SIMDAT

Data Grids for Process and Product Development using Numerical Simulation and Knowledge Discovery
Project no.: 511438

Grid-based Systems for solving complex problems – IST Call 2
Integrated project

SIMDAT

Deliverable

D17.1.1 Specification of the A-DIS toolset
D17.1.2 Initial version of the application scenario, demonstrating connectivity and the operation of the model problem

Start date of project: 1 September 2004 Duration: 48 months

Due date of deliverable: 1 September 2005
Actual submission date: 14 October 2005

Lead contractor for this deliverable: BAE Systems
Revision: 1.0

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
	Dissemination level	
PU	Public	X
PP	Restricted to other programme participant (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Revision history

Date	**Version**	**Author**	**Modification**
2005-08-16	Draft 1	Mike Boniface Stephen Phillips	Superceeds draft deliverable D16.1.1 v02 Includes additional questions regarding EADS workflow
2005-09-09		Stephen Leary	Further info on aero scenario and on aerodynamic service added.
2005-09-09		Wenbin Song	Minor changes
2005-09-20		Mike Boniface	Accepted all previous changes Elaborated Freefluo description Portal description Deployment description Development schedule description.
2005-09-20		Stephen Leary	Introduction Conclusions Minor changes to other sections.
2005-09-27		Wenbin Song	Combine Build RSM and Search RSM into one service "Build and Search RSM" Add reference to OPTIONS
2005-09-28	Final version	Stephen Leary	Minor changes

Copyright

Table of contents

1 Introduction

This document is a report that describes the initial version of the application scenario, demonstrating connectivity and the operation of the model problem in the aerospace application activity, as specified in SIMDAT Annex 1- “Description of Work” [1].

The aerospace industry deals with highly complex products that have data creation, management and curation requirements that span hundreds of collaborating organisations over a 50-year lifecycle. Partners on a product team need to collectively manage thousands of inter-related processes and this leads us to expend considerable time and effort in the access, transmission, control, translation and sharing of data.

The primary focus of the aerospace activity is the development and deployment of existing and emerging Grid technologies and concepts to improve the way we do collaborative engineering of sophisticated products. The improvement in our ability to handle complex problems is not delivered simply through the *connectivity* that Grid offers, but in the deployment of middleware and advanced ontology-based techniques to radically improve the efficiency of the data exchange – between applications and between organisations.

The research programme in the aerospace application sector is built around a use-case selected from many possible alternatives in the product lifecycle. We have chosen to simulate the multi-disciplinary collaborative configuration design of a low-noise, high-lift landing system. The scenario is typical of sub-system design problems in the context of, say, future-concept, unmanned cargo vehicles that require an ability to use airfields in noise-sensitive locations. *However, it is important to note that this is a “model problem” designed to drive a research programme that is focussed on the development and deployment of Grid technology.*

The scenario is built and developed to stress the *aggregation* of *distributed capabilities* operating *across organisational boundaries*.

This document reports advances made up to project month twelve, a major target by this stage was to demonstrate connectivity of services spanning organisational boundaries. The scenario is described in greater depth and the process by which a multidisciplinary design problem was attacked by disparate teams making use of grid technologies is discussed.

1.1 Workflow design process

Developing a rigorous workflow definition is critical for understanding how applications can be deployed on the Grid. Workflow specifications allow us to identify exactly the resources (processes, data, agreements, physical machines, databases) required by the application, who owns the resources, where these resources are deployed and how they are shared between organisations.

Our approach for converging on a complete workflow specification is based on an iterative design process that moves from a conceptual application workflow that contains high-level descriptions of process and data to an infrastructure workflow that details exactly the resources required by the application. In order to organize and formalize the description of the aerospace workflow, we define the following steps:

1. Application workflow: this is a high-level conceptual description of the workflow consisting of a set of tasks and the required data that are used as input/output from these tasks.
2. Intermediate workflow: this is an intermediate step where the workflow consists of both conceptual tasks and services and is useful when parts of the workflow are not specified yet (application part) while other are defined (infrastructure part).
3. Infrastructure workflow: this is the low-level description of the application workflow consisting of the actual service orchestration and data flow that is used at the application implementation phase. There are several notations to describe this, but here for simplification purposes we can use the similar notation with the application workflow. The difference is that instead of tasks we‘ll represent a sequence of services and a data flow between them.

Activity diagrams in UML with some minor simplifications are our preferred graphical notations for representing workflows. This section provides a brief description of the notation that we hope is sufficient to understand the workflow diagram. The main diagrammatic constructs are shown in Figure 1.

Figure 1: Activity diagram constructs for representing workflow

In addition to these constructs, the workflow makes use of connectors to simplify the diagram. Figure 2 illustrates the use of a connector. The connector is represented by a circle and is named “A”. It is used to break the dataflow from activity1 to activity2 so that the data flow does not have to cross other constructs in the diagram.

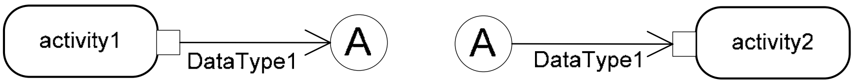


Figure 2: A connector in a workflow diagram.

Workflow diagrams may also make use of “swim lanes” to indicate the location of actions and data (See Figure 3).

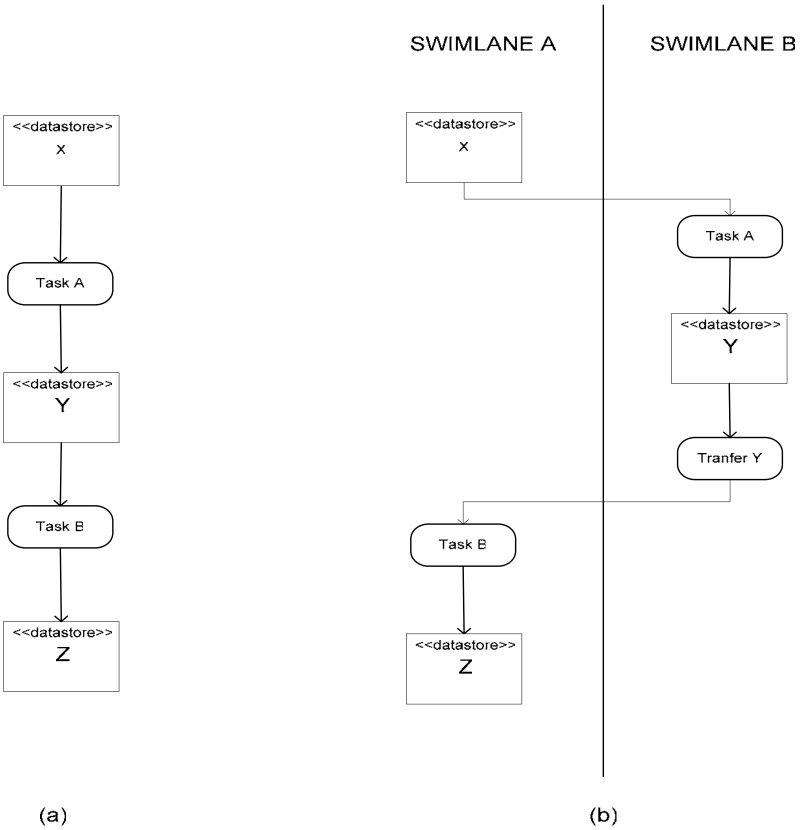


Figure 3: Workflow representing resource ownership with swim lanes

2 Aerospace Design Scenario Description

2.1 Design workflow

The challenge of the design service is to optimise the design of an aeroplane wing within a set of constraints by varying a set of parameters that describe the wing's geometry.

The design service provides a model of a wing with many variable parameters that, when set, fully define the wing's geometry. For a given set of geometric parameters and additional parameters such as the angle of attack, the lift, drag and noise of the wing may be calculated. The design service must algorithmically choose sets of geometric parameters to find a wing geometry that achieves some optimum design, where the quality of a design is measured according to the lift, drag and noise. At this stage a subset of the geometric variable parameters (relating to the position of the flap) will be altered. These variables are shown in Figure 4.

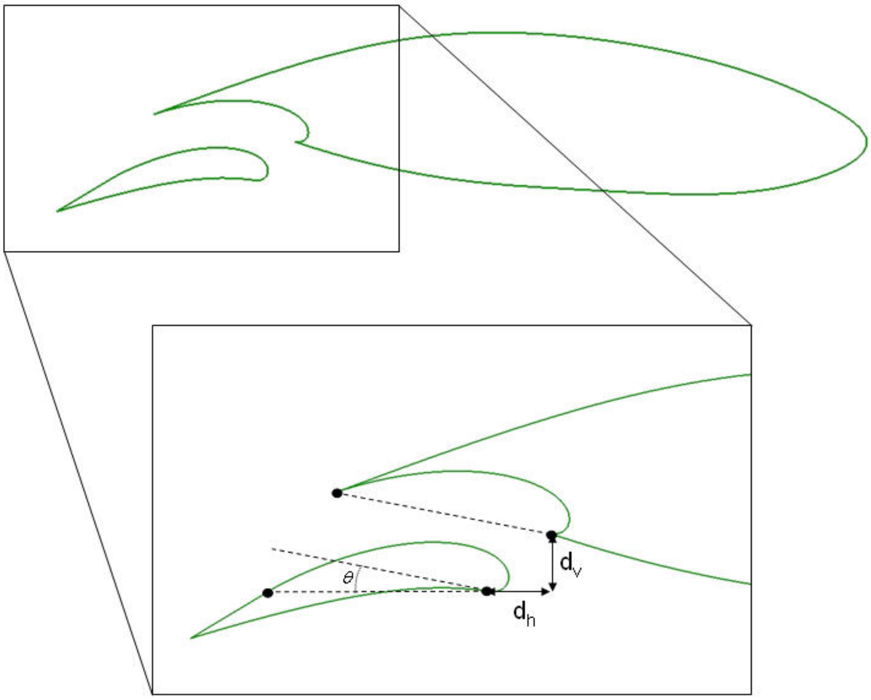


Figure 4: The variables (d_h – horizontal displacement, d_v – vertical displacement, θ - rotation) that describe the flap's position.

The major input that the engineer using the design service will provide is a set of constraints for the geometric parameters. A "best current set" may also be provided along with potentially some other constraints.

The major output of the design service will be the optimum set of parameters that satisfies the constraints. This may involve the use of different strategies include direct optimisation and response surface based optimisation.

In the course of executing design workflow, the service will make use of aerodynamics and aero-acoustics services to calculate the lift, drag and noise values for each wing design. The overall process is shown in Figure 5.

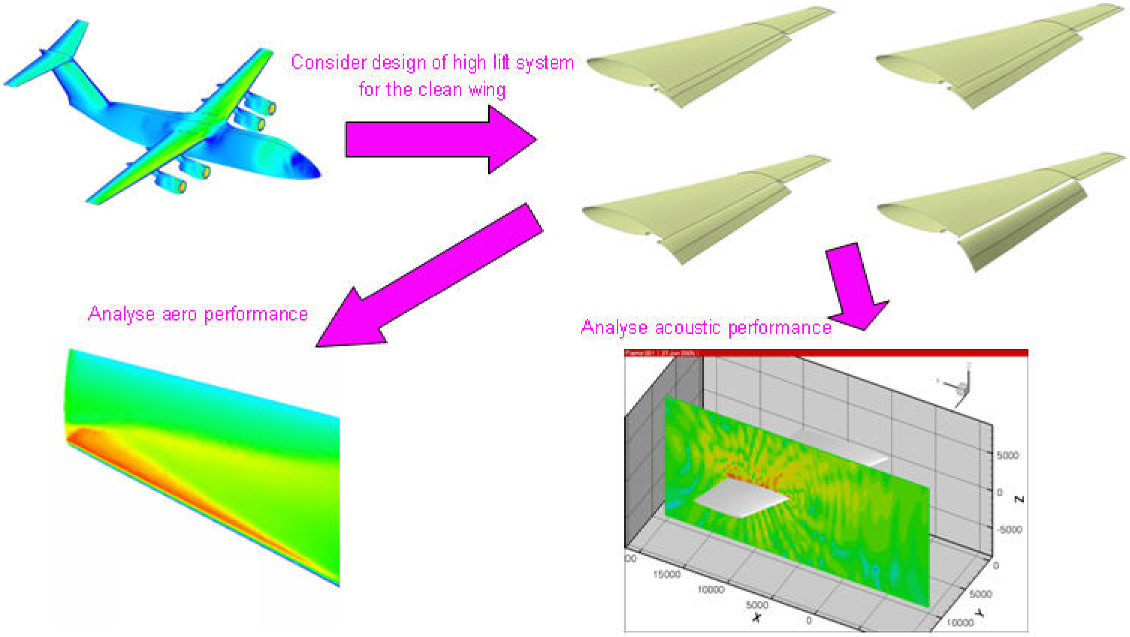


Figure 5: The overall process: a wing parameterised by the CAD service is manipulated in the design service and the aerodynamic and aero-acoustic performance of each candidate wing are calculated using the aerodynamics and aero-acoustics services respectively.

2.1.1 Optimisation process

An optimisation algorithm in general has three inputs:

1. one or more target functions that must be minimised (or maximised),
2. a set of parameters that may be varied,
3. a set of constraints.

The constraints may constrain the variable parameters directly by specifying a range of values each variable may take, and they may also constrain functions of both the variables and the target function.

The optimisation process outlined in the previous section is not easy to implement in a cost-effective way for two reasons:

1. Each set of parameters, henceforth referred to as a “design point”, that define the problem (including the wing’s geometry and attack angle) is large (approximately 30 variables).
2. To calculate the lift, drag and noise for just one design point may take as much as one day.

Therefore, the algorithm for finding the optimum design must minimise the number of design points considered whilst trying to search a very large space.

In this case there are three properties calculated for each design point: the lift, the drag and the noise. There are therefore several options for choosing the target function:

1. the designer may wish to minimise a function of lift, drag and noise, weighting the three components appropriately;
2. one of the three properties could be chosen as the target function and constraints placed on the other two;
3. a multi-objective problem using all three values could be formulated.

Before discussing the detail of the optimisation algorithm it will be useful to describe a simpler example that we may refer to in order to illustrate the process. In this simple example we will consider a wing design where all the geometry is fixed apart from two variables: the span of the wing and the sweep angle (see 6).

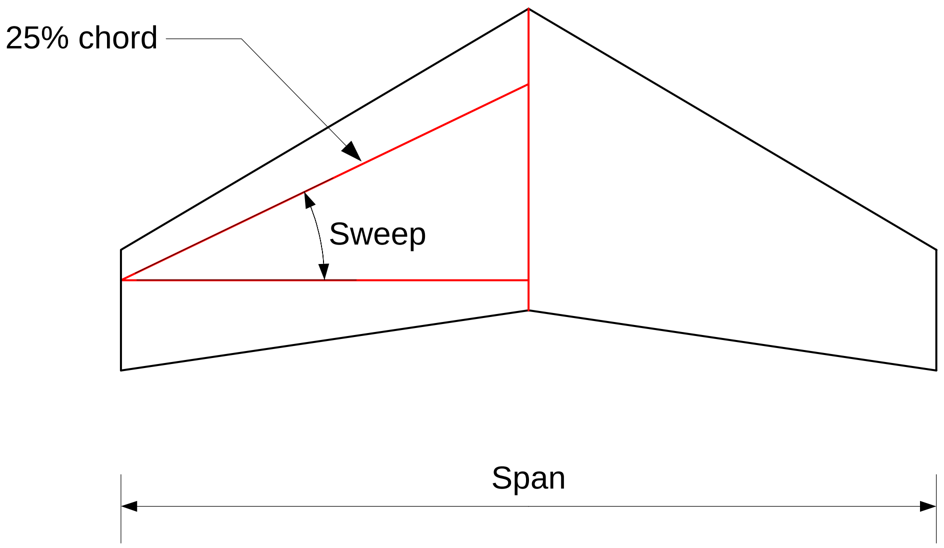


Figure 6: How the wing span and sweep are defined

We will assume that the target function is to maximise the lift while ignoring the drag and noise and that the wing span and sweep are constrained to be between 10 and 30m and 0 and 45° respectively. The optimisation algorithm chosen has two distinct stages. In the first stage an approximation of the surface representing the design variables and target function is calculated. In the second stage, the focus is on the optimum value in the surface, and the prediction of the optimum is repeatedly refined.

First a set of design points is chosen to maximise the sampling of the design space, this is known as the "design of experiments" or DOE. As a rule of thumb, the size of the set of design points should be ten times the number of variables. In our simple example this is a choice of twenty points in the plain (see Figure 7), but in the real case this would be a set of approximately 300 design points, each of which is a rank 30 vector. The DOE algorithm aims to choose a set of design points to cover the design space as efficiently as possible.

The service calculates the lift, drag and noise at each design point (either in parallel or serially). Once this calculation is completed a surface may be fitted to the points, known as the RSM. This surface fitting is an optimisation in itself as there are many surfaces that pass through a set of points but the required surface is in some way the smoothest. The fitted surface provides an estimate of the target function at any design point.

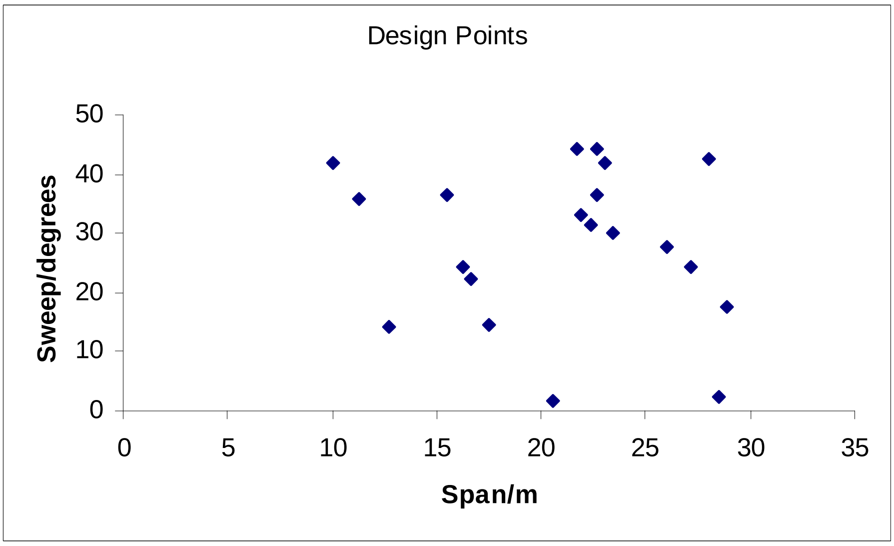


Figure 7: An example DOE

Once we have the surface we search it for the predicted optimum design point. So, going back to our example, we would search the surface for the maximum lift. This design point (length/sweep pair) is then the current best prediction of the optimum design. Figure 8 shows the surface for our simple example from which the point of maximum lift can be determined. We could stop at this point, but if time and money permit, the second stage of the process may be invoked.

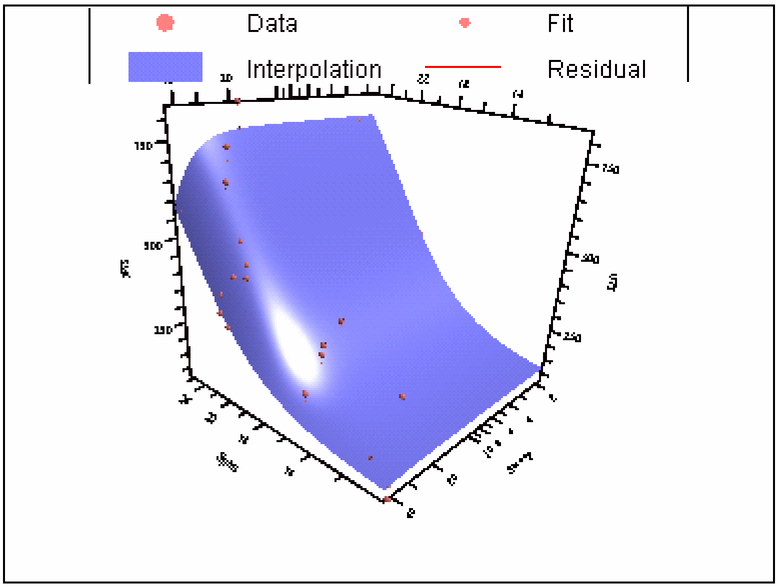


Figure 8: An example surface plot showing the predicted optimum point

The second distinct stage of the optimisation algorithm takes the best-guess design point as its input. The value of the target function at this predicted optimum is also a prediction, so to refine the solution the target function for the best-guess design point is calculated and added to the results set. We now have extra information about the surface in the location of the optimum design so the surface is recalculated and searched and the process repeated until some convergence criterion or time limit is reached.

2.1.2 Application workflow

The design service will have several distinct workflows. The optimisation workflow has been broken into the two stages referred to in Section 2.1.1 and is shown in Figure 9 and Figure 10. There are two additional services that will be provided and these are diagrammed in Figure 11.

For PM12, the first stage of the optimisation workflow was implemented. The second stage (refinement of the optimum point) will be completed after PM12.

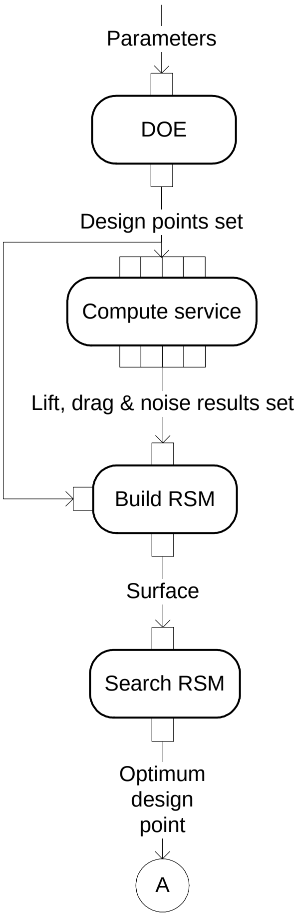


Figure 9: First stage of the optimisation workflow

Figure 9 shows the first stage of the optimisation process. The "DOE", "Build and Search RSM" tasks are all performed by the OPTIONS [2] package.

The inputs to the DOE calculation are as follows:

- The choice of DOE algorithm.
- The number of design points required.
- The upper and lower bounds of the parameters.
- A "current best set" of parameters (optional).

The output of the DOE calculation is a set of design points for which the lift, drag and noise are to be calculated. The lift, drag and noise for a design point is calculated by a service shown here as "Compute service" defined in Section 2.2. The icon for the "Compute service" has a row of four pins above and below it. These represent concurrent iteration over the input set, and the output of all the individual results as a single set.

The "Build and Search RSM" task takes the DOE set and results set as inputs and carries out the user-defined search sequences to find the predicted optimum design point which may be fed into the second optimisation stage shown in Figure 10.

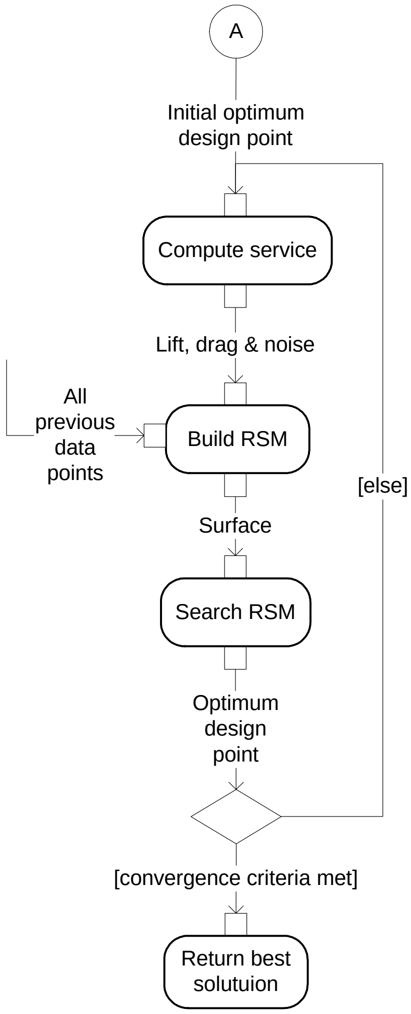


Figure 10: Second stage of the optimisation workflow

As described in Section 2.1.1, the current best prediction of the optimum design point is fed back into the “Compute service” to calculate the exact lift, drag and noise at that point. This new data is added to the existing set of design points and results and a new surface is fitted to the data and searched by the “Build and Search RSM” task to find the optimum design point on the new surface. This process is repeated until the optimum design point has converged on a value or until the client runs out of time or money.

There are two further services that may be provided by the design service that operate semi-independently of the optimisation process. The services are shown in Figure 11, and are “Evaluate point” and “Plot surface”. Both of these services operate on the RSM which will be computed from the available data. They may be used once the DOE has been calculated but do not have to wait for all the lift, drag and noise data to be computed.

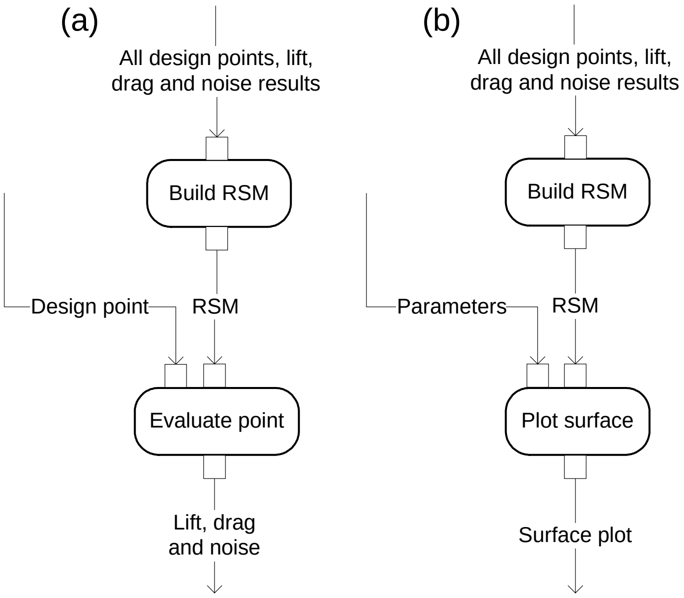


Figure 11: Two services returning data to the engineer

The “Evaluate point” service (Figure 11a) will find an approximate value of the target function for given design points. This is done precisely by the “Compute service”, but is done quickly in this case by using the fitted surface. In other words, an approximation to the real value is found by interpolating the results already calculated.

The “Plot surface” service (Figure 11b) takes the RSM and some user-defined parameters and outputs a graphic of the surface. In real design scenarios this may not be very useful as the number of dimensions in the problem is so large. It is not clear what software would be used to generate the graphics. It is probably more useful to produce predicted values for the objective function for a matrix of design points. The data produced can then be visualised using other visualisation tools.

It may also be necessary to provide a means for the engineer to instruct the design service to compute a specific design point during the optimisation process. This can be done by carrying out a single evaluation process without invoking any of the optimisation methods.

2.2 Compute service workflow

The compute service may be discussed as a separate workflow to the design service. The input to the compute service is a design point and the output is the lift, drag and noise data for the design point. Many compute service workflows may be run in parallel.

The basic steps of the compute service workflow are the following:

1. Invoke CAD service.
2. Perform aerodynamics (CFD) workflow.
3. Perform aero-acoustics workflow.

The CAD service constructs a description of the geometry of the wing. It does this by combining the design point provided with a “parametric geometry”. A parametric geometry is a geometrical description of the wing with some dimensions (e.g. lengths, angles) undefined. Combining this template with the design point information produces a fully-specified geometry instance. The parametric geometry will be supplied by the CAD service and will not be chosen by the client.

The geometry instance is used as input to the aerodynamics and aero-acoustics services (Sections 2.3 and 2.4) to produce lift, drag and noise as outputs.

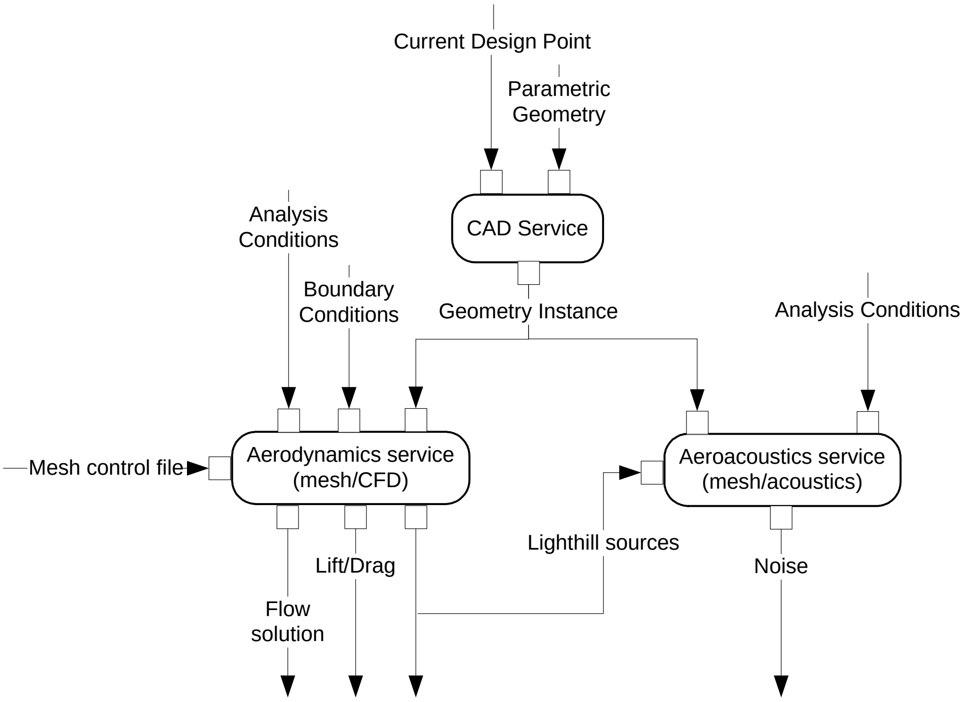


Figure 12: Compute service application workflow

2.3 Aerodynamics workflow

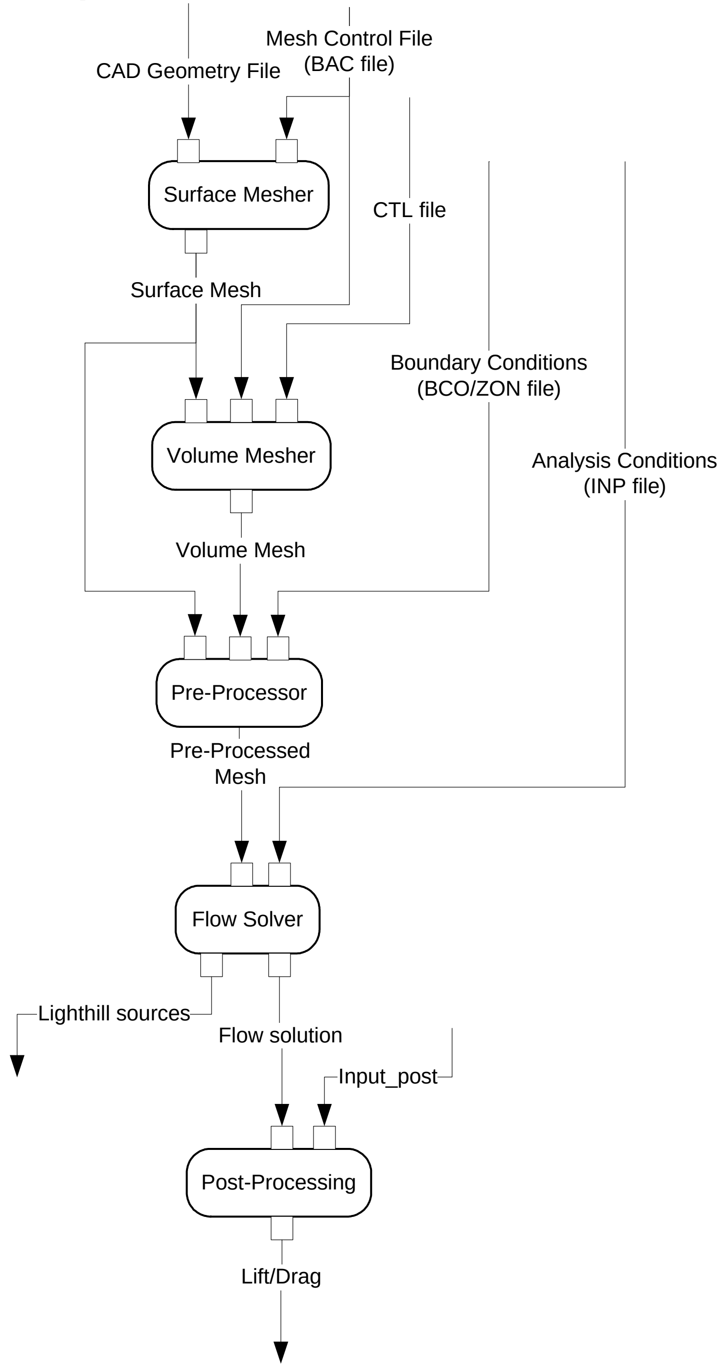


Figure 13: Aerodynamics application workflow

The Aerodynamics workflow performs a CFD (Computational Fluid Dynamics) analysis on the aircraft geometry and the corresponding mesh files in order to extract the flow solution that contains useful information about the forces on the aircraft.

The basic idea behind CFD is simple. The flow around an aircraft reacts to the shape of the outer skin and it contains vortices, shock waves, sheets of highly sheared fluid, the engine exhaust, etc. We cannot simply model the behaviour of the fluid in the whole volume around the aircraft, so we split the volume into lots of smaller volumes (meshing step). The flow-field within each sub-volume is much simpler than the entire flow-field. Then we use a flow solver to obtain a flow solution for each sub-volume. Specifically, having obtained a flow solution, the user is presented with the flow at every point in the grid. This is a vast amount of data and not very useful on its own. The last phase of the CFD process is to extract from these data all the information that the user actually wants, e.g. the forces on the aircraft such as lift and drag.

Using CFD is a process with three stages:

1. Pre-processing (Input files, “Surface Mesher”, “Volume Mesher”, “Pre-Processor” in Figure 13)
    a) Representing objects in the flow domain, e.g. the aircraft surface, using CAD tools; For this step, we use the CAD Geometry file as an input as well as the Mesh Control file (BAC).
    b) Generating a “grid” or “mesh”, i.e. splitting the flow domain into sub-volumes; For this step, we use the “Surface Mesher”, the “Volume Mesher” and a pre-processor to obtain the pre-processed mesh that will in turn be used as input to the next step.
2. Obtaining a Flow-Solution (“Flow Solver” in Figure 13)
    a) Running a flow-solver using the grid for flow conditions specified by the user;
3. Post-Processing (“Post-Processing” Figure 13)
    a) Extracting and visualizing the flow data from the results of the flow-solver. Specifically, we extract the data that are useful to the user like the forces on the aircraft (lift and drag).

For the implementation of the aerodynamics service, the SOLAR CFD system will be used. This system is a BAE in-house Navier-Stokes CFD suite.

The automation of such a workflow provides several challenges. The first relates to the number of geometry file formats in use in computational engineering design throughout the world today. Conversion from one format to another is non-trivial. This is an issue that directly affects the aerodynamics workflow as the output from the CAD service is an IGES file whilst SOLAR expects a BAE SYSTEMS in house format DAT file for geometrical input. An IGES to DAT converter does exist but this is not robust enough to cope with complex changes to the geometry. Automated geometry conversion is a real issue when virtual organisations utilising different geometry formats are set up. However, given the design parameters shown in Figure 4, DAT files can successfully be created.

An alternative to geometry conversion is to mesh directly inside a CAD tool/service. As an aside we note that in this work this would overcome the requirement for a BAC file since our CAD based meshing routine uses surface curvature in order to define mesh density so a mesh control file is not required. However, the BAE SYSTEMS CAD based mesher uses commercial CAD software. As with the CAD service (see later) there are licensing issues in utilising such software as part of our service, as part of the CAD tools API would need to be exposed to external users in a virtual organisation setting. This breaks current licensing agreements and is another significant issue when setting up such organisations.

The second challenge relates to mesh generation itself. A fully automatic mesh generator that can cope with radically different geometries is still some way off. Even such simple changes as moving a flap can cause meshing to fail. Our meshers are fairly robust to the sorts of changes made using the variables shown in Figure 4. However, we note that this may not be the case when more complex design changes are considered. Therefore it is imperative that the system copes gracefully with run failures.

2.4 Aero-acoustics workflow

The aero-acoustics workflow performs a CAA (Computational Aero-Acoustics) analysis.

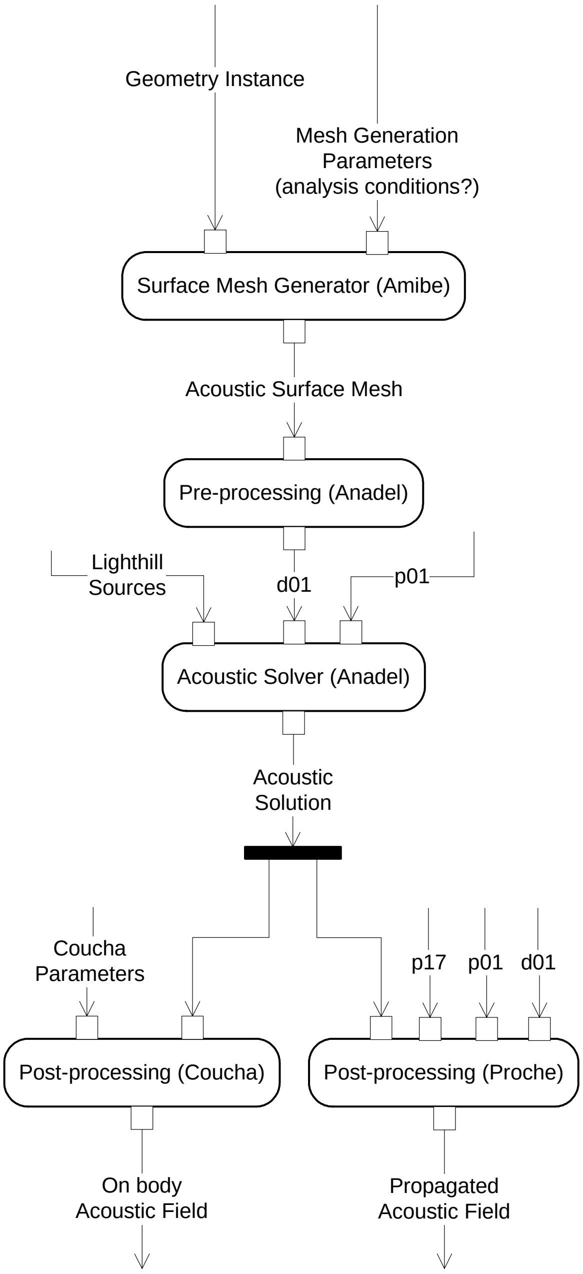


Figure 14: Aero-acoustics application workflow

The idea behind this workflow is to extract some information about the noise generated by the flow computed by the CFD analysis. The starting points of this process are:

1. The boundary representation of the object potentially diffracting the acoustic wave,
2. The Lighthill sources that are the basic model representing the elementary pressure source,
3. Since the sounds perceived by the human ear vary between 20 Hz and 20 KHz approximately; one has to provide the frequency of interest.
4. Finally, one has also to express where the noise level has to be computed. This information can be seen as the microphone position with respect to the object.

Obtaining the noise level is a three steps process:

1. Pre-processing (Input files, “Surface Mesh Generator (amibe)”, “Pre-Processor (anadel)” in Figure 14)
    a) Representing objects diffracting the acoustic waves, e.g. the aircraft surface, using CAD tools;

    For this step, we use the CAD Geometry file as an input as well as some parameters qualifying the mesh (cell size, defection parameter).
   b) Generating a mesh i.e. splitting the diffracting objects into elementary faces;
      For this step, we use the “Surface Mesher”, to obtain the set of degrees of freedom that represent the physical values to be computed. This will in turn be used as input to the next step.
2. Obtaining an acoustic solution (“Acoustic Solver (Actipole)” in Figure 14) by running an acoustic solver on the surface mesh and using the Lighthill sources as initial boundary conditions,
3. Post-Processing (“Post-Processing (Coucha) or Post-Processing (Proche)” Figure 14)
   a) This step is computing the noise data at the specified location from the results of the acoustic-solver.
      Specifically, the data that are useful to the user are computed like the pressure at a specified 3D point (usually at the ground level)

For the implementation of the aero-acoustic service, the ELFIPOLE CAA system will be used. This system is an EADS in-house acoustic solver based on the multipole method.

There are many challenges in running such a process:

- The first one has already been mentioned and relates to CAD interoperability. There are many different CAD standards that can be interpreted differently by each CAD tools. Many such problems have been encountered while reading some IGES files into our amibe mesher that was written by CATIA.
- The second one is the capability to mesh an object for an acoustic simulation. Such an analysis deals with wave propagation phenomena which require a mesh having at least 5 points per wavelength (this is the smaller number of point for “accurately” represent a sine). Therefore an object having a surface of 100 m^2 (which is a small object) will require at least 17 million elements which is quite a challenge (for example the total wing surface of the A380 is about 385 m^2)
- The final challenge is then to be able to run the analysis on such a number of degrees of freedom overnight.

2.5 Data descriptions

Table 1 is an incomplete description of the data files and types used in the entire workflow. The service that the data appears in is denoted by:

- De: design service
- C: compute service
- Dy: aerodynamics service
- A: aero-acoustics service

Identifier	**Service**	**Description**	**Format**	**Estimated size**
Design point	De, C	A set of parameters fully specifying a wing geometry and environment	Proprietary	KB
RSM	De	A surface fitted to the results of the computation for the purpose of estimating additional values	Proprietary	MB
Surface plot	De	A plot showing a view of an RSM		
Parametric geometry	C	A model of the wing with some variable geometrical parameters	CATIA part file	Up to 100 MB
Geometry instance	C, Dy, A	A fully defined geometry, created by the CAD service by combining a	IGES	Up to 100 MB

		design point with the parametric geometry		(average ~25 MB)
Analysis conditions (INP)	Dy	Solver inputs (e.g. Mach number, angle of attack, Reynolds number)	INP (ASCII)	1MB
Boundary conditions	Dy	Boundary conditions (should be constant)	BCO/ZON (ASCII)	2MB
Mesh control	Dy	Mesh spacing control (curvature based meshing would reduce the complexity of creating this)	BAC (ASCII)	10KB
CTL file	Dy	Input to the volume mesher	CTL (ASCII)	2KB
Surface mesh	Dy	Defines the mesh on the wing	XDR	50MB
Volume mesh	Dy	Defines the mesh in the air	XDR	500MB
Pre-processed mesh	Dy	Input to the flow solver	XDR	500MB
Flow solution	Dy	A flow solver output	XDR	500MB to 10GB
Input post	Dy	Input to the post-processor	ASCII	10KB
Lift/drag	Dy, De	Aerodynamic forces	ASCII	20KB
Lighthill sources	Dy, A	Location (tensors) of noise sources on airframe, as generated by CFD Flow Solver. This feeds into the aero-acoustics workflow	ASCII	200MB
Mesh generation parameters	A	Parameters for controlling the mesh generation (e.g. element size, deflection parameter, optimization iteration)	ASCII	Few words.
Acoustic surface mesh	A	Model of component surface (N.B. different to that used for aerodynamics).	ASCII/UNV	May be up to 5 GB
d01	A	Contains the degree(s) of Freedom (pressure jump/normal velocity)	ASCII	
p01 / acoustic solver parameters	A	Input parameters for the acoustic solver (e.g. frequency, filename for Lighthill sources)	ASCII	< 1kB
Acoustic solution	A	Results from Acoustic Solver.	Binary (.res)	16 octets/dof
Coucha parameters	A	Parameters to control Coucha post-processing (e.g. mainly the type of result)	ASCII	
p17	A	Parameters to control Proche post-processing (e.g. mainly the type of results and the support (mesh)).	ASCII	< 1kB
On-body acoustic field	A, De	Pressure and/or normal velocity	ASCII	Several MB
Propagated acoustic field	A, De	Pressure and/or normal velocity	ASCII/UNV	Several MB

Table 1: Design scenario data descriptions

3 Aerospace PM12 Architecture

This section describes the technologies that are deployed in the aerospace sector for the PM12 prototype. Figure 15 shows the overall architecture for the aerospace scenario. Each organisation within the aerospace virtual organisation operates as a GRIA service provider offering engineering services that form part of the overall aerospace application including an initial optimisation function (UoS), aerodynamics (BAE) and aero-acoustics (EADS).

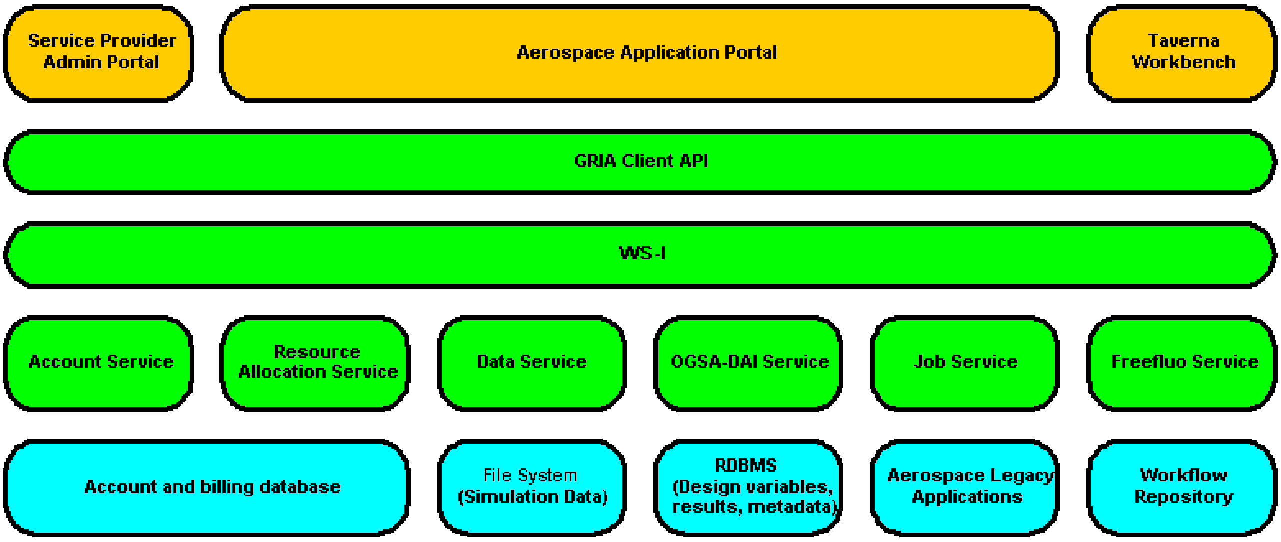


Figure 15: Aerospace scenario technology architecture

Each service provider hosts six core services:

- **Account service** records resource usage for billing purposes
- **Resource allocation service** manages and assigns computation and data resources hosted by the service provider
- **Data service** stores simulation data files
- **OGSA-DAI service** provides access to database resources for storing metadata
- **Job service** executes legacy applications
- **Freefluo service** executes application workflows containing legacy jobs and post processing tasks for writing metadata to OGSA-DAI

Three user interfaces will be provided including:

- A **service provider administration portal** that allows service providers to manage resources such as accounts, data and applications
- **Taverna workbench** allows expert engineers to develop application workflows and deploy them to a Freefluo service
- **Aerospace application portal** provides a business-focused interface to the aerospace design workflow.

3.1 Grid infrastructure

GRIA has been deployed as the Grid infrastructure to support the aerospace application scenario. GRIA's capability to support dynamic virtual organisations allows a project manager to create a multidisciplinary design team where distributed engineers working in different organisations can participate in the design and development of a low-noise/high-lift landing system.

Requirement	Description
WS-I compliant services	Standards compliant service interface
Identity Management	Ability to create and manage user digital identities CA, Revocation lists
Authentication	Ability to authenticate users (X509 based) Authenticated transactions Ability to handle federated identity for exploitation phase
Authorisation	Policy driven access control to resources Dynamic policy management for exploitation phase
Access to legacy applications	Service container for legacy applications including submission access to compute service
Data transfer/access service	Transfer/access of flat files Database access including schema publishing in exploitation phase Policy based access control
Single interface to compute service	Ability to access compute services with different scheduling requirements through single interface Reservation of compute resources Sandboxing runtime in compute service including ability to specify sandbox environment
Resource discovery	Ability to discover alternate services due to service failure or unavailability

Table 2: Aerospace Grid infrastructure requirements

Table 2 shows the Grid infrastructure requirements as defined in D15.1.1 and D15.1.2 [3]. Clearly, the security of engineering data is an essential requirement for the aerospace scenario. Users must be authenticated with X509 credentials, data must remain confidential when transferred between different organisations, and access to resources must be to authorised users only. The aerospace activity operates a public key infrastructure using OpenCA for the certificate authority run by BAE. The PKI provides an X509 trust domain allowing clients and service providers to be authenticated. GRIA satisfies the aerospace security requirements in the following ways:

- X509 token manager allows the integration on X509 credentials and trusted certificate chains
- Mutual authentication of client and service provider provided by TLS
- Message confidentiality provided by TLS
- Message integrity provided by WS-Security
- Authorisation and delegation is provided by Process Based Access Control

The aerospace application codes such as CATIA and Options (Section 4.3) have been integrated as legacy codes within the GRIA job service providing a file-compute execution model.

3.2 Distributed data access

OGSA-DAI WS-I has been integrated with GRIA to provide a solution for distributed data access for relational data and simulation files [4]. OGSA-DAI WS-I is used to store metadata with references to data files stored by GRIA data services. The integration of OGSA-DAI WS-I with the GRIA service container enforces a business model appropriate to the aerospace application scenario by:

- Allowing service providers to account and bill for database resource usage
- Allowing clients to manage the lifecycle of database resources
- Leveraging the inherent security mechanisms provided by the GRIA infrastructure to authenticate users, control access to database resources and ensure confidentiality and integrity of messages

A database schema has been developed for the aerospace application that is used to store design of experiments along with the resulting lift, noise and drag for each design point and sufficient metadata to determine how the results were derived. This schema is shown in Figure 16.

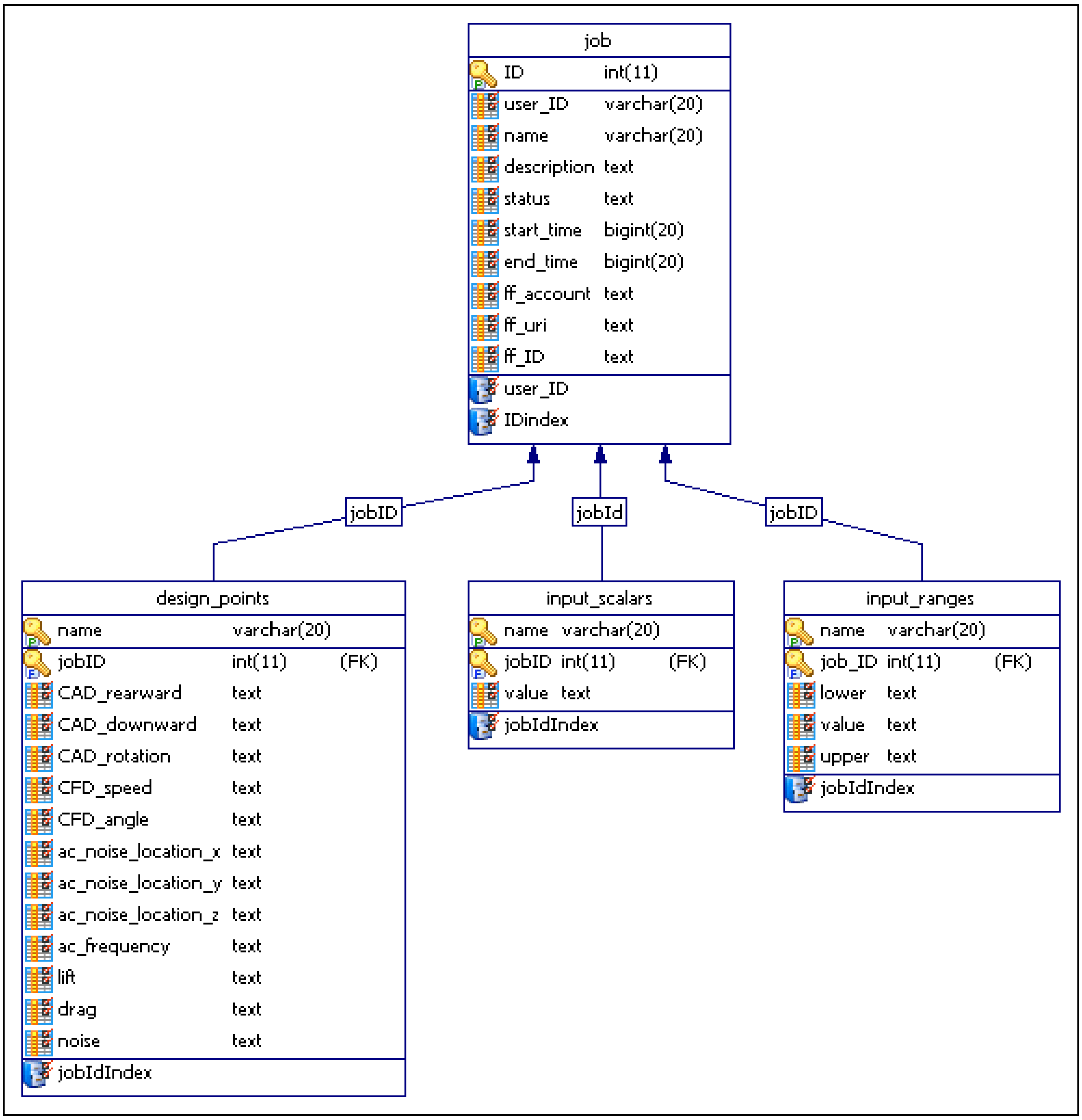


Figure 16: Database schema

3.3 Workflow

Taverna/Freefluo has been integrated with GRIA to provide workflow composition and enactment. Expert engineers have developed the design, compute, aerodynamics and aero-acoustics workflows using the Taverna workbench and then published the workflow to a Freefluo GRIA service deployed at a service provider. Clients can then execute the deployed workflow like any other application provided by a service provider. Figure 17 shows how a client (BAE) with an account at the Design service provider can execute deployed workflows that access other distributed resources maintained. The concept is to support a subcontractor relationship between service providers. In this case BAE subcontracts to the design service (CEDG) and the design service subcontracts to the aero-acoustics service (EADS). Each service provider updates client accounts with resource usage. The liability propagates up the subcontractor chain until the final cost of the job appears in the client's account.

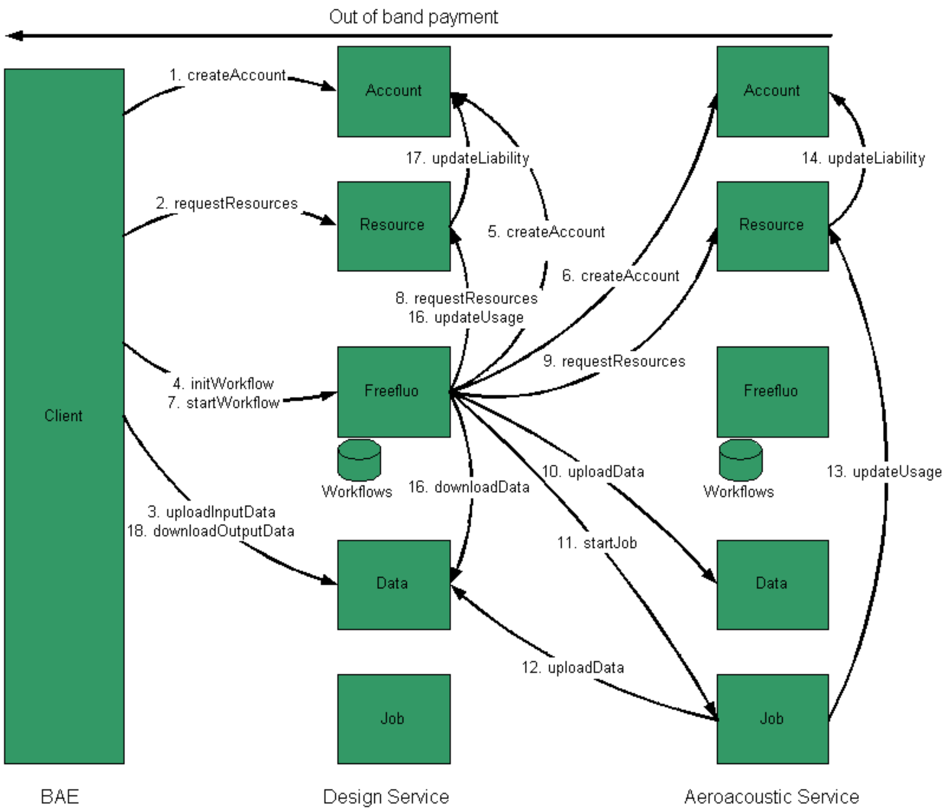


Figure 17: Federated workflow

3.4 Aerospace application portal

A portal has been developed that allows engineers to execute the design workflow and access the simulation result data. The portal has been deployed within the GridSphere portal framework. The purpose of the Portal is to provide engineers with a basic interface for executing the design workflow using some constrained parameters. The aerospace activity decided that the Portal would have a short life-time, as other domain specific interfaces such as ModelCenter would become more appropriate following the PM18 point.

4 Development and Deployment

This section describes the infrastructure and application developments undertaken to deliver the aerospace PM12 prototype.

4.1 Deployment

Figure 18 shows the overall deployment of the aerospace prototype. The deployment demonstrates how the Grid infrastructure technologies can support pan-European inter-Enterprise collaborative development of complex products.

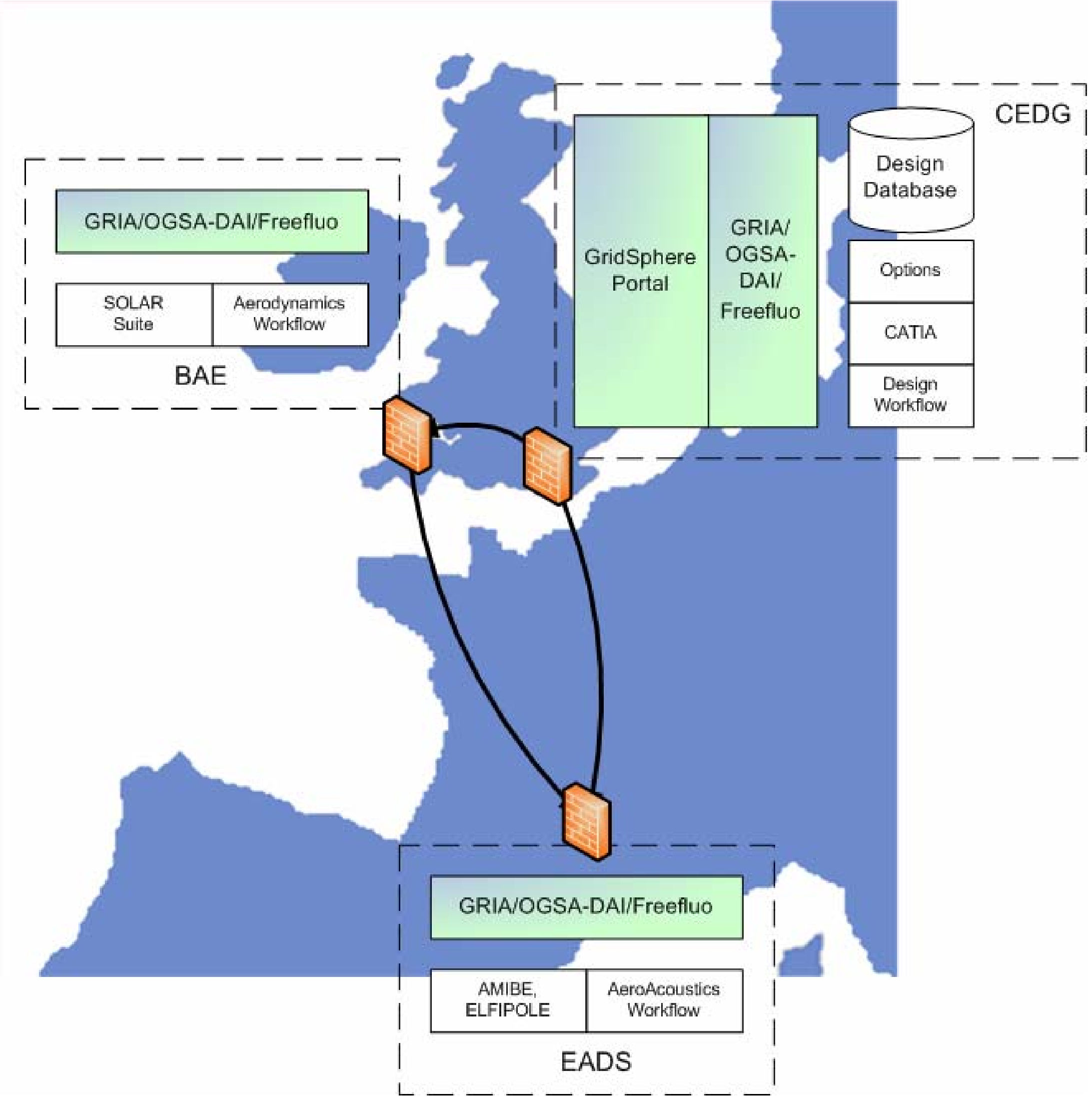


Figure 18: Aerospace deployment

The Grid infrastructure is deployed at three sites

Site	Description
CEDG, University of Southampton, UK	Provides the access point to the aerospace application. Provides the design workflow, design database and portal along with associated Optimisation and CAD services.
BAE, Filton, UK	Provides the aerodynamics capability through integration of the BAE SOLAR suite and development/deployment of associated workflow
EADS, Toulouse, France	Providers the aero-acoustics capability through the integration of the EADS AMBIE/EFIPOLE solvers and development/deployment of associated workflow

4.2 Development and deployment schedule

The delivery of aerospace prototype required inter-organisation development and deployment of both infrastructure technology and aerospace applications. The aerospace prototype covered many of the SIMDAT technology layers including grid infrastructure, distributed data access, virtual organisations, workflow and analysis services. The prototype did not address ontologies and knowledge services, which will be covered during the next project phase.

The development schedule consisted of many parallel tasks and we needed to ensure that the release schedule enabled the team to work most efficiently. The key challenge was that the infrastructure software for distributed data access and workflow was not available so we had to adopt a staged deployment approach. Deployment of the technology was seen as critical to the success of the prototype due to previous experience of Grid deployments that cross organisation boundaries. Therefore, the strategy was to deliver new components regularly to the aerospace partners that increased the level of functionality but also were deployable within the real-world context.

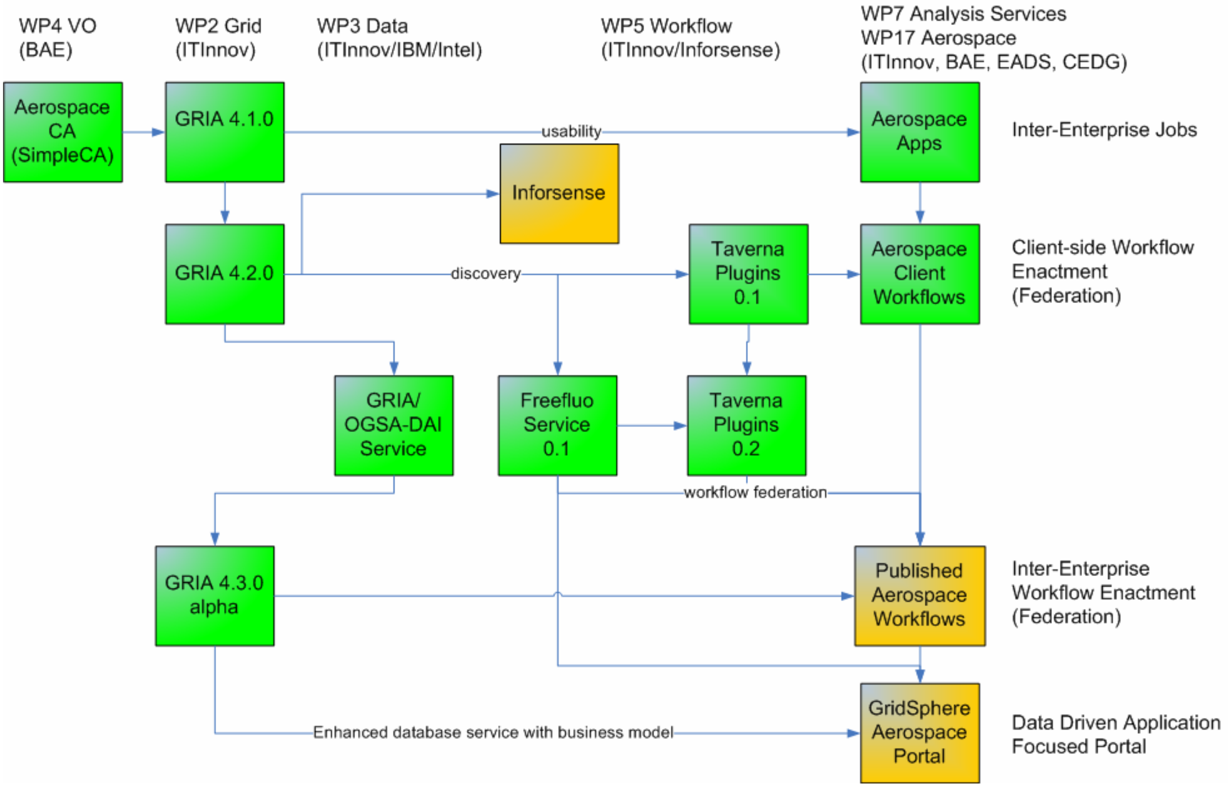


Figure 19: Aerospace prototype infrastructure development schedule

Figure 19 shows the developments contributing to the aerospace prototype. The key tasks are described below:

- BAE set up a public key infrastructure to support issuing of security credentials within the aerospace activity.
- All aerospace partners deployed the first release of the basic grid infrastructure GRIA 4.1.0. The purpose of this release was to improve the usability of GRIA whilst enabling the aerospace partners to start the integration of legacy applications (SOLAR, Options, ATIA) and inter-enterprise testing of jobs.
- Concurrently the technology work packages continued development of distributed data access components (OGSA-DAI), workflow technologies (Taverna, Freefluo, InforSense) and portal (GridSphere)
- All aerospace partners deployed GRIA 4.2.0 with additional information services to support discovery of GRIA applications required by workflow clients. GRIA Taverna plug-ins released and installed allowing the aerospace partners to

development workflows based on GRIA jobs and test the aerospace workflows using Taverna's client-side enactment.

- All aerospace partners deployed the GRIA Freefluo service enabling the workflows to be deployed and advertised as services to other clients. This enabled federated workflows to be composed and hence the development of a design workflow that called the aerodynamics and aero-acoustics workflows between enterprises.
- GRIA 4.3.0 alpha and GridSphere portal was deployed in Southampton to provide a simple data driven portal for the aerospace application.

4.3 Analysis Services

Identifier	**Description**	**Organisation**	**Application**	**Platform**
DOE	Design of experiments	CEDG	OPTIONS	Windows/Linux
Build RSM	Fit a surface to the results data	CEDG	OPTIONS	Windows/Linux
Search RSM	Find the optimal point on an RSM	CEDG	OPTIONS	Windows/Linux
Evaluate point	Evaluate a point on the RSM	CEDG	OPTIONS	Windows/Linux
Plot RSM	Plot an RSM	CEDG	OPTIONS MATLAB	Windows/Linux
CAD	CAD service	CEDG	CATIA	Windows
Aerodynamics services	The aerodynamics service is made up of a surface meshing service, a volume meshing service, a mesh pre-processing service, a flow solution service and a post-processing service. The service evaluates the flow field over the geometry supplied by the CAD service.	BAE	SOLAR CFD	UNIX
Aero-acoustics services	The aero-acoustics service is made up of a surface meshing service, a mesh pre-processing service, an acoustic solver and a post-processing service. The service calculates the noise associated with the geometry supplied by the CAD service, making use of flow field information supplied by the aerodynamics service.	EADS	AMIBE, ELFIPOLE	UNIX, Windows

Table 3: Analysis services

4.4 Infrastructure workflows

The descriptions of the design service infrastructure workflows presented here are a first draft and not meant to be complete. Each workflow is followed by a list of unresolved issues.

4.4.1 Design infrastructure workflow

These infrastructure workflows make explicit use of GRIA data stagers and of databases. The databases would be accessed using the GRIA/OGSA-DAI service currently being written. The database is used to store the design points and the results (lift, drag and noise). It is expected that the database will be queried by e.g. the “Plot RSM” service while the optimisation calculation is in progress.

Figure 20 uses the full form of the iteration notation introduced in Figure 9. The dashed box represents a workflow that will be executed for each of the design points. It is not specified in this figure whether the execution is sequential or parallel.

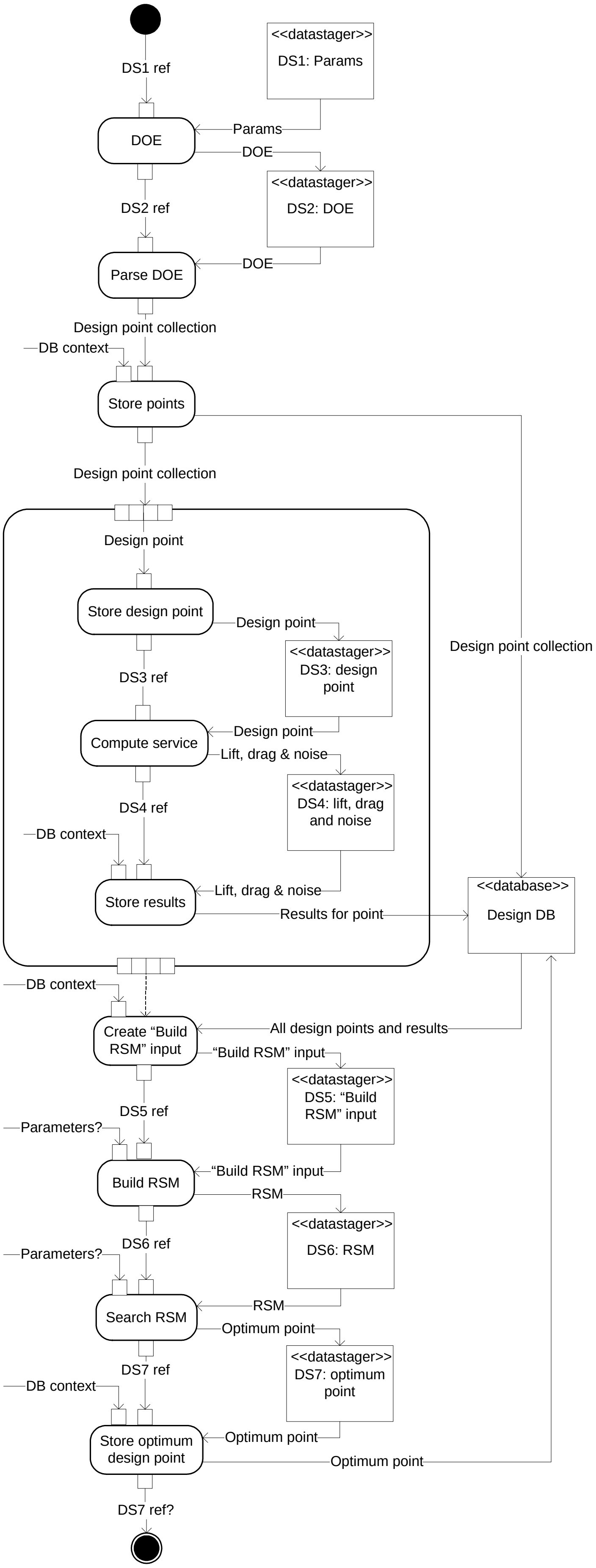


Figure 20: Design service infrastructure workflow part 1

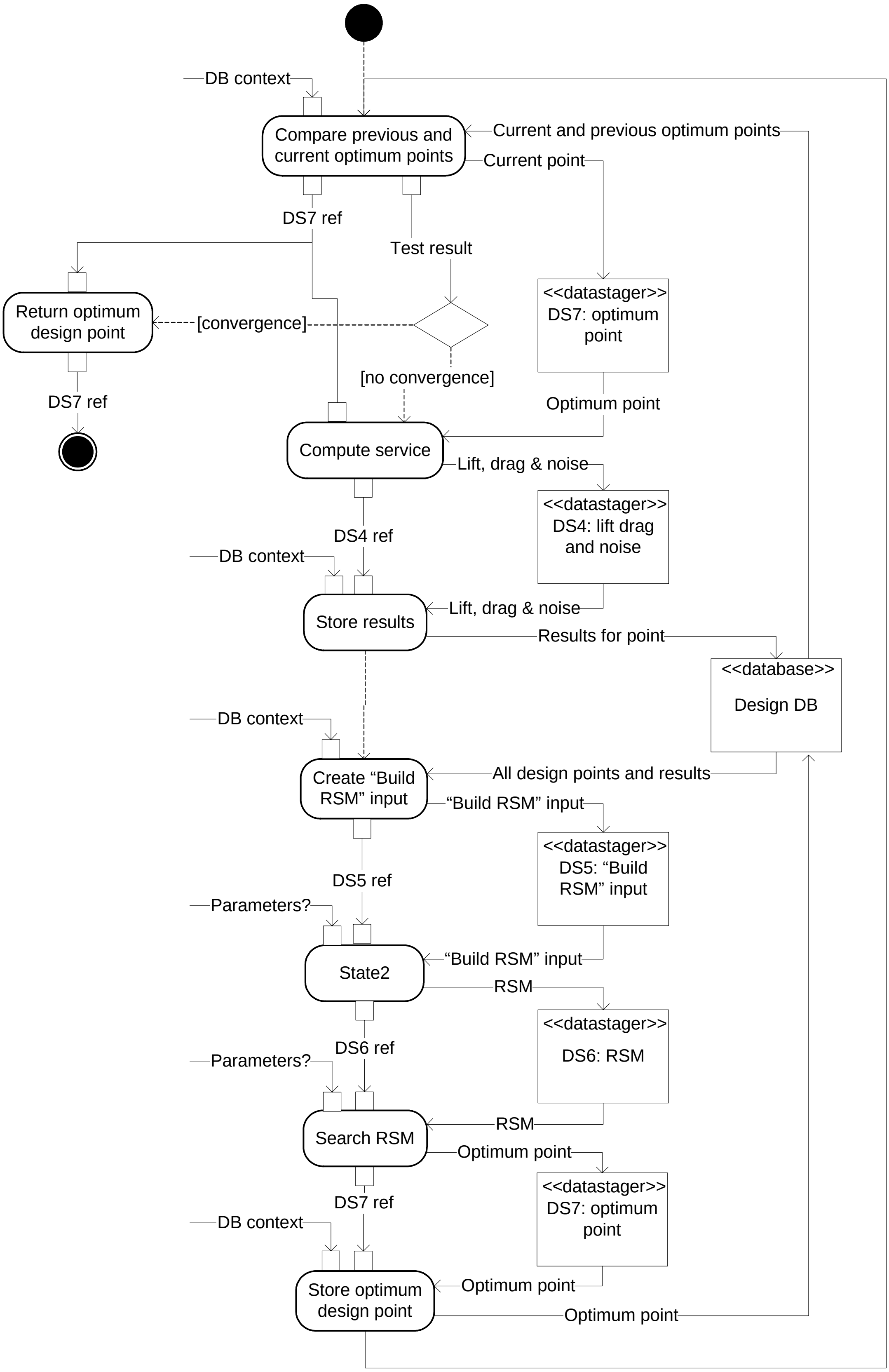


Figure 21: Design service infrastructure workflow part 2

4.4.1.1 *Issues*

There are various unresolved issues. These include:

- How should be iteration be performed?

  - There are two iteration points: converting design points to geometry instances and running the geometry through the aero-acoustics and aerodynamics services.
  - Ideally the compute service should run all jobs concurrently and leave the aero-acoustics and aerodynamics services to handle queuing issues.
- It is not clear whether the two stages of optimisation are really separate or how we would control the number of iterations of stage 2.
- There must be some database context (e.g. experiment ID) for the actions that access the database. How are we going to pass around database context?

4.4.2 Compute service infrastructure workflow

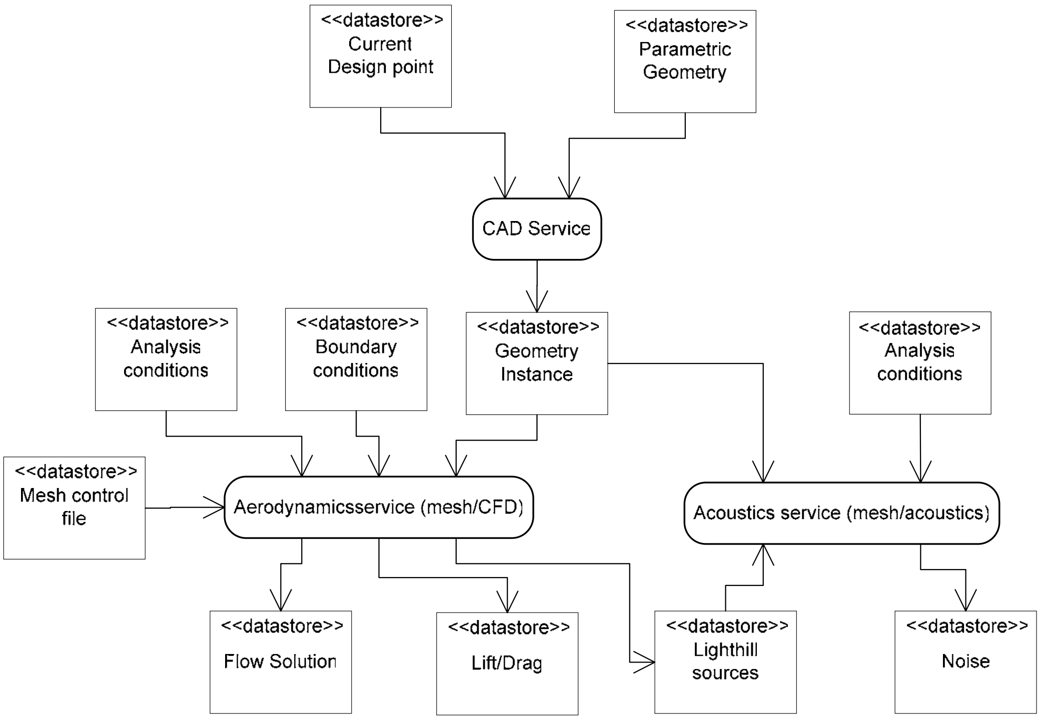


Figure 22: Compute service infrastructure workflow

4.4.2.1 *Issues*

- ➢ The CAD service is driven by CATIA V5. CATIA is a commercial product and there are implications in exposing its functionality as a service: only licensed users (Southampton University staff) can currently access the service.
- ➢ Deployment of the CAD service is problematic since the software used resides within several firewalls. This will be addressed by purchasing a dedicated machine and software license that will reside outside the CEDG and University firewall.

4.4.3 Aerodynamics infrastructure workflow

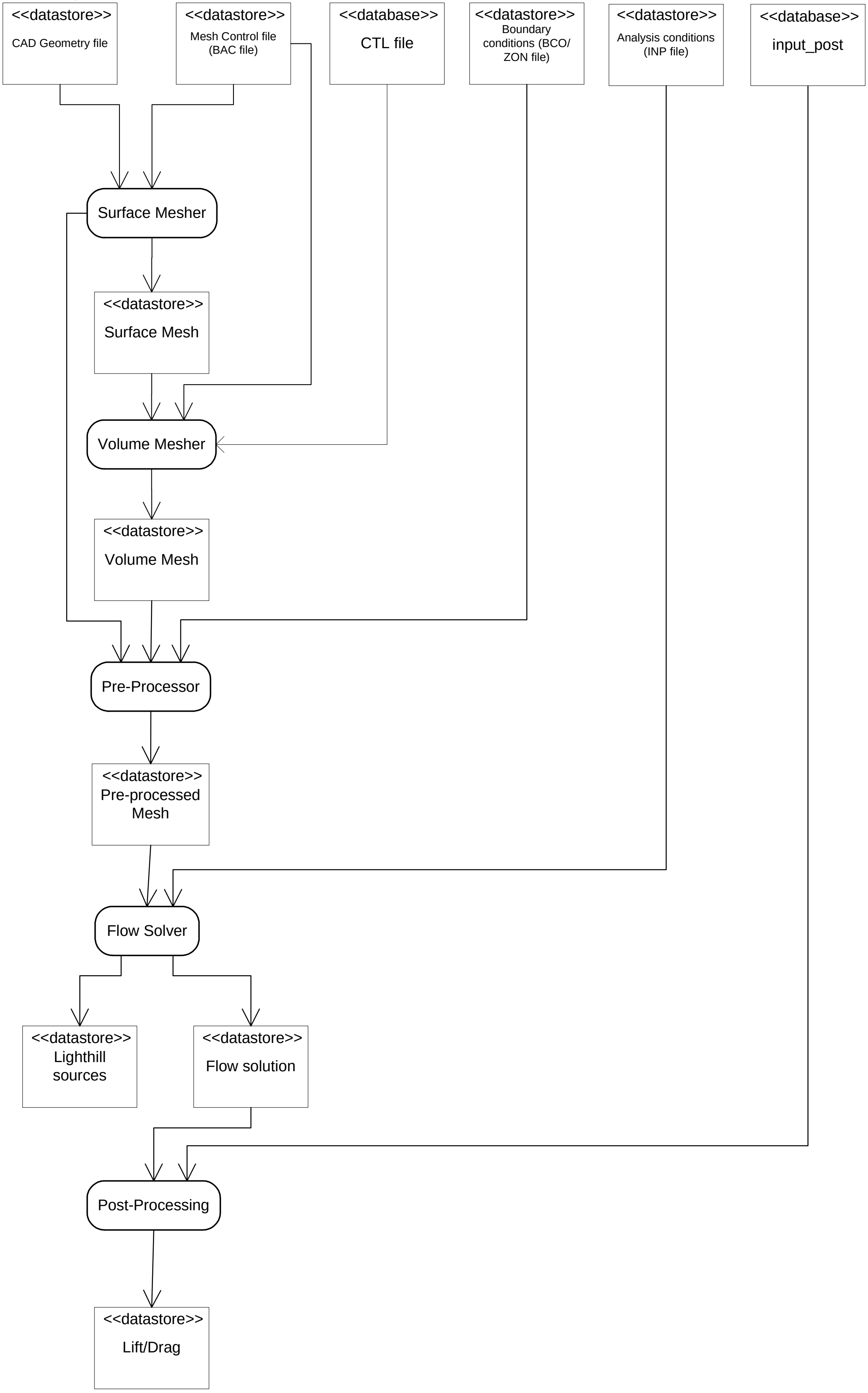


Figure 23: Aerodynamics infrastructure workflow

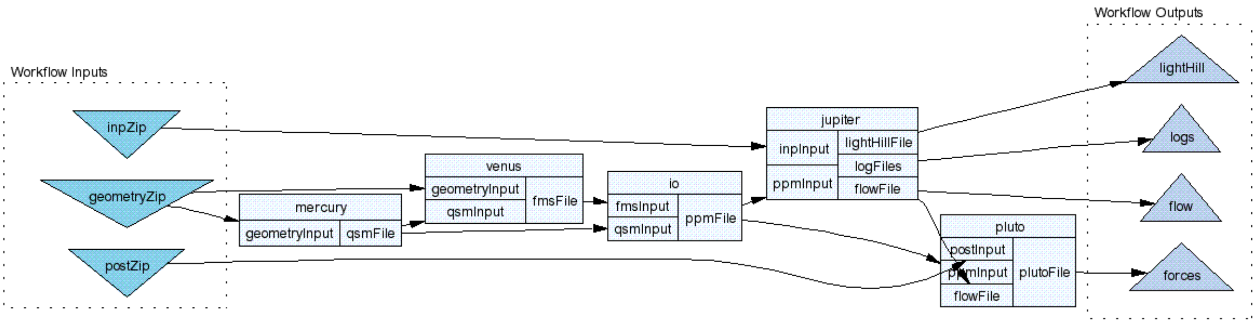


Figure 24: Aerodynamics workflow in Taverna

4.4.3.1 Issues

- We do not currently store all the data relating to an aerodynamics analysis due to its size. Only the final quantities of interest are stored. Three options are available to us:
  1. Carry on without storing all files relating to an analysis.
  2. Purchase sufficient memory so as to allow results to be stored.
  3. Seek an external data storage provider.

4.4.4 Aero-acoustics infrastructure workflow

The p01 and p17 files both contain file references as well as numeric control parameters such as frequency. p01 contains a reference to the d01 file and the Lighthill file. p17 has references to p01, d01, the acoustic solution (.res) file and the sources (Lighthill). The Lighthill sources, d01 and .res files will all be in data stagers and the p01 and p17 files must reference their data-stager filenames. The p01 and p17 files are generated from template files. These files are adapted to the kind of problem that needs to be solved: aero-acoustics. They are tuned with the correct parameters coming out either from the GUI (web) or from the CFD solver.

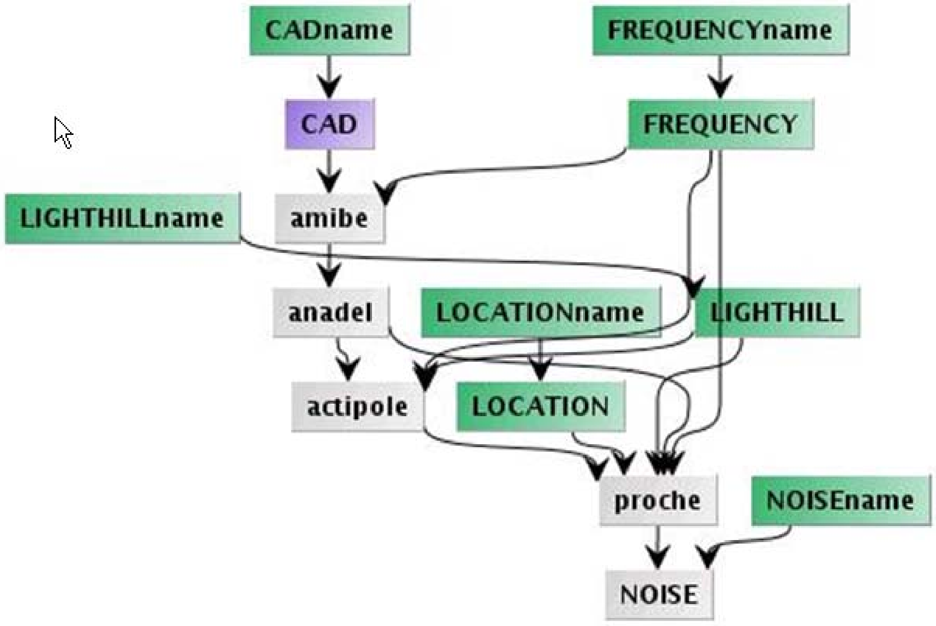


Figure 25: Acoustics workflow running through Taverna

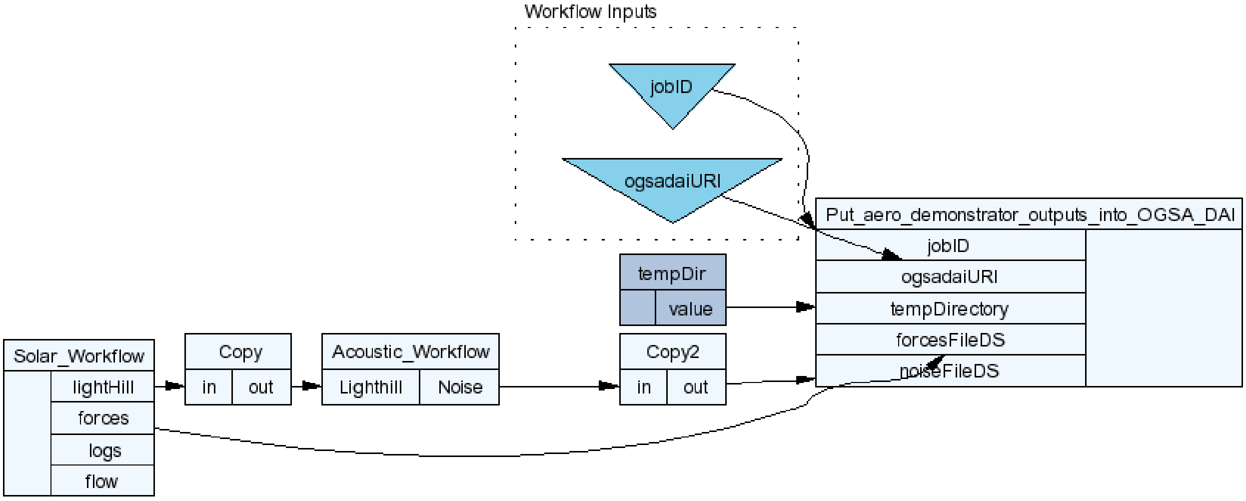


Figure 26: Linking the Aerodynamics and Aero-acoustics services and storing the results in the OGSA-DAI database

4.4.4.1 Issues

- The integration of multiple GRIA server should be addressed at some point since amibe can be run using the condor scheduler and the actipole solver using the Torque scheduler. This will raise the problem of distributed data access and some possible connection with jdistfs (Java distributed file system).
- Integration with other GridSphere portlets (such those developed at EADS) will be considered in the next phase of the project.

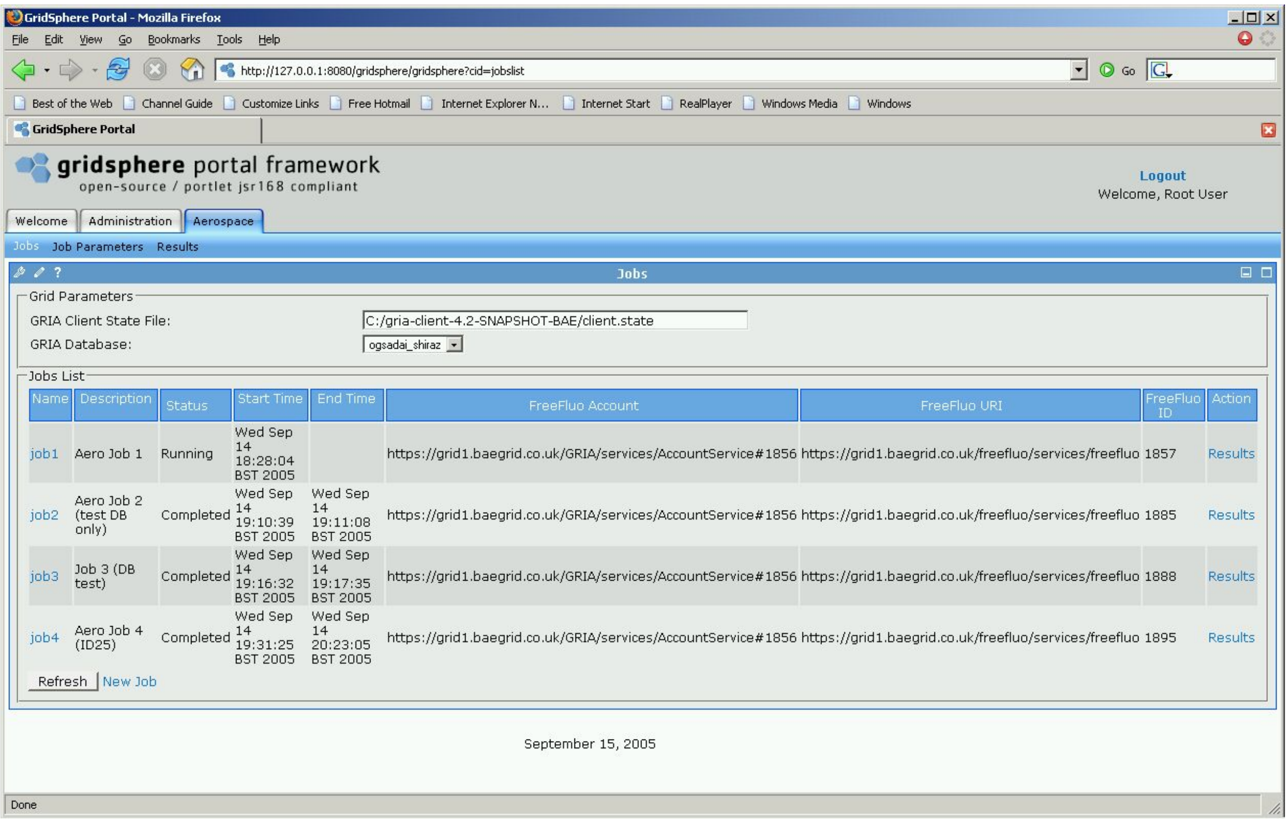
Up to PM12 the DoE phase of the workflow shown in Figure 20, the CAD service shown in Figure 22, the aerodynamics service workflow shown in Figure 23 and the aero-acoustics service workflow shown in Figure 25 have been published as GRIA services using Taverna. Further, workflows combining these services have been produced. Figure 26 shows an example of this whereby the aerodynamics and aero-acoustics workflows together with the OGSA-DAI database are all linked, again using Taverna.

4.5 Portal Description

The portal was designed specifically for the model problem defined by aerospace application scenario focusing on providing enough functionality to demonstrate the technical objectives of the prototype. The functionality provided to the user includes:

- Create, edit and delete jobs (See Figure 27)
- Parameterisation of input specification (See Figure 28)
- View job state during execution (See Figure 27)
- View the results (See Figure 29)

The aerospace application portal was developed using GridSphere. The GridSphere portal framework provides an open-source portlet based Web portal that is 100% JSR 168 compliant. The portal provides a data-driven design interface where the data is accessed using an OGSA-DAI service running at the design service provider. When a user creates a new job the job specification page is displayed. The user can enter design parameters and save these parameters to the OGSA-DAI data service. Once all parameters have been saved the user can start the job by specifying the Freefluo service where then application is deployed along with the design database service to be used. The application is then executed and tasks within the workflow stage the input data files by reading from the OGSA-DAI service and stage output metadata by post-processing output files and writing to the OGSA-DAI service. As the application executes and results are written to the OGSA-DAI service then cab can be retrieved by the portal and viewed by the user



Grid Parameters

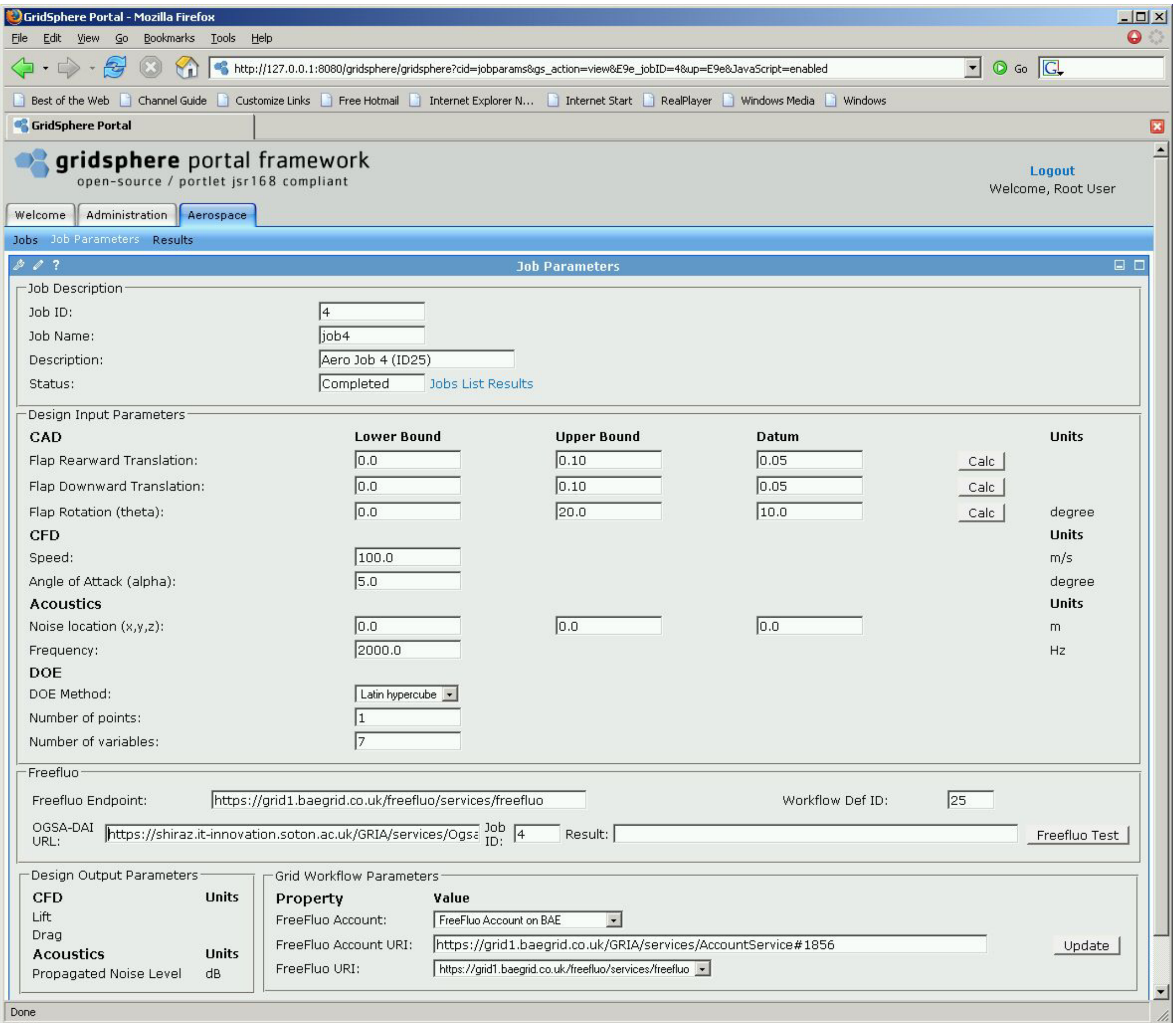
GRIA Client State File: C:/gria-client-4.2-SNAPSHOT-BAE/client.state

GRIA Database: ogsadai_shiraz

Jobs List

Name	Description	Status	Start Time	End Time	FreeFluo Account	FreeFluo URI	FreeFluo ID	Action
job1	Aero Job 1	Running	Wed Sep 14 18:28:04 BST 2005		https://grid1.baegrid.co.uk/GRIA/services/AccountService#1856	https://grid1.baegrid.co.uk/freefluo/services/freefluo	1857	Results
job2	Aero Job 2 (test DB only)	Completed	Wed Sep 14 19:10:39 BST 2005	Wed Sep 14 19:11:08 BST 2005	https://grid1.baegrid.co.uk/GRIA/services/AccountService#1856	https://grid1.baegrid.co.uk/freefluo/services/freefluo	1885	Results
job3	Job 3 (DB test)	Completed	Wed Sep 14 19:16:32 BST 2005	Wed Sep 14 19:17:35 BST 2005	https://grid1.baegrid.co.uk/GRIA/services/AccountService#1856	https://grid1.baegrid.co.uk/freefluo/services/freefluo	1888	Results
job4	Aero Job 4 (ID25)	Completed	Wed Sep 14 19:31:25 BST 2005	Wed Sep 14 20:23:05 BST 2005	https://grid1.baegrid.co.uk/GRIA/services/AccountService#1856	https://grid1.baegrid.co.uk/freefluo/services/freefluo	1895	Results

Figure 27: Jobs portlet



Job Description

Job ID: 4
Job Name: job4
Description: Aero Job 4 (ID25)
Status: Completed Jobs List Results

Design Input Parameters

CAD	**Lower Bound**	**Upper Bound**	**Datum**		**Units**
Flap Rearward Translation:	0.0	0.10	0.05	Calc	
Flap Downward Translation:	0.0	0.10	0.05	Calc	
Flap Rotation (theta):	0.0	20.0	10.0	Calc	degree
CFD					**Units**
Speed:	100.0				m/s
Angle of Attack (alpha):	5.0				degree
Acoustics					**Units**
Noise location (x,y,z):	0.0	0.0	0.0		m
Frequency:	2000.0				Hz
DOE					
DOE Method:	Latin hypercube				
Number of points:	1				
Number of variables:	7				

Freefluo

Freefluo Endpoint: https://grid1.baegrid.co.uk/freefluo/services/freefluo Workflow Def ID: 25

OGSA-DAI URL: https://shiraz.it-innovation.soton.ac.uk/GRIA/services/Ogsa Job ID: 4 Result: Freefluo Test

Design Output Parameters

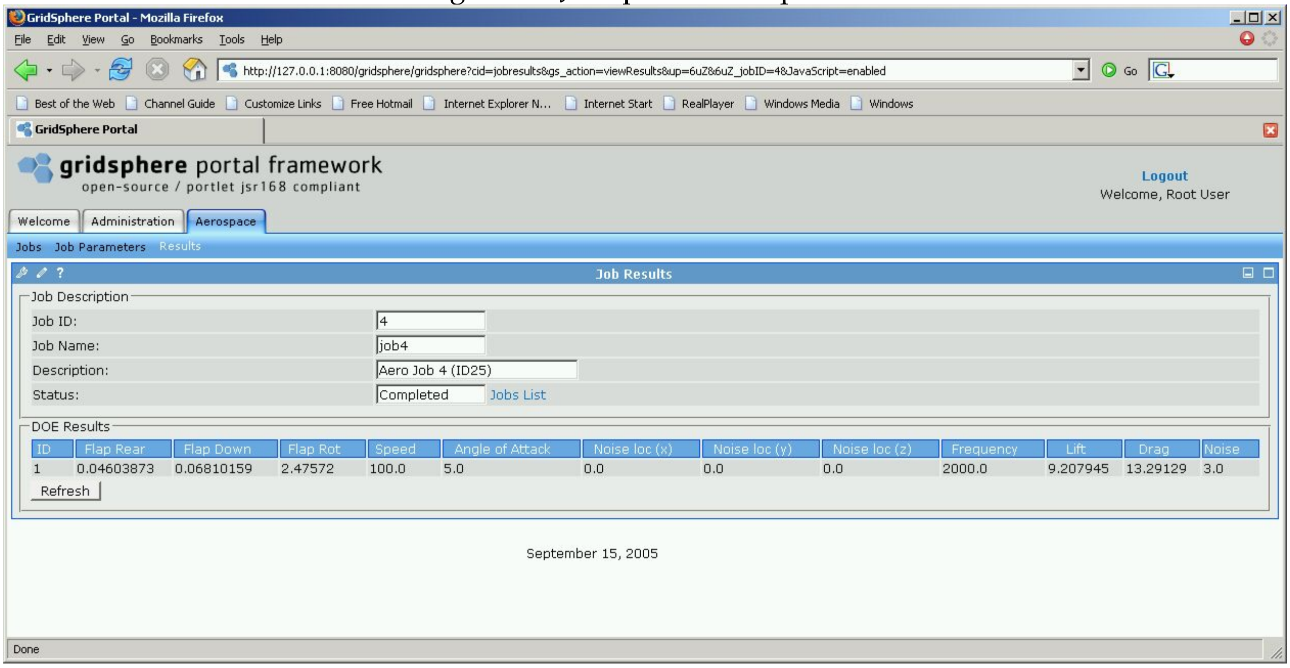
CFD	**Units**
Lift	
Drag	
Acoustics	**Units**
Propagated Noise Level	dB

Grid Workflow Parameters

Property	**Value**
FreeFluo Account:	FreeFluo Account on BAE
FreeFluo Account URI:	https://grid1.baegrid.co.uk/GRIA/services/AccountService#1856
FreeFluo URI:	https://grid1.baegrid.co.uk/freefluo/services/freefluo

Update

Figure 28: Job specification portlet



Job Description

Job ID: 4
Job Name: job4
Description: Aero Job 4 (ID25)
Status: Completed Jobs List

DOE Results

ID	Flap Rear	Flap Down	Flap Rot	Speed	Angle of Attack	Noise loc (x)	Noise loc (y)	Noise loc (z)	Frequency	Lift	Drag	Noise
1	0.04603873	0.06810159	2.47572	100.0	5.0	0.0	0.0	0.0	2000.0	9.207945	13.29129	3.0

Refresh

September 15, 2005

Figure 29: Results portlet

5 Conclusions

The objective of the work to date was to demonstrate connectivity of analysis codes across organisational boundaries. In order to demonstrate this, an application an example use case that required analyses from experts across disparate design teams and organisations was created. We chose to simulate the multi-disciplinary collaborative configuration design of a low-noise, high-lift landing system. The scenario is typical of sub-system design problems in the context of, say, future-concept, unmanned cargo vehicles that require an ability to use airfields in noise-sensitive locations. This required issues such as design, aerodynamic performance and acoustic performance to be considered. Experts from the University of Southampton, BAE SYSTEMS and EADS were called upon to address these respective issues. The enabler for linking systems developed by these separate groups of engineers was the GRID Resources for Industrial Applications software GRIA. This allowed individual groups working on their own services to link their software to other groups in different companies in order to form virtual organisations that were both secure and accountable. This work was further underpinned by Taverna which allowed workflows for the above to be created and published.

Many challenges were encountered throughout the course of this work. Technical terminology was one barrier. Another was data formats: different companies will invariably use different products with different data formats in the design process and when virtual organisations are set up data conversion becomes an issue. In the aerospace scenario both the aerodynamics and aero-acoustics services need to generate meshes on a variety of geometries produced from the CAD service. Automatic mesh generation is still fraught with difficulties and this is something we have had to contend with. There are several reasons for this, one is data formats as highlighted above and another is the fact that seemingly watertight geometries are not always exported correctly from CAD. Large data files, long run times and firewalls are other issues in the aerospace scenario.

Perhaps the biggest issue of all is that of software licensing. Much work in the aerospace community makes use of commercial software products. An example is CATIA V5 that was used in the CAD service. If virtual organisations similar to the above are to be set up in the future, it is a certainty that somewhere along the line a commercial product will be used. Current licensing agreements would be violated if part of such a product was included in a service and made available to those not registered to use the product.

6 References

1. SIMDAT Annex I - “Description of Work”
2. OPTIONS Design Exploration System, http://www.soton.ac.uk/~ajk/options/welcome.html
3. D.15.1.1 Detailed scenario description and architecture document (aero) and D.15.1.2 Consolidated aerospace requirements statement
4. T14 GRIA OGSA-DAI Integration Design, SIMDAT deliverable D.3.1.2.

End of Document