b shows some of the design space results with regard to the life and the weight (volume) of the disc. The number of relevant parameters is obtained from sensitivity studies which indicate which of the design parameters are critical to the objective functions of the disc.

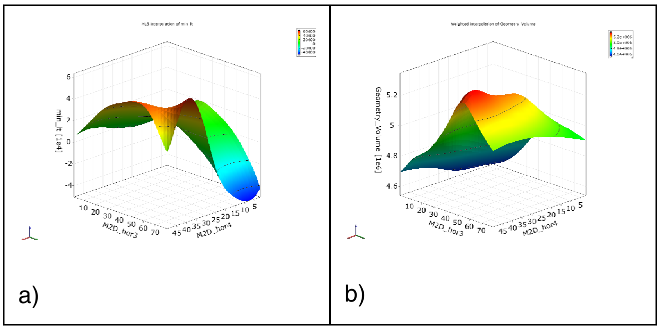

FIG 10. Design space results based on response surface methods. a) disc life, b) disc volume

Finally the results of the optimisation (weight and predicted life of the disc, the optimised geometry of the optimised disc) are automatically uploaded onto the common data storage in Sweden and therefore accessible for the integrator.

The developed design framework is flexible with regard to the design and analysis tools (commercial and proprietary software). It allows the distribution of the design and analysis process on different workstations with different operating systems (e.g. Windows, UNIX) and has a collaborative engineering capability (share of processes via a secured internet connection). Some more details are provided in [6] and [7].

5. SINGLE DISC OPTIMISATION

For the benefit of the presentation of the results, this section will describe the optimisation of only two parameters Disc 1 (X1, X2) of a single parameterised disc as depicted in Figure 11.

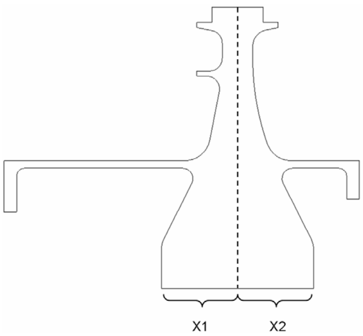

FIG 11. Selected parameters of a single disc.

For a single design, i.e. (X1, X2) value, the full multidisciplinary analysis described in the previous sections needs to be performed. As the analysis tool chain does not provide gradient information, gradient based optimisers need more computationally expensive evaluations, typically in the order of tens to a hundred. In case the design objective has a complicated behaviour in the design space considered, gradient based algorithms might not be robust and genetic algorithms may need to be referred to which typically need in the order of thousands of multidisciplinary evaluations. Meta modelling or response surface methods have been shown to be an effective solution to represent the behaviour of the objective and constraints within the design space considered with high computational efficiency and sufficient accuracy.

Using a design of experiments a sampling of the design space (X1, X2) is made with the full multidisciplinary analysis capability. Note that as these design variants are independent, their analysis could be performed in parallel, similar to the parallelisation described in the detailed airfoil design section. Subsequently a response surface is fitted through the results for the selected objective. The accuracy of the fit can be determined by for instance the "leave-one-out" method, in which a fit is made through all design results but one. The actual result of the remaining design analysis is compared with the predicted value to estimate the fit accuracy. This is done for each analysed design. The average provides an estimate of the fit accuracy. As the best fit method turns out to be sufficiently accurate for our optimisation, Figure 12 depicts the response surface for the engine disc lifing in the design space (X1, X2). The predicted disc life is expressed in the number of loading cycles. This response surface has been used for the subsequent disc mass optimisation.

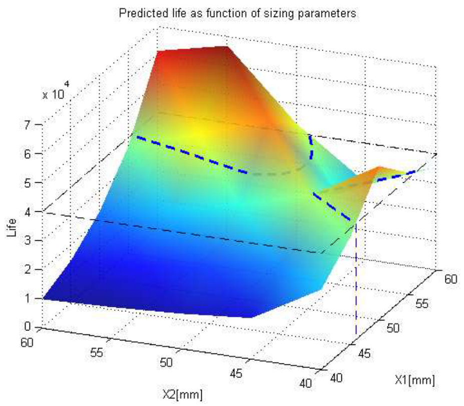

FIG 12. Calculated lifing in engine disc design space. The part above the waterline satisfies the minimum lifing constraint (40000 flight cycles).

Figure 12 shows two distinct areas where the constraint for minimum life is satisfied. This result requires advanced optimisation methods (e.g. genetic algorithms), since not all existing gradient-based optimisers can handle such constraint behaviour adequately and robustly.

The area of the disc and therefore the weight is minimal if the sizing parameters are minimal. Figure 12 shows that minimum disc mass satisfying the minimal life constraint of 40000 cycles will be reached in the right hand part with a disc design at (X1, X2) = (45, 40). The lifing of this disc design has to be verified by re-running the multidisciplinary analysis tool chain for the approximated optimal design parameters. The response surface approach to find the design optimum is computationally more efficient than combining the tool chain with a gradient based optimiser.

To illustrate the flexibility of the multidisciplinary tool chain based on the same design space sampling, the predicted disc life is maximised. Physically the life of the disc can be increased by adding mass, so the turbine disc mass is constrained. Figure 13 provides the response surface approximation of the predicted disc life with the mass constraint of 95.0 kg. The maximum predicted disc life of 53200 flight cycles is found at (X1, X2) = (55, 50), a significant 19% increase in lifing at slightly lower weight. This optimum disc design is close to the starting design of (X1, X2) = (55, 55). The original disc was obviously designed for maximum life. Still with the MDO approach a significant lifing improvement can be obtained without incurring a weight penalty. This is consistent with [4] where multidisciplinary design optimisation is considered a key technology to improve aircraft performance. A second but lower disc optimum is found around the minimum mass disc discussed above. This illustrates the importance of expert judgement when selecting the design space and constraints.

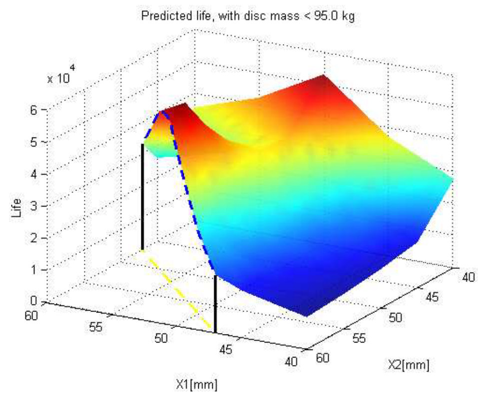

FIG 13. Calculated lifing in engine disc design space. The feasible design space satisfies the maximum disc weight constraint.

In practice, the optimal design will usually be a trade-off involving the simultaneous optimisation of more than one objective. As illustrated above, it is unlikely that a single design will be optimal for different objectives. A Pareto set contains those designs where optimising one objective will deteriorate another objective. Pareto optimisation leads to an entire curve or surface of points whose shape indicates the behaviour of the trade-off between different objectives.

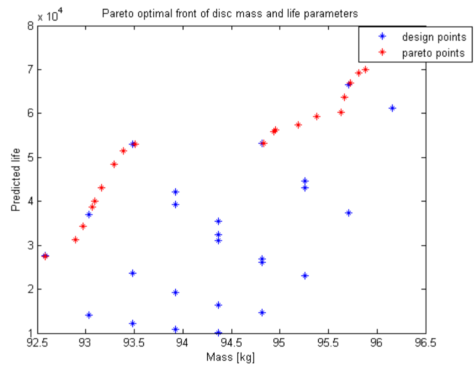

FIG 14. Pareto front (red points) of lifing and weight objectives. Note that disc designs with mass between 93.5 and 95 kg do not comply with the minimum lifing constraint of 40000 flight cycles causing a discontinuous Pareto front. The blue points depict the design space sampling.

Figure 14 shows the Pareto optimal front of the disc mass and disc lifing in objective space. It is interesting to note that disc designs with a mass between 93.5 and 95.0 kg do not comply with the minimum lifing constraint of 40000 flight cycles resulting in a discontinuous Pareto front. Following the Pareto front by increasing the mass leads to an increase of the predicted life. Any design expert in the virtual enterprise team can select from this a suitable combination of predicted disc life and corresponding disc mass. More detail on the engine disc MDO analysis and the results is provided in [3].

Like the tool chains of the previous two partners of the virtual enterprise, this tool chain has been made accessible for the entire virtual design team.

6. PROCESS INTEGRATION - THE VEC HUB

To fulfil the requirements mentioned in the section 'Introduction and Background' the concept of a Virtual-Enterprise-Collaboration-Hub, or VEC Hub[13], was developed within the VIVACE project.

This section first presents the underlying concepts and then explains the implementation as realised within this project. It enables the collaboration of the different geographically dispersed partners, performing their design work as described in the previous sections.

6.1. Concept

When initiating a Joint Venture (JV) Collaboration the JV needs to define WHO, will do WHAT and HOW. The VEC Hub concept offers a means to define and manage the (O)rganisation (WHO), the (P)roduct data (WHAT) and the (W)ork processes (HOW).

The use of services and standards based on platform independent application integration standards as Web Services Definition Language (WSDL) allows, from a technical point of view, this neutrality. A core concept to realise this functionality, without specifying details about which application/system/tool is providing it, is the use of Web Services as defined in open standards published by the W3C-Organization[14].

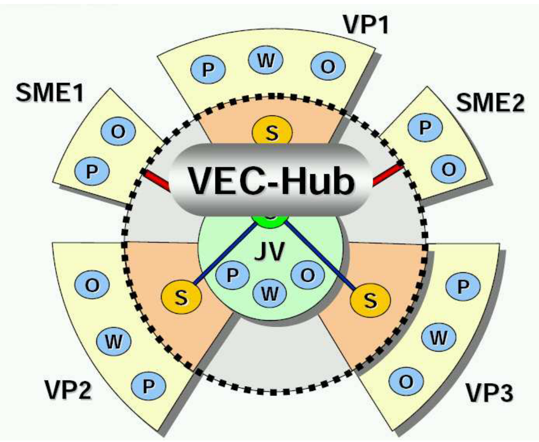

FIG 15. The VEC hub concept

In the middle of the wheel picture of Figure 15, the VEC Hub Core services are located. They provide, in a standardised way, the necessary services to enable the partners to share product data (P in Figure 15), organizational data (O) and processes, i.e. workflows (W).

In the terminology of the VEC Hub, the design activities from the Virtual Partners (VPs) the airfoil designer, the Disc 1 detailed designer and the Disc 2 detailed designer are Partner Supplied Services. They are identified with the "S" in the spokes of the wheel in Figure 15. The VEC Hub provides the means to enable partners to share workflows, running in their environment and still control the access and usage of those services. As for the core services the partner supplied services are described and accessed in a standardised way.

By relying on standards in the workflow description, as well as in the product data management, the necessary flexibility can be guaranteed that allows for easy addition or removal of partners, if the design process evolves and changes in the organizational structure of the VE become necessary.

In such a distributed, collaborative environment, as the VEC Hub provides, security issues are extremely important, for authentication, authorization, transport, data storage and access. Moreover the security requirements of each VP's IT security policies are different and the VEC Hub provides a means to handle them. For example, the authentication step for accessing the partner provided services can be handled either by a traditional user ID/password combination (that is stored encrypted within the VEC Hub core) or with a public/private key infrastructure.

Some of the VP's IT infrastructures might not even permit any outside access to internal computer networks. For these cases the VEC Hub provides a special polling mechanism that starts internal processes based on VEC Hub events that the partner workflows poll at prescribed intervals.

Any data parameter and process that resides within the VEC Hub core is protected with a fine grained set of access control rules. Traceability and repeatability of the design process is guaranteed by versioning any information stored within and any process executed at the VEC Hub. If the design task requires it, this versioning can go down to individual parameter level for each execution of an optimization loop.

6.2. Implementation

The implementation of the VEC Hub that is used for this example is based on the commercial software[15] that provides the product data and reference data management facilities and COTS[16] that provides the functionality to handle the processes and the organizational structure.

Since due to space limitations, this paper is only concerned with the detailed engine design process, the overall process that consists of the preliminary design process is not discussed here. But this preliminary design is captured in the same workflow. The master workflow that controls both design phases is executed by the integrator at the VEC Hub.

The detailed design workflow then runs as follows; after the preliminary design is finished, airfoil designer retrieves the necessary information from the VEC Hub storage to start its detailed blade design process.

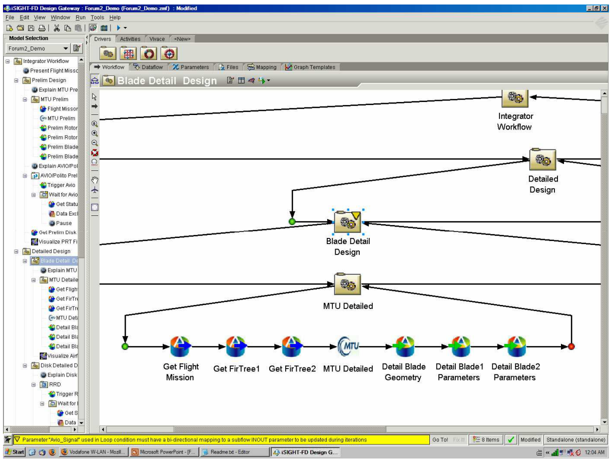

FIG 16. Detailed airfoil design

Figure 16 shows this process. The components "Get Flight Mission", "Get Fir TreeX", "Detail Blade Geometry" and "Detail BladeX Parameters" are VEC Hub core services that enable the automated communication with the VEC Hub. The icon "MTU" represents the blade design process executed by that virtual enterprise partner. This, and the necessary input and output parameters, is all that the VPs and the integrator see from the internal processes. No additional information needs to be exposed and all VP´s intellectual property is guaranteed.

After the design of the blades is fixed, the necessary information on mass and centre of gravity can be provided to the companies that design the discs. Their design work can be performed in parallel.

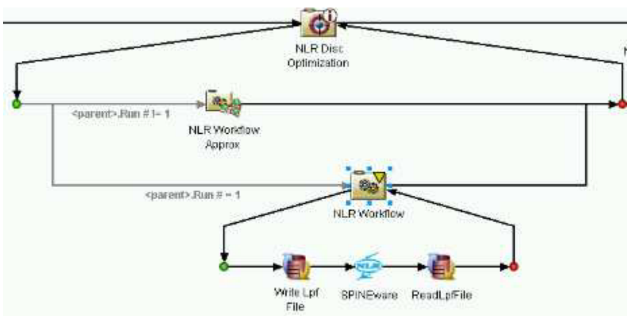

FIG 17. Detailed Disc 1 design

In this example the Disc 1 design partner doesn't use internal optimisation methods but the VEC Hub provided an optimisation loop around the partner services, see Figure 17. The input file for the NLR internal workflow is automatically prepared at the VEC Hub and then sent as a parameter to the partner's web service which is also supported by COTS[17].

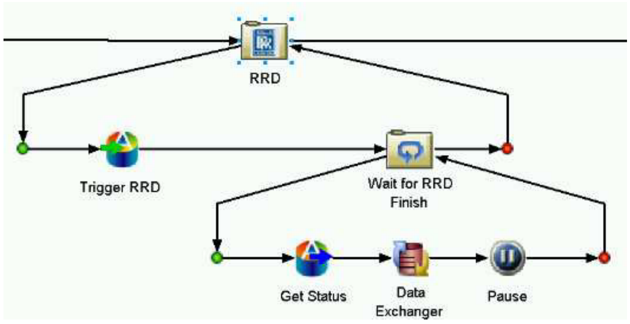

FIG 18. Detailed Disc 2 design

The disk design workflow at the Disc 2 design partner depicted in Figure 18, presents a special case, since external access and triggering of internal workflows can not be implemented within their IT environment. Therefore the polling mechanism mentioned in the previous subsection is used here.

The component "Trigger RRD" updates a file at the VEC Hub, with the information that all necessary information for the Disc 2 design partner to start its design process is available. The internal workflow at that partner queries this file in specified intervals and executes the internal design task only if this file gives the start signal. This internal workflow uses the API interface at the VEC Hub to retrieve and upload the necessary data automatically.

At the VEC Hub a similar loop then executes at the same time, the "Wait for RRD Finish". It makes a query to a different file that is updated once the Disc 2 workflow has finished and only continues the master workflow if it finds the required information there. With this, the detailed design process is finished and all data that the VPs want to share is stored at the VEC Hub and available to subsequent steps and design review meetings.

7. CONCLUSIONS

Based on the organisational requirements common in aircraft engine programmes a solution has been implemented which allows many distributed partners, without one dominant partner who can prescribe the tool suite to be used, to collaborate in the design of a critical engine component. This is a typical example of a virtual enterprise tailored to the needs of the aeronautical community. The resulting integrated design capability supports the flow of technical design information without infringing the intellectual property of the collaborating partners. The chosen realisation of the tool chain allows flexible addition or modification partners and/or proprietary tool suites. The fully automated design capability efficiently supports the numerous design changes of the evolving engine design, obviating the need for time and effort consuming paper-based engineering change processes. In this way the design capability provides power to all engineers in the collaboration. To our knowledge this is the most comprehensive multinational collaborative engine design environment at this level of integration.

The fully automated airfoil design process has been successfully implemented and subsequently been made available to the virtual enterprise through web services. In addition a security concept for these services has been demonstrated.

The next partner has implemented an automated process for the Disc 2 analysis and design based on the airfoil design information. This tool chain is flexible with respect to the tools included

The multidisciplinary design optimisation approach, demonstrated significant improvement of the Disc 1 design with respect to one objective. This is a typical result for applying multidisciplinary design optimisation with respect to a baseline design.

Based on the design capabilities provided as web services the next partner in the virtual enterprise was able to create a workflow encompassing all three designs in a consistent way.

ACRONYMS

CAD Computer Aided Design

CAE Computer Aided Engineering

COTS Commercial Off The Shelf Tools

DMZ DeMilitarised Zone

FE Finite Elements

JV Joint Venture

LDAP Lightweight Directory Access Protocol

MTU MTU Aero Engines

NLR National Aerospace Laboratory

RRD Rolls Royce Germany

SOAP Simple Object Access Protocol

VE Virtual-Enterprise

VEC Hub Virtual-Enterprise-Collaboration-Hub

VP Virtual Partners

WSDL Web Service Definition Language

XML eXtensible Markup Language

ACKNOWLEDGEMENT

This study is partly performed in the frame of the VIVACE integrated project, which is partly sponsored by the Sixth Framework Programme of the European Community (2002-2006) under priority 4 “Aeronautics and Space” as integrated project AIP3 CT-2003-502917.

8. REFERENCES

[1] Chalmeta R., Granger R., *ARDIN extension for virtual enterprise integration*, Journal of systems and Software, (67) 3, 2003, p 141-152.
[2] Parchem R., Meissner B., Arendsen P., Schultheiss B., Nagel M., Barner J., Lindeblad M., Wänman F., Demonstration of Collaborative Engineering Distributed Whole Engine Design Process. In *VIVACE Forum 2* CD-ROM, Den Haag, the Netherlands, 24-26 October 2006. Also published as NLR-TP-2007-525.
[3] Kesseler E., van Houten M.H., Multidisciplinary Optimisation of a Turbine Disc in a Virtual Engine Environment, 2nd European Conference for Aerospace Sciences (EUCASS), Brussels, Belgium, July 2007. Also published as NLR-TP-2007-526.
[4] Chaput E., Design challenges in the Airbus A380 and A350 projects. Key note *ECCOMAS CFD*, Egmond aan Zee, the Netherlands, 5-8 September 2006.
[5] Nagel M., Baier R.-D., Experimentally Verified Numerical Optimisatiuon of a 3D-Parametrised Turbine Vane with Non-Axisymmetric End Walls, ASME Turbo Expo GT-2003-38624, Atlanta. USA, 2003
[6] Parchem R., Meissner B., Multidisciplinary Design and Optimisation of a Whole Engine Component, *VIVACE Forum 1* CD-Rom, Warwick, UK, 22-24 October 2005
[7] Lewis G., Parchem R., Armstrong C., Karl A., Distributed Design Task using the Collaboration Hub, *VIVACE Forum 1* CD-Rom, Warwick, UK, 20-21 September 2005
[8] Alexiou A., Baalbergen E., Kogenhop O., Mathioudakis K., Arendsen P. PROOSIS: Advanced gas turbine engine performance simulations. In *ASME Turobo Expo 2007*, Montreal, Canada, 14-17 May 2007. Also published as NLR-TP-2007-513.
[9] Visser W.P.J., Kogenhop O., Oostveen M. A generic approach for gas turbine adaptive modelling. In *ASME Turbo Expo 2004*, Vienna, Austria, 14-17 June 2004. Also published as NLR-TP-2004-391.
[10] Anonymous, ISIGHT, Engineous, http://www.engineous.com/index.htm, accessed 18 June 2007
[11] Anonymous, Parasolid, http://www.ugs.com/products/open/parasolid/ , accessed 18 June 2007
[12] Anonymous, Uni-Graphics, http://www.ugs.co.uk/, accessed 18 June 2007
[13] Anonymous, VIVACE Project, http://www.vivaceproject.com/content/advanced/reports.php, section “Collaboration Hub for Heterogeneous Enterprise”, accessed 6 July 2007
[14] Anonymous, World Wide Web consortium, http://www.w3.org/, accessed 18 June 2007
[15] Anonymous, Share-a-Space, http://www.share-a-space.com/, accessed 18 June 2007
[16] Anonymous, FIPER, http://www.engineous.com/product_FIPER.htm, accessed 18 June 2007
[17] Anonymous, SPINEware, http://www.spineware.nl/index.html, accessed 18 June 2007