H2020-INFRAEDI-2018-2020

The European Centre of Excellence for Engineering Applications

Project Number: 823691

D4.2
Report on the Service Portfolio

The EXCELLERAT project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823691

Workpackage:	WP4	Enhanced services
Author(s):	Niclas Jansson	KTH
	Ralf Schneider	USTUTT
	Dennis Grieger	USTUTT
	Janik Schüssler	SSC
	Christian Gscheidle	FRAUNHOFER
	Adam Peplinski	KTH
	Gavin Pringle	UEDIN
Approved by	Executive Centre Management	
Reviewer	Claudio Arlandini	CINECA
Reviewer	Maike Gilliot	TERATEC
Dissemination Level	Public	

Date	Author	Comments	Version	Status
2019-10-14	R. Schneider	Initial draft	V0.1	Draft
2019-11-15	N. Jansson	Initial contributions	V0.2	Draft
2019-11-20	R. Schneider	Draft ready for QA – Round 1	V1.0	Draft
2019-11-29	R.Schneider	Second version for QA – Round 2	V2.0	Draft
2019-11-29	R.Schneider	Updates from QA – Round 2	V2.1	Draft
2019-11-29	R.Schneider	Document accepted following review	V3.0	Final

List of abbreviations

AHM	*All Hand Meeting*
AMR	*Adaptive Mesh Refinement*
BoF	*Birds of a Feather*
CFD	*Computational fluid dynamics*
D	*Deliverable*
DMD	*Dynamic Mode Decomposition*
DOA	*Description of Action*
EPI	*European Processor Initiative*
FPGA	*Field programmable gate array*
GA	*Grant Agreement*
GLL	*Number of collocation points*
gPCE	*Generalized polynomial chaos expansion*
GPU	*Graphic processing units*
HPC	*High performance computing*
KPIs	*Key Performance Indicators*
LES	*Large eddy simulation*
POD	*Proper Orthogonal Decomposition*
QoIs	*Quantities of interest*
UQ	*Uncertainty quantification*
WP	*Work Package*

Executive Summary

The deliverable (D) *4.2 Report on the Service Portfolio* describes EXCELLERAT's approach to develop and evolve the initial service portfolio, which was defined and described in D4.1, towards a portfolio of fully marketable services. Further on it presents the work performed in the tasks of workpackage (WP) 4 as they form an essential part of the evolved service portfolio.

With respect to the evolution of the service portfolio it is elaborated on, how the methodology applied to derive further service requirements from the work done within the use cases during the first 10 project months enabled the alignment of the service requests to the perspectives of the different actors involved in the development of the exascale engineering cycle. Further on, the deliverable presents the explicit topics of the service requests formulated by the EXCELLERAT use-cases along with their mapping to different actor perspectives and service categories which finally forms the initial marketable service layout which will be implemented via the EXCELLERAT portal.

Table of Contents

Table of Figures

Table of Tables

1 Introduction

This deliverable presents the evolution of the initial service catalogue, outlined in the description of action (DOA) and elaborated on in detail in D4.1, towards the currently envisaged marketable service layout. To this end, beside the detailed description of the envisaged marketable service layout i.e. the detailed description of the current service portfolio of EXCELLERAT, it will be first of all elaborated on the procedure how the list of service requests was gathered. It will further on be shown how the service portfolio, developed from the gathered service requests, can be categorised to target the different perspectives by which the currently identified user groups of EXCELLERAT potentially approach the centre.

These topics are covered in Section 2 and Section 3 of this document. In addition to that, the work conducted in the tasks of WP4 is described in Section 4. With respect to these on first view rather distinct two parts the following has to be noted. Even though WP 4 is named "Enhanced services" it became obvious during the first project year that the development and evolution of the service portfolio of EXCELLERAT is misplaced in WP4. First of all, there is no task dedicated to this effort in this WP4. Second of all, based on the WP communication structure, WP4 is not in the position to steer the development of what will basically be the foundation of the entity EXCELLERAT. Therefore, and in order to ensure that the development of the entity EXCELLERAT will include the complete consortium, this work will be incorporated into the coordination efforts located in WP1 as part of an amendment.

Note:

Since we will use throughout this document the abbreviations for the core-codes and their respective use-cases as defined in Table 3 "Applications and their domains" in the Grant Agreement (GA), the table is repeated here (Table 1) in parts to recall the abbreviations used for the core-codes and the topics of the use-cases to the reader.

Nek5000 (C1)	**C1U1**: Automotive - Automated design cycle and error control of air intake systems of engines
	C1U3: Aerospace - High fidelity simulation of rotating parts
Alya (C2)	**C2U1**: Automotive/ Aerospace - Emission prediction of internal combustion and gas turbine engines
	C2U2: Aerospace - Active flow control of aircraft aerodynamics including synthetic jet actuators
	C2U3: Transport systems - Coupled simulation of fluid and structure mechanics for fatigue and fracture
AVBP (C3)	**C3U1**: Aerospace and energy - Combustion instabilities and emission prediction
	C3U2: Safety applications - Explosion in confined spaces
FEniCS (C5)	**C5U1**: Aerospace and Automotive - Adjoint optimization in external aerodynamics shape optimization
FLUCS (C6)	**C6U1**: Aerospace - Design process and simulation of full equipped aero planes
	C6U2: Aerospace – computational fluid dynamics (CFD) coupling with computational structural mechanics including elastic effects

Table 1: Applications and their respective use-cases

2 Methodology to Evolve the Service Portfolio

To be able to evolve EXCELLERAT's service portfolio in a targeted way it was first of all necessary to develop a methodology by which additional services could be extracted from the use cases. Additionally, a way had to be found to identify the different perspectives from which EXCELLERAT will potentially be approached by the actors within the high performance computing (HPC) engineering community.

To this end, the first step was to translate the use-cases into so-called user-stories to reveal the relationship between the different actors potentially taking part in the realization of a use-case. How this was done is described below in Section 2.1 "Deduction of user stories".

In a second step the collected user-stories were analysed with respect to the different perspectives from which EXCELLERAT will potentially be approached by the actors within the HPC engineering community. This analysis resulted in the identification of four user-groups each with a different perspective towards the services provided by EXCELLERAT. The four user-groups, i.e. perspectives are described in Section 2.2 "Service perspectives".

The third step was to analyse the collected user-stories once again, this time with respect to the service requests posed in the user-stories. This second analysis led to a list of service request that could be organized into different service categories. These categories are described in Section 2.3 "

Categories of marketable services".

The fourth step in the evolution of the service portfolio was to relate the derived service perspectives of the different user-groups with the derived service categories. This exercise finally led to the initial layout of service perspectives and service categories, which can be implemented via EXCELLERAT's service portal and that is described in Section 3 "Initial Service Layout".

2.1 Deduction of user stories

As mentioned above, the first step in the evolution of the service portfolio was to translate the use-cases into so-called user-stories. While the use-cases focus on specific technical problems to be solved, the user-stories focused on the stakeholders engaged in the realization of a use-case and their relationships. Typical stakeholders are e.g. the (end-) user, and the technology (or know how) provider. This approach allowed the members of the consortium, some of whom contributed both, the use cases and the core codes used in them, to put themselves in the shoes of EXCELLERAT's potential external customers.

As an example, for this approach we will consider C1U3 - High fidelity simulation of rotating parts with NEK5000. The constellation of actors participating in the use-case as it was originally set up is depicted in Figure 1. Shortly the interaction of the three parties can be described as follows: A helicopter manufacturer is looking for a solution to execute high fidelity simulations of helicopter rotors. CINECA who has experience with NEK5000 would be able to come up with such a solution even though there is a lack of experience with respect to some needed features like e.g. adaptive mesh refinement and high order meshing for complex geometries which KTH, as one of the code owners and main developers of NEK5000, can provide.

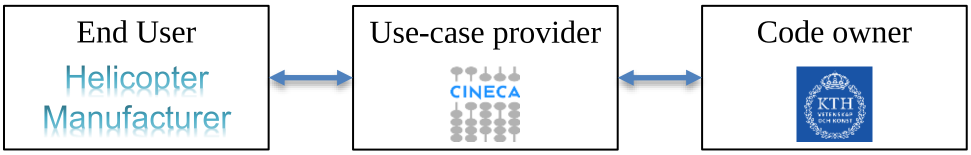


Figure 1: Initial actor constellation and information flow in C1U3.

After starting to work on the use-case and discussing with the EXCELLERAT consortium the different perspectives from which one is able to shed light onto this use-case from an outside point of view, CINECA was able to develop seven user stories with whose problems and resulting requirements EXCELLERAT could potentially be confronted once it is established to be the European knowledge hub for exascale engineering applications. To illustrate the user-stories three examples will be given:

- First of all, it would be possible for EXCELLERAT to be approached directly by the industrial end-user who would like to execute high fidelity simulations of rotating parts with an open-source simulation code and is looking to find a suitable workflow to execute this exascale engineering problem efficiently.
- Secondly, it would be possible for EXCELLERAT to be approached by the use-case provider who is already collaborating with the industrial end-user outside the framework of EXCELLERAT to provide the requested workflow and is now searching for expertise in robust strategies for rotating parts modelling at high Mach numbers in spectral element codes since NEK5000 is proven only at low Mach numbers.
- The third example shows how the change of perspective revealed further collaboration opportunities within the EXCELLERAT consortium and enabled the generalized extension of the service portfolio beyond the original goals of the use-cases. Thus, it is possible that an end-user, building on the already successfully implemented original use-case, also performs the simulation of rotating parts and finds in this case that it would significantly increase the efficiency of the method if a special evaluation program could be coupled in-situ to NEK5000. The search of the end-user for required expertise would expand the circle of actors to include the code developers of a corresponding visualization program as well as the developers of a data analytics library to be used, who would be able to provide the corresponding know-how.

By these three examples it can already be recognised that on the one hand, the problems and requirements derived from the user-stories are not necessarily focused to the narrowed scope of the use-cases but can be formulated to fit in a more generalized scope in which a service can be provided that targets the engineering community. On the other hand, the requirements can be categorized and associated with different user-groups like application end-users and code developers.

2.2 Service perspectives

From the analysis of the user-stories, as the second step in the evolution of EXCELLERAT's service portfolio, it was possible to identify four user-groups. Each of this user-groups represents a perspective by which EXCELLERAT can be potentially approached.

To illustrate this, the example of service requests for consulting on in-situ visualization is taken. The analysis of the user-stories showed, that consulting on in-situ visualization was requested several times within different user-stories. On the one hand, there was a detailed service request regarding implementation methods and efficient procedures for the coupling of in-situ libraries to a code with elements of high order. On the other hand, the request was formulated with respect to the handling and concrete implementation of a special application case with an already in-situ capable code. This means that from this example one can already extract two service perspectives. The first request is obviously posed from the perspective of the code-developer while the second one is obviously posed from the perspective of the end-user.

By analysing all user-stories in way demonstrated by the example above, the following four perspectives from which services can be requested from EXCELLERAT could be extracted:

1. **The perspective of the application end-user.**

   If EXCELLERAT is approached from this perspective, it should provide services that deliver solutions to simulate a given engineering problem by executing an exascale engineering cycle. This perspective will further on be named "**Solution evolution**". It should be explicitly noted that at this point solution means the processing of an engineering problem along the entire exascale engineering cycle and not just the solution of a discretized problem in the sense of executing a simulation code or a numerical solver. Under this perspective, all services are united that will evolve the solution of engineering problems towards exascale.

2. **The perspective of the code-developer.**

   If EXCELLERAT is approached from this perspective, it should provide services that deliver expertise in areas that enable the code-developer to evolve engineering software packages towards extreme-scale applicability. This perspective will further on be named **"Code / Application evolution"** with application meaning all software components being used in implementing the engineering cycle. Under this perspective, all services are united that will evolve engineering codes / applications towards exascale by means of massive parallelism and extreme scalability while maintaining the applicability to real-world problems and improving the ease of use for the end-user.

3. **The perspective of the vendor.**

   In this context vendor refers to the group of system integrator, hardware developer, system-software developer or hardware vendor. If EXCELLERAT is approached from this perspective, it should provide services that deliver input and discussion partners to evolve the future HPC-systems to be ready to efficiently execute the exascale engineering cycle.

4. **The perspective of the HPC engineering community.**

   In this perspective any other actor who is already or wants to become part of the HPC engineering community is included.

The development of the services perspectives described above will help to develop a strategy for the targeted user guidance within the EXCELLERAT portal i.e. help to develop a strategy for the targeted presentation of the services to EXCELLERAT's user community in general. This must be seen as particularly important as it will help to conduct a more fine-grained and targeted market analysis that will significantly contribute to the development of a sustainable business model.

2.3 Categories of marketable services

As already mentioned in the introduction of Section 2, the third step towards the evolution of EXCELLERAT's service portfolio was to analyse the user stories with respect to the service requests posed in the user-stories. This second analysis led to a list of service requests that could be organized into six different service categories. Four of these categories concern different types of consulting, one targets the provisioning of tools for the exascale engineering cycle and another one the provision and hosting of datasets. The detailed description of the service categories is given in Sections 2.3.1 - 2.3.3 below. The service requests posed from each of the four service perspectives in the respective service category are afterwards listed in Section 3.

2.3.1 Consulting service categories

The most prominent service category for which all user-stories contained requests was consulting i.e. know-how transfer in its different appearances. The currently identified different appearances of consulting were further on separated into four different types of consulting:

1. **Active one-on-one consulting**

   In this type of consulting an expert or a group of experts helps the service inquirer to develop or execute / implement a feature, problem, method etc.

2. **Passive one-on-one consulting**

   With this type of consulting it is referred to best-practise guidelines which deal in a technically and detailed way with the execution / implementation of a feature, problem, method etc.

3. **One-on-n consulting**

   This type of consulting refers to training courses.

4. **n-on-n consulting**

   In the form of access to community events like expert networks, specific workshops or targeted symposia.

In turn, these four types of consulting could in part be related to all of the perspectives described in Section 2.2 even though additionally a relation between the requested type of consulting and the inquirer's background i.e. affiliation became visible. E.g. requests originating from actors with academic affiliation and the code-developer perspective mostly targeted consulting requests in form of best practise guidelines, training courses and dedicated workshops whereas requests from actors with an industrial affiliation and the end-user perspective mostly targeted one-on-one consulting and specific training courses.

2.3.2 Provisioning of tools

Besides consulting services, the analysis of the user stories with respect to the service requests revealed that a demand for provisioning of dedicated tools for the exascale engineering cycle exists. As already defined in the grant agreement [1] these tools complete the exascale engineering cycle with respect to pre- and post-processing, data management and usability. The persisting request for these tools confirmed the consortium in its decision to grant the enhanced services, developed in Task 4.1 – Task 4.5 the envisaged prominent position within the evolution of the service portfolio. The work conducted in the tasks of WP 4 along with their envisaged relation to the service category "Provisioning of tools" is explicitly described in Sections 4.1 – 4.5.

2.3.3 Dataset hosting

From the analysis of the user-stories with respect to the service requests as well as from ongoing discussions within the consortium the service request to host and provide large data sets was extracted. Even though the specific purpose of the datasets to be hosted was not mentioned in the user-stories from the perspective of the end-users, the requests posed from the code-developers' perspective in that respect were more clear. Since in almost every use-case a dataset for validation or verification is used this led to the conclusion that in the first place such datasets should be targeted. These datasets, either from reference simulations or measurements, used to proof the correctness of the implemented methods, models, algorithms and approaches will be beneficial for code-developers as well as for end-users, trying to verify and validate a model setup.

The overview about the data-sets currently used by the use-cases as input and validation data is given in the list below.

- NASA CRM Modell at stall. [2]
- JAXA high lift configuration standard model. [3]
- Injection problem - Menard et.al 2007 [4]
- Configuration of gas turbine combuster [5]
- NACA0012 [6]

The current idea is to set up a repository for datasets and data sources which can be directly hosted by EXCELLERAT or linked into the repository from external sources. In addition to the datasets directly used by the use-cases, standard cases like e.g. the Taylor Green Vortex [7] can be added to the repository.

Further on the idea exists to provide access via the repository to reference result datasets produced in the frame of EXCELLERAT even though the discussion whether it is reasonable to host complete datasets is still ongoing since this approach would need a significant amount of resources.

3 Initial Service Layout

After having elaborated on the perspectives under which services are requested in Section 2.2 and on the service categories that resulted from the complete list of service requests in Section 2.3, this Section introduces the currently selected layout for the service implementation which is shown in Figure 2.

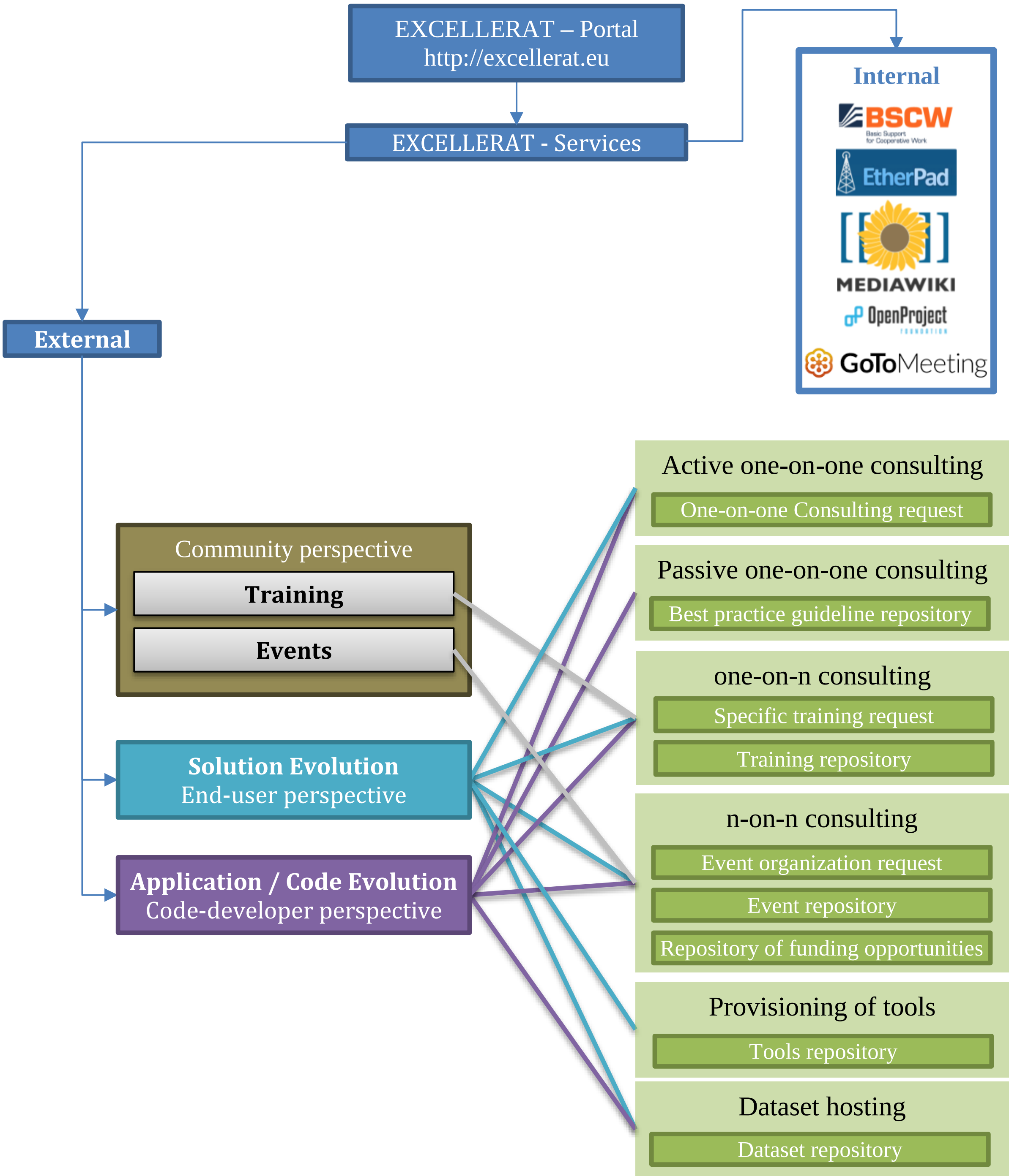


Figure 2: Initial service layout

As can be seen, the EXCELLERAT service portal will be accessible form the EXCELLERAT homepage. From the service portal the external and the internal services can be accessed. Currently the internal services are only accessible for members of the EXCELLERAT consortium. Since the internal services are currently not considered as marketable services they are listed in this deliverable only for completeness. Currently the following list of tools for collaborative work and project management is provided to the members of the consortium:

1. BSCW: File sharing.
2. EtherPad: Collaborative text editing.
3. MEDIAWIKI: Internal documentation.
4. OpenProject: Project Management.
5. GoToMeeting: Conference Calls and Webinars.

If a non-consortium user enters the service portal she or he will be guided to the external services where the user can decide from which perspective the services of EXCELLERAT should be approached.

As shown in Figure 2, if a user wants to approach EXCELLERAT from general perspective of the HPC engineering community she or he can further choose to approach the training, i.e. the one-on-n consulting service category or the events, i.e. n-on-n consulting service category. This further differentiation was done since training is the most advanced and elaborated form of know-how transfer. This was taken into account already during the proposal writing phase by creating the dedicated task 5.4 and now by implementing a dedicated approach perspective.

Further on the differentiation within the community's approach perspective was done since from the perspectives of the end-user, the code-developer and the vendor know-how transfer in form of n-on-n consulting was requested. This means all three groups that approach EXCELLERAT with focused service requests would like to interact with the community. Conversely, the conclusion was drawn that the community should be given the opportunity to interact with these groups via direct this access to the n-on-n consulting service category.

In Figure 2, it can further on be seen that not all service perspectives are connected to all service categories, e.g. the solution evolution perspective is not connected to the passive one-on-one consulting service category. This resulted from the fact that not from all service perspectives service requests were posed in all service categories. In fact, since all use cases are driven by the requirements of the user perspective, which in turn induce the requirements from the perspective of the code developers, the requests arising from the perspective of system evolution i.e. the vendor are at the moment relatively underrepresented. Due to that the implementation of this service perspective will be postponed and it is not shown at all in the initial service layout.

In addition to the connection of the service perspectives to the service categories in Figure 2 the services which are currently planned to be implemented under each service category are shown. It can be seen, that initially nine different services will be set up of which six are planned to be implemented as tagged repositories and three as service request forms.

After the presentation of the first service layout, in the following Sections we will elaborate on in more detail from which service perspective in which of the various service categories service requests were posed.

3.1 Solution evolution

As stated above the requests targeting the perspective of the end-user mainly concerned the processing of an engineering problem along the entire exascale engineering cycle and not just the solution in the sense of executing a simulation code or a numerical solver. This means that the requests assume that there is already a method or toolchain for the given problem to be solved and that consulting is "only" needed in relation to the application of the method, the process chain or its components i.e. tools. If, on the other hand, the further development of individual components of the method becomes necessary i.e. implementation and code-development has to be conducted, the requests target one-on-one consulting by the code developer in which it is assumed that the developer does the actual development work until a ready to use solution exists. As examples for these statements, C1U3 and C5U1 can be taken into account. In these use-cases, CINECA as the use-case owner is provided consulting services by KTH – the code-owner – in terms of code usage and case-setup. Additionally, implementation work has to be conducted. In the case of NEK5000 adaptive mesh refinement methodologies have to be further developed and implemented and in the case of FEniCS matrix assembly routines have to be modified to make the code usable in the targeted exascale uncertainty quantification scenario. Both implementations are not carried out by CINECA but as said by the code-owner KTH.

In addition to the insights discussed in the previous Sections, the analysis of the user stories and subsequent discussions within the consortium indicated that the requests made from the end user's perspective are much more specific than those made from the other perspectives. Meaning that the requests are much more targeted towards the solution of given problems by specific codes and approaches rather than towards more generic methodologies. These considerations, as well as the assumption that later most of these request will originate from industrial end-users, lead to the conclusion that for this perspective best practise guidelines even though requested by the user-stories would have two major disadvantages:

1.) The production of best practise guidelines in advance would require a significant amount of person month to be invested now. But from these efforts guidelines with a relatively narrower scope would result in a limited impact to the community. Even though, ideas exist how the production of best practises guidelines targeting the end-user perspective can be part of the business model to be developed.

2.) If the production of guidelines would be postponed until a specific request is posed, i.e. The production of the guidelines would be done on-demand, it would require too much time and due to that would most likely be superseded by one-on-one consulting activities.

In view of these two disadvantages, the implementation of the consulting category 1 from the end-user’s perspective will be postponed. Additionally, the analysis of the service requests posed from the end-user perspective gave on little differentiation between the service categories 2 and 3. Due to that, for the remainder of this Section, we will disregard the difference between the first three consulting categories and only list requests targeting the fourth service category separately.

3.1.1 Service requests targeting one-on-one and one-on-n consulting

- Visualization methods for specific engineering work flows (C1U1, C1U3, C5U1).
- Data analytics as pure post processing and in-situ (C1U1, C1U3, C5U1).
- Visual analytics (C1U1, C1U3, C5U1).
- Data management for large data sets (C1U1, C1U3, C5U1).

- Meshing techniques/software (C1U1, C1U3, C5U1).
- Guidelines for scalable simulation workflows (C1U1).
- Optimal use of resources, when running uncertainty quantification (UQ) enhanced simulation (C1U1, C1U3, C5U1).
- Service to advise on runtime approach i.e. how to obtain best efficiency vs. elapsed time trade-off (C1U3, C5U1).
- Application of mesh adaption methods and for front tracking (C3U2).
- Application of multi-code coupling technologies (C2U2).

3.1.2 Service requests targeting n-on-n consulting

Information about and access to n-on-n consulting activities such as conferences, workshops and symposia were requested not that frequently from the end-user perspective compared to code developers and hardware manufacturers perspectives.

From the end-users' perspective especially the topic of a cross-competency experts panel with respect to multi-physics model configurations was requested.

3.1.3 Provisioning of tools

As already mentioned in Section 2.3.2, besides consulting services that a demand for provisioning of dedicated tools for the exascale engineering cycle exists and that this confirmed the consortium in its decision to grant the enhanced services, developed in task 4.1 – task 4.5 their prominent position further on. From the perspective of the end-user especially requests targeting the work done in task 4.2 – task 4.5 were posed.

3.1.4 Dataset hosting

As already mentioned in Section 2.3.3, from the end-users' perspective service requests for dataset hosting were posed even though the specific purpose of the datasets to be hosted was not mentioned.

3.1.5 Services to be provided from the end-users' perspective

The summary of services that are envisaged to be provided from end-user-perspective are shown in Figure 3. Namely, these are:

- Management of requests targeting one-on-one consulting activities.
- Management of requests for special training courses or events.
- A training courses repository listing trainings.
- Management of requests about organization of special events like cross competency expert panels.
- An event repository listing community events like workshops and symposia targeting specific topics.
- A tools repository providing access to exascale ready tools for the engineering cycle. If tools are developed within the framework of EXCELLERAT or can be made available directly by EXCELLERAT due to their licensing, their provision will be included in this service.

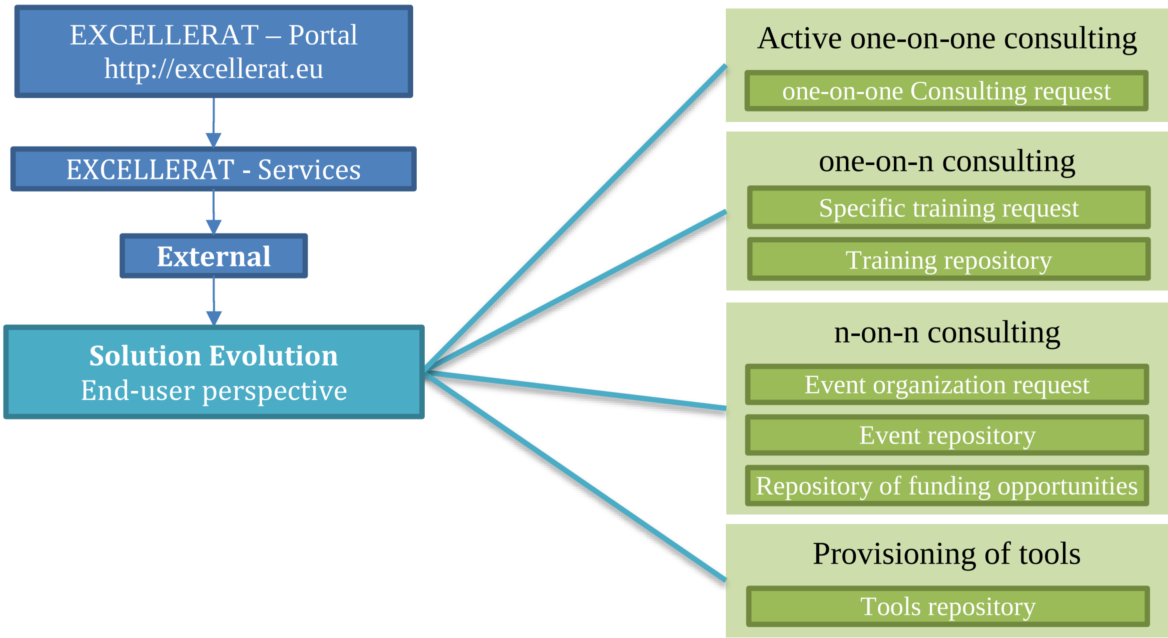


Figure 3: Currently envisaged list of services from the end-user perspective

3.2 *Application / Code evolution*

As stated above, from the code developers' perspective in the first place best practise guidelines with rather generic topics targeting methodologies were requested. Discussions within the consortium on this topic showed that the focus on basic methods is due to the fact that highly scalable applications have very specific implementation strategies. For instance, a mesh partitioning scheme can be transferred from one code to another if both codes feature the same or a similar mesh topology description even though this does not apply to its explicit implementation. In most cases, the direct transfer of an implementation will fail due to the programming language and code-specific data structures used. For this reason, code developers are apparently not interested in explicit guidelines tailored to a code, but in descriptions of methodological procedures. However, it should be noted that a clear difference between scientific publications and the requested best practice guidelines could be recognized during the discussions. In this case, methodological description does not mean the mathematically correct formulation of a numerical method, but rather the technical methodological description of a procedure of the type: “The network partitioning library was used in the following way in connection with a CFD finite volume code and the following results were achieved with regard to efficiency, etc.”.

3.2.1 Service requests targeting passive one-on-one consulting

Explicitly the following list of topics for best practice guidelines was extracted from the requests formulated in the user-stories. For reference the core-code along with the use-case to which the request applies is given in brackets behind the respective topic:

- Porting of legacy applications to modern hardware (C5U1).
- Know-how about efficient data transfer/management (C1U1, C1U3).
- In-situ analysis workflows (C5U1).
- Development of automated process chains (C6U1).

- Implementation for In-situ methods (C1U1, C1U3, C5U1).
- Error indicators and estimators (C1U1, C1U3, C5U1).
- Using accelerators/heterogeneous systems (C1U1, C5U1).
- Efficient data redistribution (C1U1, C1U3, C5U1).
- Testing and validations procedures - a debug procedure (C3U2).
- Handling of adaptive mesh refinement with physical constraints (C2U1).
- Multi-code coupling technologies and implementation strategies (C2U2).
- Two-layer wall models (C2U2).
- Optimal use of resources, when running UQ enhanced simulation (C1U1, C1U3, C5U1).

3.2.2 Service requests targeting active one-on-one and one-on-n consulting

In this Section, the overview of the requested consulting topics which were categorised in the active one-on-one and also the one-on-n consulting categories from the code-developer perspective is given.

First of all, the analysis of the requested topics showed the expected demand for expertise in the direction of performance engineering, code efficiency improvements as well as topics connected to those like programming models and knowledge about hardware. Explicitly the following topics were named:

- Detailed performance analysis for identification of code sections with high potential for runtime improvements (C6U1).
- A benchmark ready to test the load balancing performances (C3U1).
- Know-how of using accelerators/heterogeneous systems (C5U1).
- Expertise to select and implement suitable programming models for enhanced code efficiency (C6U1).
- Service to advise on runtime approach (how to obtain the best efficiency vs elapsed time trade-off) **(**C1U3, C5U1).

Beside the requests for performance engineering, which targeted directly the codes, there were some requests identified, which were collected under the topic of "system-application-interaction". This means, these requests cannot be handled by one performance engineering of a single application but could be tackled with an integrated approach based on the integration of the system's and the application's performance monitoring and error handling. Explicitly, the following topics were named:

- A detailed performance assessment that gives pertinent figures at the end of a year of production (C3U1).
- Error handling strategy that goes beyond codes, to relay precise error messages (C2U1, C3U2).

The third area in which topics for one-on-one consulting and training were requested from the code developer perspective was concerned with numerical and algorithmic methods.

- Smart load balancing (C3U1, C1U3, C1U1).

- Multi-code coupling technologies (C2U2).
- Mesh adaption techniques for front tracking (C2U1, C3U2).

Furthermore, an interesting request was made for temporary technical project management, which takes over the coordination and organisation of complex code development tasks e.g. like the ones originating from the implementation of end-user solutions. Even though currently no final strategy is developed to implement this as a dedicated service it will for now be continued under consulting and also being kept on the agenda of the business development task.

3.2.3 Service requests targeting n-on-n consulting

From the code-developers' perspective several requests directly targeted access to n-on-n consulting events such as conferences, workshops and symposia even though most of the topics additionally appeared also in requests towards the other consulting categories. This was not that much of a surprise since many of the code-developers of highly scalable engineering applications are affiliated with academic research institutes in which the know-how exchange via such events is more present than within industry. Explicitly the requests targeted the following topics

- New architectures (C1U3, C5U1).
- Optimization method (C3U2).
- HPC resources (C3U2).
- Numerical methods (C3U2).
- Programming models for enhanced code efficiency (C6U1).
- Co-Design.

Even though not directly named within the user stories, the topic of Co-Design was also grouped under the code-developers' perspective since based on ongoing discussions within the consortium about the topic in addition to the dedicated task, a transversal working group was established. This group is comprised of members from different partners, tasks and WPs to discuss their approach and findings with hardware vendors, system integrators etc. This means for this group a service which provides dedicated workshops and connection to the system and hardware developers would be of great help.

3.2.4 Dataset hosting

As stated in Sections 2.3.3 and 3.1.4 the service to host and provide datasets for validation and verification was extracted from the requests posed in the user-stories from the end-users' perspective but was also derived from the ongoing work carried out within the use-cases.

3.2.5 Services to be provided from the code-developers' perspective

The summary of services that are envisaged to be implemented from the code-developers' perspective are shown in Figure 4. Namely, these are:

- Management of requests targeting one-on-one consulting activities.
- A repository of best practise guidelines.
- Management of requests for special training courses or events.
- A training courses repository listing trainings.
- Management of requests about organization of special events like cross competency expert panels.

- An event repository listing community events like workshops and symposia targeting specific topics.

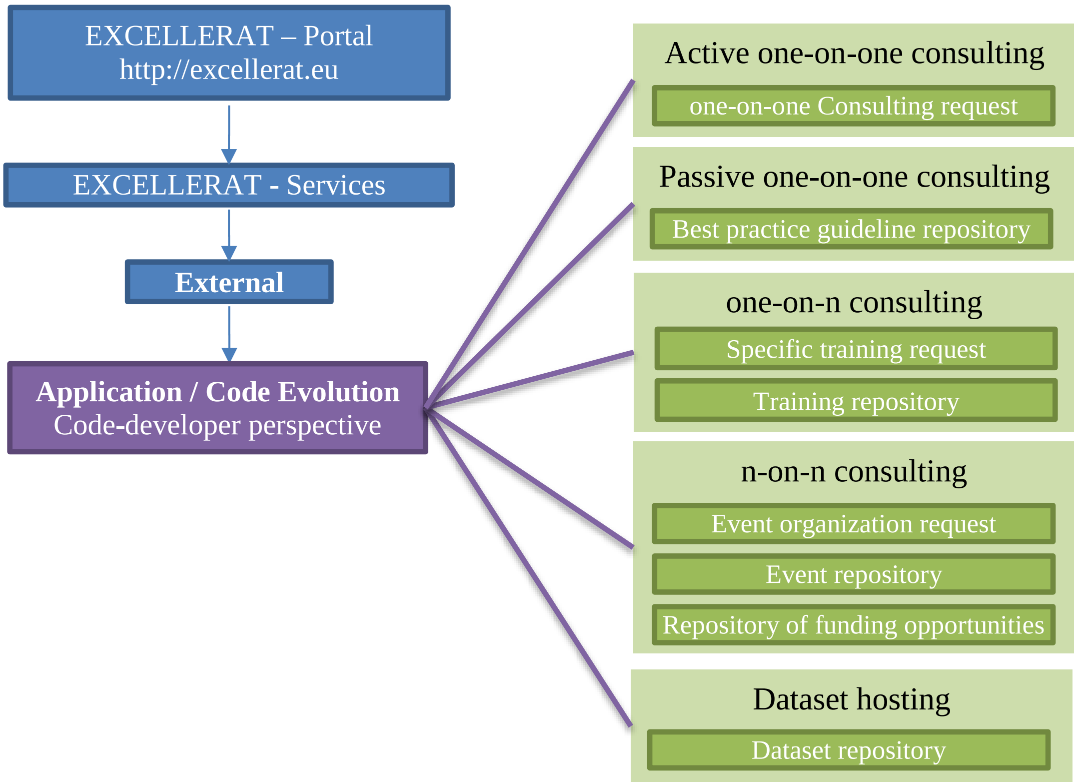


Figure 4: Currently envisaged list of services from the code-developers perspective

3.3 System evolution

Since all use cases are driven by the requirements of the user perspective, which in turn induce the requirements from the point of view of the code developers, the requests arising from the perspective of the system integrator, hardware developer, system-software developer or hardware vendor are at the moment relatively underrepresented. Currently the three topics directly related to system and hardware engineering are:

- Co-design.
- Detailed performance assessment that gives pertinent figures at the end of a year of production.
- Error handling strategy that goes beyond codes, to relay precise error messages.

Since these topics are already dealt with from the user and code developer perspective, the implementation of the vendors' service perspective is postponed for the time being. Once task 4.1 and the Co-Design working group are producing meaningful output, the implementation of the respective consulting and community services will be done.

3.4 Training

Since training is the most advanced and elaborated form of know-how transfer in all of the areas mentioned so far, this was taken into account during the proposal writing phase by creating a special task for it and now by implementing a dedicated option to approach the training services from the community perspective. Special topics for training courses requested within the user-stories covered so far:

- Using accelerators/heterogeneous systems (C5U1).
- Porting legacy F77 codes to modern hardware (C5U1).
- External aerodynamics and adjoint-based optimization (C5U1).
- Implementation of scalable simulation workflows (C1U1).
- Multicode coupling technologies (C2U2).
- Load balancing libraries (C3U1).
- Mesh adaption libraries (C2U1, C3U2).
- Adaptive mesh refinement techniques (C2U1).
- Two layer wall models (C2U2).
- Simulation of rotating parts and meshing (C1U3).

3.4.1 Services provided within the training option of the community perspective

The summary of services that are envisaged to be provided under the training category are shown in Figure 5. The services to be implemented are:

- Management of requests for special training courses or events.
- A training courses repository listing trainings.

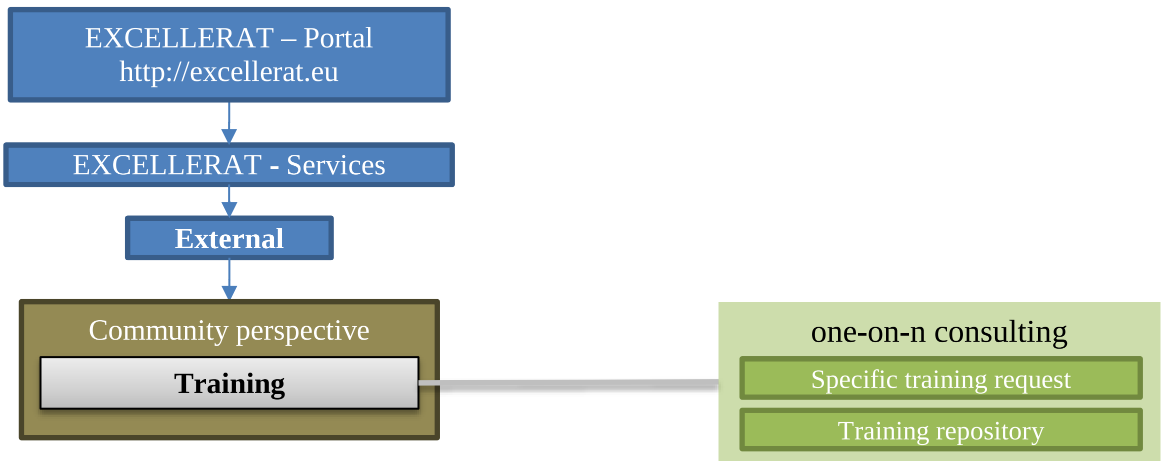


Figure 5: Currently envisaged list of services to be provided within the training category

3.5 Events

Since from all three perspectives end-user, code-developer and hardware / system provider know-how transfer in form of n-on-n consulting was requested, this category of services will, like training, also be covered separately. Specific requests for networking events and access to expert networks were posed with respect to the following topics:

- Accelerators/heterogeneous systems.
- Modern and future hardware.
- Optimization methods.
- HPC resources.

- Numerical methods and algorithms.
- New programming models.
- Physical modelling.

Additionally, besides the requests extracted from the user-stories the two service topics envisaged in the original proposal:

- Support access to industry funded research opportunities;
- Promoting outstanding applications i.e. exascale demonstration runs as well as high capacity runs

will be implemented as soon as respective opportunities can be linked and results of outstanding applications within the framework of EXCELLERAT are produced.

3.5.1 Services provided within the events option of the community category

The summary of services that are envisaged to be provided under the community category are shown in Figure 6. Namely, these are:

- Management of requests about organization of special events like cross competency expert panels.
- An event repository listing community events like workshops and symposia targeting specific topics.
- A repository with funding opportunities targeting HPC engineering.

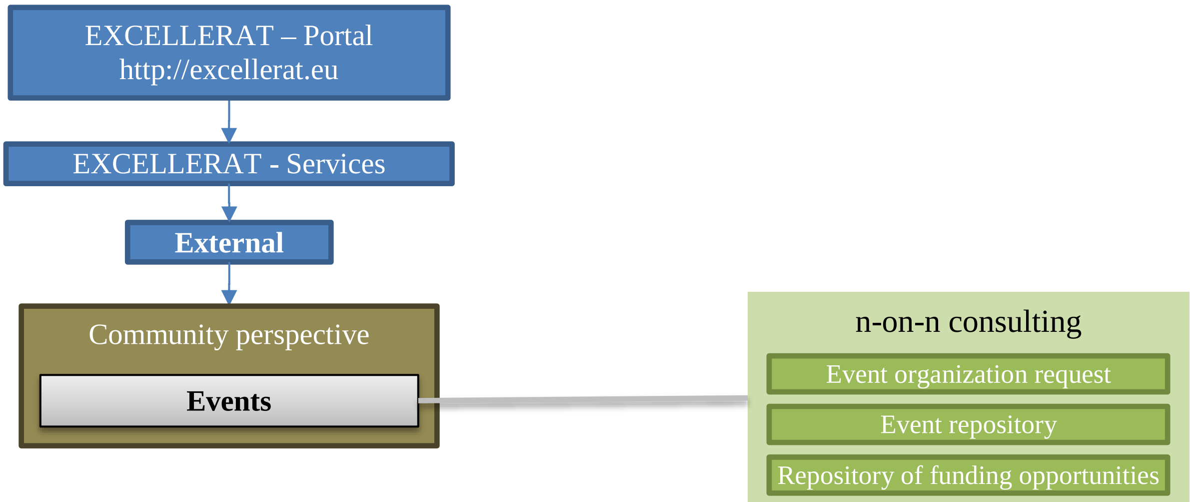


Figure 6: Currently envisaged list of services within the community category

4 Development of the Enhanced Services

In this Section, the work done in the tasks of WP 4 tasks during the first project year is presented. As already elaborated on in Section 1 and Section 2.3.2 this work is described in this deliverable since it is still considered as an essential part for the development of EXCELLERAT's service portfolio. For the remaining duration of the project an amendment is planned that will separate the reports on the service portfolio away from the technical efforts conducted in the tasks of WP4. To ensure that this work still will be monitored and the WP communication structure is represented by official deliverables, it is further on planned to introduce intermediate reports on "Best Practices and Tools for Visualization, Data Management and Analytics of Engineering Simulations at Exascale".

4.1 Co-design

This Task started M6 due to staffing issues at the Lead Partner, namely UEDIN.

Co-design is generally regarded to be application experts working closely with hardware developers, e.g., Intel, ARM, NVidia, AMD, Seagate, the European Processor Initiative (EPI), etc. However, throughout the history of HPC, the development of new hardware has never been influenced by a single application or a small collection of applications from a single scientific domain. Here, we define this as direct co-design with vendors.

Indirect co-design with vendors, on the other hand, is possible. For instance, emulators of future hardware have been shared with application developers, typically at US Department of Energy sites, which permits the application owners to tweak their code to perform well on the emulator and, therefore, exploit the hardware once it is realised. This does mean hardware-specific optimisations which are, of course, non-portable.

Another example of indirect co-design with vendors occurs during the procurement of new hardware, many HPC Centres employ their own benchmark suite reflecting the future target simulations. Vendors will then adapt the configuration of the underlying hardware to support the required performance of this suite. As such, for EXCELLERAT, one possible course of action is to ensure EXCELLERAT Applications and their associated Use Cases, are included in these benchmark suites.

Another example of indirect co-design with vendors occurs when centres, such as EXCELLERAT, gain access to the early release of state-of-the-art hardware, where this hardware is available typically due to a close working relationship between the vendor and the HPC centre which houses this hardware. This will permit EXCELLERAT to exploit trends in software and hardware and match them to the code design issues of our Reference Applications running our Use Cases.

This final example of co-design, albeit indirect, is the route that EXCELLERAT shall follow for T4.1. Indeed, during the all hands meeting (AHM) in CINECA, Bologna, in November 2019, it was realised that many logical sub-tasks of T4.1 are, in fact, existing tasks from other work packages, for instance,

- T3.1: Node-level performance optimisation,
- T3.2: System-level performance towards exascale,
- T3.4: Test lab for emerging technologies,
- T3.5: Validation and benchmarking suites,
- T4.1: Co-design,

- T4.3: Data analytics, and
- T5.5: HPC service provisioning.

As such, a Co-Design Working Group was established and will be managed by T4.1.

At this AHM, the Co-Design Working Group formed and produced an initial yet clear roadmap of work. Specifically, for each Reference Applications we will determine what are the current bottlenecks when running the target simulations on emerging technologies, where T4.3 will provide key linear algebra routines to test alongside our Reference Applications. T3.4 will help locate and document emerging technologies, T5.5 will provide access to cutting edge platforms or emulators of future platforms, and T3.5 will help to determine bottleneck kernels via profiling the reference applications. T3.1 and T3.2 will perform the node-level/accelerator tests. The POP centre of excellence [8] can also be exploited for this work.

To date, T4.1, thanks to input from the Co-Design Working Group, has agreed that the codes/libraries involved, at least initially, are NekBone (Nek5000), AVBP, and dense linear algebra solvers from ScaLAPACK. It was further agreed that we may consider ARM (via EPI and the original equipment manufacturers ARM and E4), field programmable gate arrays (FPGAs), graphic processing units (GPUs), and a NEC Vector Machine.

The initial plan currently contains options, including but not exclusively

- porting the three codes/libraries to existing hardware or emulators,
    - optimisation if time permits,
- initial profiling to locate kernels of interest,
- owner provides a benchmark for target hardware,
    - associated key performance indicators (KPIs) are created,
- profiling and performance measurements,
- using emerging libraries where possible,
- executing tests and measure KPIs.

Lastly, it is worth mentioning that a new co-design paradigm has started to emerge, and it is one which does not actually involve hardware developers. This new paradigm is where dynamic teams of specialists all work together towards preparing Applications for future hardware. These specialists can include the Application Authors, Specialists in the scientific domain itself, Workflow Specialists, and exascale Experts specialising in MPI, OpenMP, and accelerators such as GPUs, FPGAs, Quantum accelerators, etc.. Such dynamic teams can arise organically or by design, at Hackathons, Workshops Conferences, Webinars, etc. This new expanded definition is employed by several EU projects, such as E-CAM [9].

Future plans for this Task

- Finalise the Co-Design Working Group's Activities.
- Organising a joint Birds of a feather (BoF) session with EXCELLERAT and ETP4HPC [10] at ISC'20.
- Creating a crib-sheet for authors to prepare their codes for exascale, for portable optimisations that are not tied to any particular hardware.

4.2 Visualization

Analysis of the Market revealed that the is-situ libraries Catalyst [11] featured by ParaView [12] and LibSim [13] featured by VisIt [14] already found integration in a large number of simulation codes. From the analysis of the user-stories as well as from ongoing discussions within the consortium it was derived, that the situation with neither of the both libraries is currently satisfying for the user. Additionally, the problem exists, that both libraries do not share compatible interfaces. Therefore, the current objective of this task is to put effort into the generalization of in-situ visualization for the exascale engineering cycle.

The easiest way to address the same variety of simulations, is to integrate an adaptor to Catalyst's or LibSim's interface in Vistle[1]. A possibility how such a connection could be implemented is shown in Figure 7. In this case, the simulation is expanded by an adaptor code that can be treated as a Vislte-module running in the simulation's process space. In that way, the simulation can use Vistle methods to create the data objects needed for post-processing. The communication between simulation and module might use LibSim's interface or could be handled by a Catalyst adaptor script. If the simulation code already uses Catalyst or LibSim it does not need to be changed to use Vistle. The communication between module and Vistle uses Vistle's usual communication patterns which are shared-memory (SHM) within a node and message passing interface (MPI) between nodes. To set these communication channels up, the simulation sends its MPI parameters to Vistle which vice versa sends the SHM parameters back. This communication could be simply handled by some initial Transmission Control Protocol (TCP) messages.

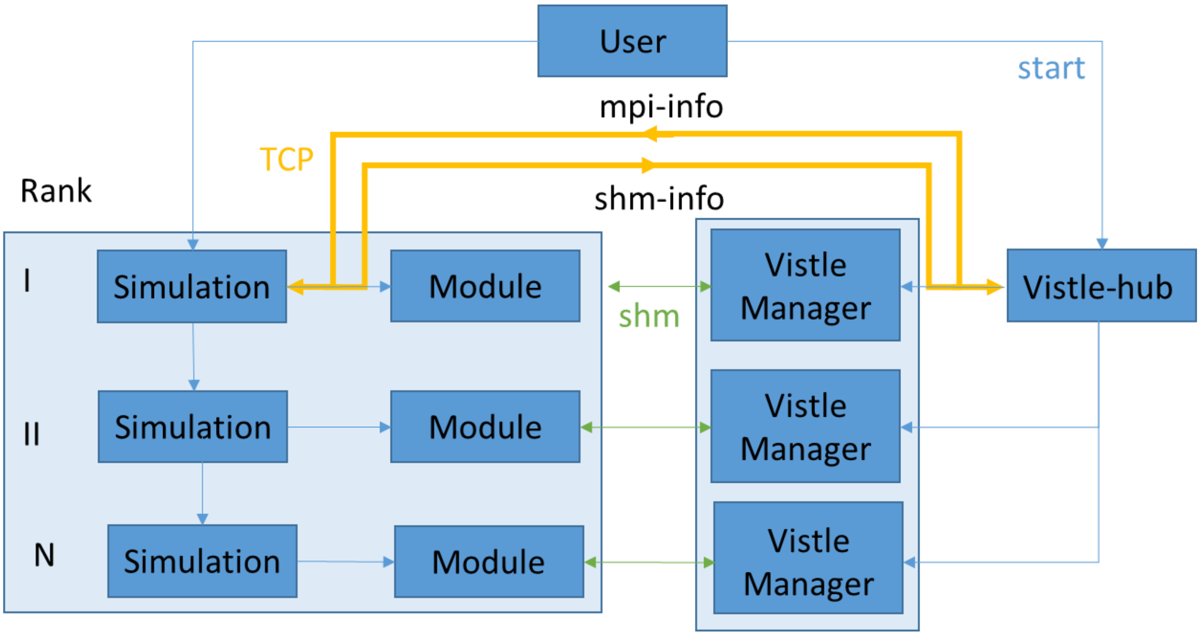


Figure 7: Approach to connect Vistle to a simulation Code.

A downside of this approach is less performance in case the reused interfaces between simulation and module require data transformations. This phenomenon must be evaluated with the example codes. If it requires too much computational effort the approach must be adjusted, e.g. by providing a special adaptor script for Vistle that avoids data transformation and copy as much as possible.

---

[1] https://www.hlrs.de/vistle

The first example code to get involved in testing the in-situ approach will be Nek5000. To achieve this, a data reader for Vistle had to be implemented. The reader is capable of reading Nek5000 data-sets parallel on up to as many nodes as the simulation has used. By using the .map file created by a Nek5000 simulation, associated data blocks can be merged together within a computation node. This reduced overhead in storage and makes the following computations on the data more performant. The .map file info can also be used to create Ghost Cells without much effort. Figure 8 shows flow through a suddenly expanding pipe simulated with Nek5000 and visualized with Vistle. The simulations single data output file was read parallel on 8 nodes. The outer surface is coloured according to pressure while the cutting surface displays the magnitude of the velocity within the pipe.

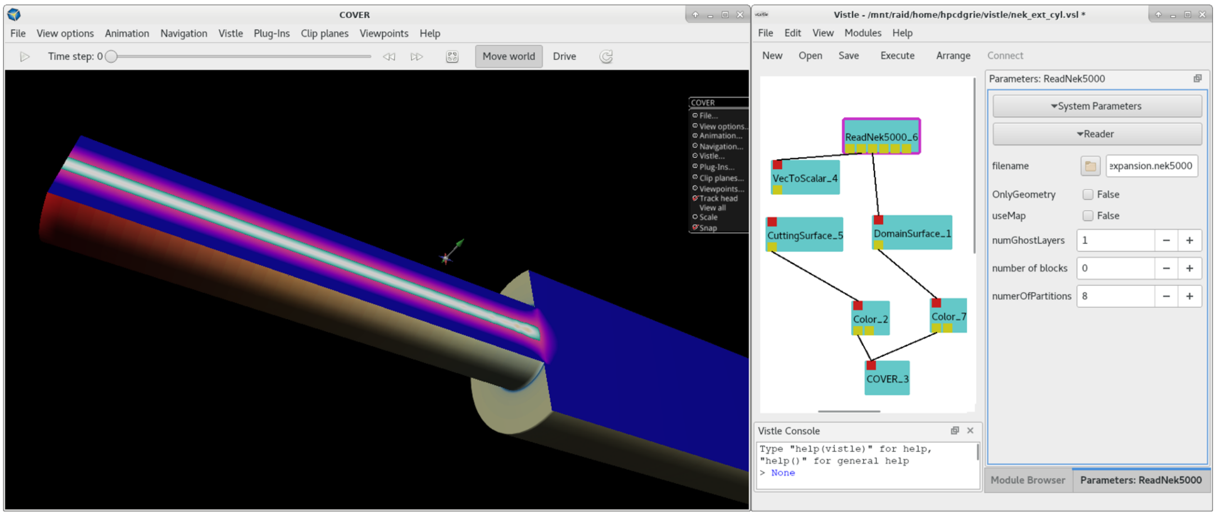


Figure 8: Visualization of the Nek5000 example simulation “expansion” created by Vistle. The right side shows the Vistle pipeline that produces the output shown on the left.

Next steps will be reading Nek5000 data like this during simulation using the approach presented above. The adapter module will first be designed to fulfil the needs of Nek5000, then those of the other example codes (as described in D4.1) and finally be expanded to fit more general simulations. Since some of ParaView’s or VisIt’s functionalities defer significantly from Vistle’s, general support for all of their in-situ-functions will most likely be too much effort. This and the above-described overhead in storage and processing power through unnecessary data transformations means that for full functionality and best performance adapter modules designed for Vistle must be implemented. In cooperation with Tasks 3.1 and 3.2 a guideline to design a performant exascale ready adaptor module will be created.

Furthermore, the other services described in D4.1 will be implemented.

4.3 Data analytics

4.3.1 Dedicated Tools for In-Situ Analysis (FEniCS – C5U1)

The goal of in-situ analysis is a reduction of data in- and output by processing the simulation results as long as they are still in memory. This allows an identification of trends during the simulation and monitoring the results depending on input parameters. While in the past, this has been done for simple scalar quantities or numerical residuals, the work in EXCELLERAT focuses on the in-situ calculation of more global spatial and temporal structures in the solution by data-driven techniques.

So far, most efforts were put into setting up an interface between general Machine Learning (ML) libraries (e.g. skit-learn [15], Tensorflow [16]) as well as proprietary tools developed by Fraunhofer to a representative CFD workflow with OpenFOAM [17] and Catalyst [11]. Two basic concepts are followed to design the software infrastructure. First, an intuitive Python API is defined to allow a simple reuse of existing ML routines and a fast cross-validation of different techniques and parameters during algorithm development. Second, all time-consuming calculations are performed by optimized mathematical libraries that are implemented in pure C/C++ or Fortran. This approach provides both usability and a high efficiency at the same time.

Furthermore, different kinds of data representations (e.g. wavelets, spectral bases, data-driven decompositions) where investigated regarding their applicability as input features for Machine Learning. They were compared regarding compactness as well as their ability to separate physical effects. Thereby, data-driven features, such as principal components, can be superior for a given dataset but usually lack general applicability. On that account, a database of suitable representations for different kinds of flow types will be built in the future. Further work will focus on setting up a client-server infrastructure based on an IPython server [18] to allow an interactive analysis of the solution during runtime. A connection to EXCELLERAT's core code FEniCS will be set up to demonstrate the general applicability of the software infrastructure and algorithmic approaches.

4.3.2 Dedicated Tools for Comparative Analysis (FEniCS – C5U1)

Software tools for comparative analysis of simulation bundles aim to assist the engineering design process. Thereby, data-driven techniques are applied to identify similarities, e.g. characteristic spatial or dynamical structures, in the flow. These results will lead to a better general understanding of the flow behavior and indicate relevant flow regions that can further be analyzed by more quantitative approaches. Bundles of simulations with different boundary conditions, numerical settings or design parameters, are evaluated to find dependencies of the flow behaviour based on these inputs. Data-driven models can be applied to predict the behaviour for parameter settings where no high-resolution data is available.

From a technical perspective, the focus has been on the application of unsupervised Machine Learning techniques. These have the advantage that the identification of patterns is not restricted to an apriori defined set of features. Instead, the results will be arranged into clusters of similar behaviour that can be investigated by engineering experts. To find low dimensional representations of spatial flow snapshots, different architecture of neural networks, non-linear manifold learning as well as topological data analysis have been applied to the large eddy simulations (LES) of two simple flow fields, the flow around a cylinder and a channel flow with an obstacle.

While evaluation is still in progress, the methods seem capable of archive compact and physical meaningful low-dimensional representations of spatial snapshots and the dynamics of the system. Examples for features are bifurcations, coherent and periodic structures, boundary layer instabilities or transient behavior towards a quasi-periodic state. Additional research will be conducted to investigate data-driven approaches from theoretical background to build more trust in these methods as well as providing a physical interpretation of the results. Both are inevitable in an engineering context. To demonstrate results on a representative industrial use-case, the developed tools will be applied to the use-case data of FEniCS. Investigating the influence of changes in the car's geometry onto the transient flow behavior is very challenging and of great practical interest.

4.3.3 Dedicated Tools for Calculation of Modal Decompositions (Alya – C2U1)

Highly accurate, turbulence scale resolving simulations, i.e., large eddy simulation and direct numerical simulation, have become indispensable for scientific and industrial applications. Due to the multi-scale character of the flow field with locally mixed periodic and stochastic flow features, the identification of coherent flow phenomena leading to an excitation of, e.g., structural modes is not straightforward. A sophisticated approach to detect dynamic phenomena in the simulation data is a reduced-order analysis based on dynamic mode decomposition (DMD) or proper orthogonal decomposition (POD).

In order to be able to perform the analysis of the large amounts of data that will arise from the use cases in the EXCELLERAT project, a key component in the software tools is an efficient and scalable parallelization concept. In the first year of EXCELLERAT, the already existing software tools have been optimized using MPI and ScaLAPACK to efficiently perform modal decompositions in parallel on large data volumes. DMD and POD are data-driven decomposition techniques, for which the time resolved data has to be read for the whole time interval to be analysed. To handle this large amount of input and output, the software tools have been optimized to effectively read and write the time resolved snapshot data parallel in time and space. Since different solution data formats are utilised by the use cases in EXCELLERAT, a flexible modular interface has been developed to easily add data formats of other simulation codes. In addition, the modular architecture enables to easily include further analysis tools into the software to deal with prospective future customer requirements. For the integration of the analysis tools into the visualization concept developed in WP 4.2 within EXCELLERAT, interfaces have been defined such that the DMD and POD modes can be visualized.

4.3.4 Data-driven modeling of Turbulence (Alya – C2U1)

Related to data-driven approaches in turbulence modeling, two main questions where investigated. Firstly, which data flow case should be used for training a machine learning model, and for which flows is the resulting model applicable? And secondly, what choice of flow variables are to be included in the data for training the models?

Towards answering these questions, we have considered three main categories of inputs for building and evaluating gradient boosting ensemble models, using data from LES simulations of fully developed turbulent flows in a channel, at Re = 1000 and Re = 180. Three models were trained on data with inputs based on primitive flow variables, physics-informed variables and dimensionless variables, respectively. Their performance has been evaluated by first training on a flow of lower Re and tested on a higher Re flow, and then vice versa. For the initial test case, all the trained models underestimated the mean streamwise velocity profile, with only the model trained on dimensionless inputs predicting the lower part of the profile accurately than the other models. When trained on data from a higher Re and tested on a lower Re case, the models built using physics informed and dimensionless variables, captured the underlying physics and produced predictions that followed the law of the wall quite accurately. The performance of the models has been further tested on a scaled version data of Re=180. Only the model trained on dimensionless variables consistently predicted the mean velocity profile of the modified test data. The results of the predictions and the residuals have been further analyzed to confirm the consistency of the outputs of this model.

4.3.5 Dedicated Tools for Quantification of Uncertainties (NEK5000 – C1U1)

Assessment of numerical uncertainties in turbulent flow simulations

There are various sources of numerical uncertainties which can affect the simulations of turbulent flows. To assess the uncertainties, special UQ techniques are required, considering the complexity of the Navier-Stokes equations and the high computational costs of the scale-resolving simulations of turbulent flows.

In this regard, we have been studying the influence of relevant numerical parameters on the accuracy of the wall turbulence simulations by Nek5000. These parameters include the elements size, number of collocation (GLL) points per element, filtering method and filtering parameters. The QoIs (quantities of interest) are defined to be different flow statistics. An in-house code has been developed based on the combination of surrogate models, generalized polynomial chaos expansion (gPCE) and Sobol indices in order to:

- Estimate statistical moments of the uncertain QoIs when numerical parameters are allowed to simultaneously vary.
- Estimate sensitivity and robustness of the flow QoIs with respect to the uncertain parameters.
- Investigate the accuracy of the QoIs through comparison with reference data.
- Draw a set of best-practice guidelines for accurate simulations of canonical wall turbulence with Nek5000.

The methodology is general and designed to interact with a CFD code in a non-intrusive way. Therefore, the framework can be used for assessment of accuracy, robustness, and sensitivity of any other computational code.

Estimation of uncertainties due to finite time averaging

For reliable computation of the flow statistics, averaging over a long time span is mandatory. The fundamental question is that how much uncertainty is involved in the statistical moments when time averaging is limited. We have been aiming to develop fast and accurate tools for estimating this type of uncertainty. A framework has been developed in which the performance of these techniques is compared. The final aim is to possibly implement the best technique in Nek5000 statistics toolbox for on-the-fly estimation of uncertainties due to time averaging.

4.3.6 Training Modules for Data Analytics

Training modules are designed in a modular concept covering basic and advanced data analytics topics. In the context of the Fraunhofer Alliance for Big Data [19], a large variety of data analytics workshops are already offered outside of EXCELLERAT. These will form the basis for advanced trainings that focus on specific questions related to the analysis of engineering simulation data aiming at industrial end-users. The corresponding educational material will be created parallel to the technical developments of data analysis tools in EXCELLERAT and will be available for trainings towards the end of the project.

4.4 Data management

The general goal is to combine data transfer and data management. The vision is to provide a new software solution, on which the data, that needs to be calculated, is uploaded, sent to the cluster, compiled and executed. Further services could be:

- Possibility to interact with cluster through a command line interface.
- Visualization of result data.
- Data transmission in encrypted form.
- Fast data transfer due to a data reduction technique.

Visual feedback on cluster allocation in form of a dashboard platform will be connected to all HPC centres in the project. At any time, there should be traceability of what happens to the data or where the data is located.

In the first few months the goal is to develop a prototypical application in which a first real HPC use case can be mapped. The first use case will be a ZFS [20] use case from RWTH

Aachen, that will be implemented into the system. That means, the source code of the solver will be compiled and integrated into the software. When all configuration files are available, the actual solver execution will take place and the result data is going to be transferred back. For the future, the following feature could be implemented:

- Compression of the returned data.
- Data encryption of the transferred content.
- Addition of more solvers.
- Possibility to connect all HPC centres.

After successful implementation of the ZFS use case, further use cases are to be integrated.

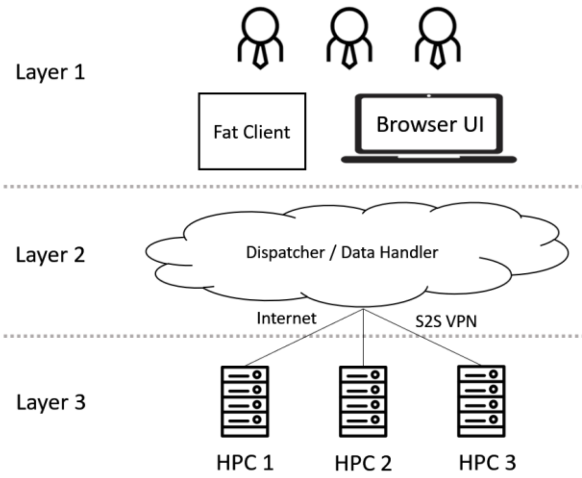


Figure 9: Conceptual Model

Figure 9 gives an initial overview of how the new system could look like. The complete service should be distributed over three layers. The first layer are the users and the available interfaces they can use. For example, there should be a web interface and a small client. The web interface could be used to handle smaller amounts of data and the client should perform more complex actions like delta building or data compression. With both interfaces it should be possible to upload data, configure jobs and download or view the result data.

In the second layer, a central distribution mechanism would be installed. This would be a data dispatcher which would also be responsible for data management. This layer can either run in one of the HPC centres or in the cloud, where a very high data throughput is possible - for example Google or Amazon cloud services. Of course, the data would not be stored there and only passed to the appropriate HPC centre. The transition between the second and the third layer could be controlled either directly via the Internet or, for example, with a Site-2-Site VPN.

The lowest layer is formed by the individual HPC centres on which the code is executed and each of these centres needs to run a small application, in order to communicate with the second layer.

Furthermore, the creation of data deltas could contribute to a good data management system. Each file will have a unique content identifier which is built for example by 1 MB blocks and each of these files has its one hash. In the end there is one big tree with all the hashes. That means if a file changes, only the changed blocks and not the whole file has to be transferred.

Figure 10 represents a first draft of a general HPC workflow, which should be transformed onto the new software.

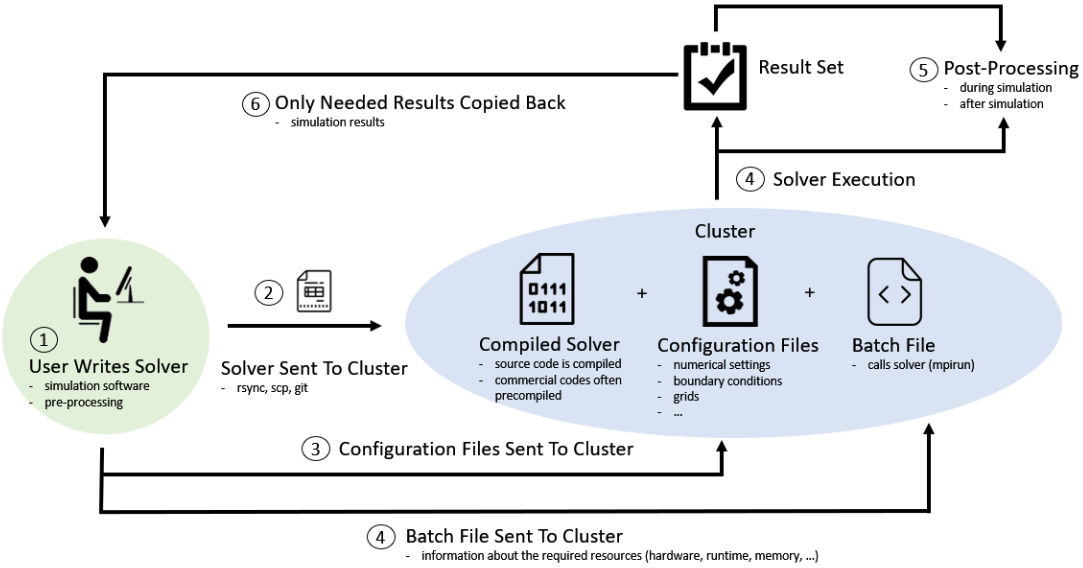


Figure 10: HPC Workflow

The following description will give a technical overview of how the all-in-one platform is built and which technology is used.

The platform is based on a container-based infrastructure and microservices. Microservices are small, autonomous services that have a single job and work together. In order to run the networked services in a secure and connected way, Istio is used as service mesh.

The container architecture in this case is Docker, which is managed by Kubernetes. Kubernetes is an open-source system for automating deployment, scaling and management of containerized applications. To operate and scale the Kubernetes cluster on an infrastructural level, the cloud service provider from Google Cloud is used at the moment and the package manager Helm [21] is used to provide applications into Kubernetes.

The programming languages used in the repositories are mainly Java and Typescript. The following markup, style and script languages are also partly used: HTML, CSS and JavaScript.

The whole source code of the platform is managed by a self hosted GitLab. GitLab offers a location for online code storage and collaborative development of software projects. Figure 11 gives an overview of the project structure.

Each project folder is responsible for single functionality in the platform. For example, the "web-ui" provides the web server Nginx, that stores web site files and broadcasts them over the internet. The "gateway" and "projects-query" folders contain various applications like MongoDB or Micronaut.

Micronaut is used as JVM-based, full-stack framework for building modular microservice applications. Gradle is used as build tool behind, that assembles the individual components into finished JAR files, which are then transformed into docker images using the Java Jib plugin. In order for all components to be built successfully, a separate bash script is used to build finished docker images from the gateway, project, and Web UI components.

Figure 11: Project Structure

For storing all the data, the two databases MongoDB and Neo4j are implemented. The document database MongoDB stores data in JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time. On the other side, Neo4j is an open source graph database management system. A graph is a pictorial representation of a set of objects where some pairs of objects are connected by links. It is composed of two elements - nodes and relationships.

In order to publish and subscribe to streams of records or to store and process streams and events, the platform Apache Kafka is used. It also provides multiple interfaces for writing data to Kafka clusters, reading data, importing and exporting data to and from third-party systems and it acts as a messaging system between the sender and the receiver. The Kafka client sends any event into the project queue, which can then be consumed by anyone.

To enable messaging, in order to connect and scale the all-in-one platform, the message broker RabbitMQ is used. It is a message-queueing software to which all HPC centres of the all-in-one platform are connected to.

4.5 Usability

The general goal of the usability task is to provide workflow and best practices for an engineering simulation's entire life cycle, from pre-processing, including modelling and meshing, execution of simulations to post-processing. In order to derive efficient workflows, this task gathers the expertise and methods developed in the CoE. In the first year the main focus has been on pre-processing workflows, in particular formulating a meshing workflow for complex geometries using hex-based meshes. This was a requirement from the C1U1 use-case, but the derived workflow described below, applies to any code with hex-based meshing.

One of the main goals of EXCELLERAT work devoted to Nek5000 is to improve the adaptive mesh refinement (AMR) version of this code to make it a robust solver proper for simulating industrially relevant flows. The capability of dynamical mesh adaptation during the run is a desired feature of any solver, as it allows to control computational error making the solver more robust and reliable. It simplifies as well meshing process providing flexibility of nonconforming meshes. However, at the same time it makes meshing more demanding, as efficient AMR requires relatively coarse initial mesh, that still properly represent domain geometry. This can be a challenging task for Nek5000, as this solver is based on Spectral Element Method and requires high order, hex-based meshes. In the following paragraphs, we will shortly describe the developed workflow for generating such meshes for EXCELLERAT C1U1 test case: NACA0012 aerofoil with rounded wingtip. We focus mostly on proper representation of object surfaces (e.g. wing surface) and division of the computational domain into a set of hexahedral sub-domains.

Nek5000 package provides meshing software suitable for spectral element method called pretex, but this tool has significant limitations when it comes to surface representation and three-dimensional visualisation, so it is not suitable for general meshing of relatively complex objects. To generate a wingtip mesh, we have chosen *gmsh*, as it is an open source software and there is a mesh converter *gmsh2nek* available on *github*. The main drawback of this choice is the fact, that *gmsh* is not a multi-block mesher, so the whole meshing process could require more effort. On the other hand, this software provides flexible scripting language which is a useful completion to a graphical interface.

We used *gmsh* version 4.1.5 [22] and we found several code limitations when it comes to generation of hex-based mashes. The first one is the lack of three-dimensional algorithm for mesh recombination, which means that a number of automatic meshing tools cannot be used in three-dimensional cases. On the other hand, two-dimensional algorithms for recombination of triangular meshes into quad ones are effective and can be used. To build consistent hex-based three-dimensional mesh we had to use *Transifinite* algorithm, which allows building volume and surface meshes based on defined split of edges. Although this gives the user full control over e.g. element aspect ratio, it is very labour-intensive as the user has to work with low-level objects like volume edges and split them manually building next higher-level objects. In addition, there is no clear way of preserving edge properties when duplicating higher-level objects like volumes by e.g. translation or symmetry reflections, so the edges of newly created objects have to be split manually. The other important operation is geometry consistency check done by *Coherence*, which replaces multiple overlapping object with the unique ones. The object properties are not always preserved in this case and the edge *Transfinite* split has to be sometimes redefined by hand. Because of this, one should rather avoid building higher-level objects first and next operating on them. The more efficient way is to generate all the edges first and next group them into surfaces and volumes. Although this order of operations is not very critical for a graphical interface, it becomes important when scripts are used.

The other issue is related to the representation of the surface geometry. Although the surface of the wing with rounded tip can be easily generated even with built-in kernel using *Extrude* (Figure 12 a), the generated this way split into sub-surfaces (marked by different colours in the plot) makes it difficult to generate coarse mesh close to the singular point located at the end of the trailing edge. It is because each mesh cell must be entirely contained within a single sub-surface, and to avoid very skewed elements one has to split the wing surface in a different way, e.g. the one shown in (Figure 12 b). Unfortunately, the built-in kernel does not provide much control over the shape of such surfaces, and even operations like:

Surface(tip_srf_up) = {crvl};

Curve{wtip_crv_list()} In Surface{tip_srf_up};

which should embed a set of curves in the surface, do not guarantee expected surface appearance. OpenCASCADE [23] kernel is much more flexible in this case, so defining a set of points and including them in the surface with

Surface(tip_srf_up) = {crvl} Using Point{wtip_pts_list()};

can significantly alter the result. Unfortunately, this method is far from perfect, as in the more complex cases one can notice significant oscillations of the surface (Figure 12 b). These oscillations can be suppressed by careful tuning the size and shape of the sub-surfaces defined with points, but this usually increases mesh complexity (see Figure 12 c) and does not remove corrugation completely. The best way to define surface shape seems to be the generation of the set of curves immersed in this surface and next combining them through the *Wire* and *ThruSections* operations (Figure 12 d):

crvl = newll;

Wire(crvl) = crv_wtip_wsrfc(0);

For il In {0:npts/nsec-2}

Wire(newll) = crv_wtip_wsrfc_shape_up(il);

EndFor

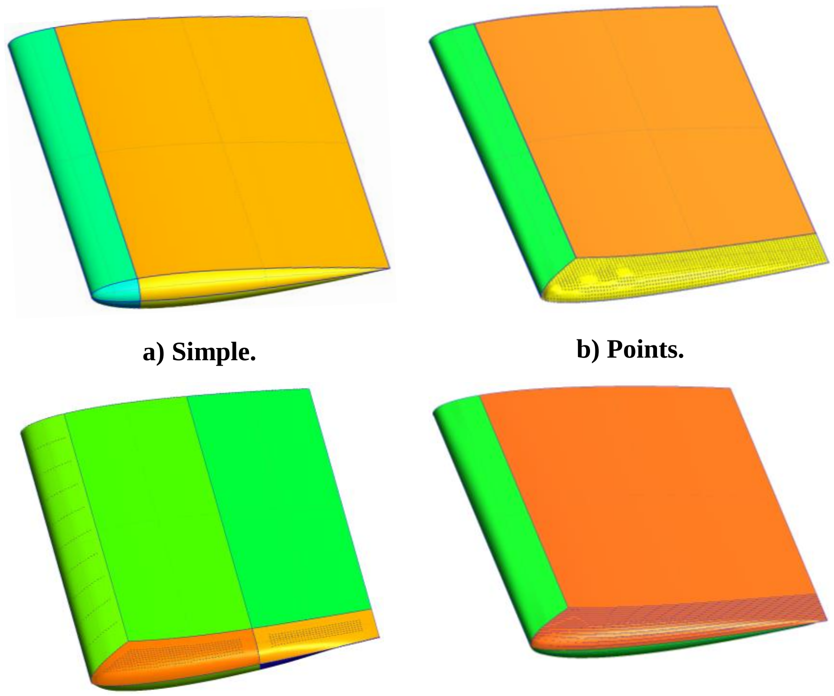
Wire(newll) = crv_wcut_wsrfc(0);

srf_wing_surface() = ThruSections{crvl:crvl+npts/nsec};

Unfortunately, this method has an important drawback, as it increases significantly the cost of mesh consistency checks done by *Coherence*. This step is actually necessary, as *ThruSections* duplicates surface edges and *Coherence* is the only way to restore mesh consistency. We have tried to minimise the problem by applying this operation to required mesh part only (e.g. common edge of two surfaces generated with *ThruSections*) and replacing *Coherence* with

BooleanFragments {Surface{srf_wing_surface(0),srf_wing_surface(1)}; Delete; }{}

but it did not bring much improvement. That is why we generated the meshes for the wing vicinity and the bounding box separately and merged them later with Nek5000 tools.



a) Simple. **b) Points.**

c) Multiple points. **d) Points and curves.**

Figure 12: Representation of wing tip surface for three different methods: a) built in kernel with Extrude; b) OpenCASCADE kernel with use of points: minimal required surface split; c) tuned sub-surface size, shape and number of points; d) OpenCASCADE kernel with use of curves and *ThruSections* (curve). Different colours represent wing sub-surfaces used to generated mesh volumes. Points a) and b) and curves d) used to define wing surface are presented in the suitable plots. The surface corrugations are clearly visible in a).

The wing vicinity mesh is visualised in Figure 13. We have to mention here, that regardless of the method we used for representing the wing geometry we had to finally project the surface points in Nek5000.

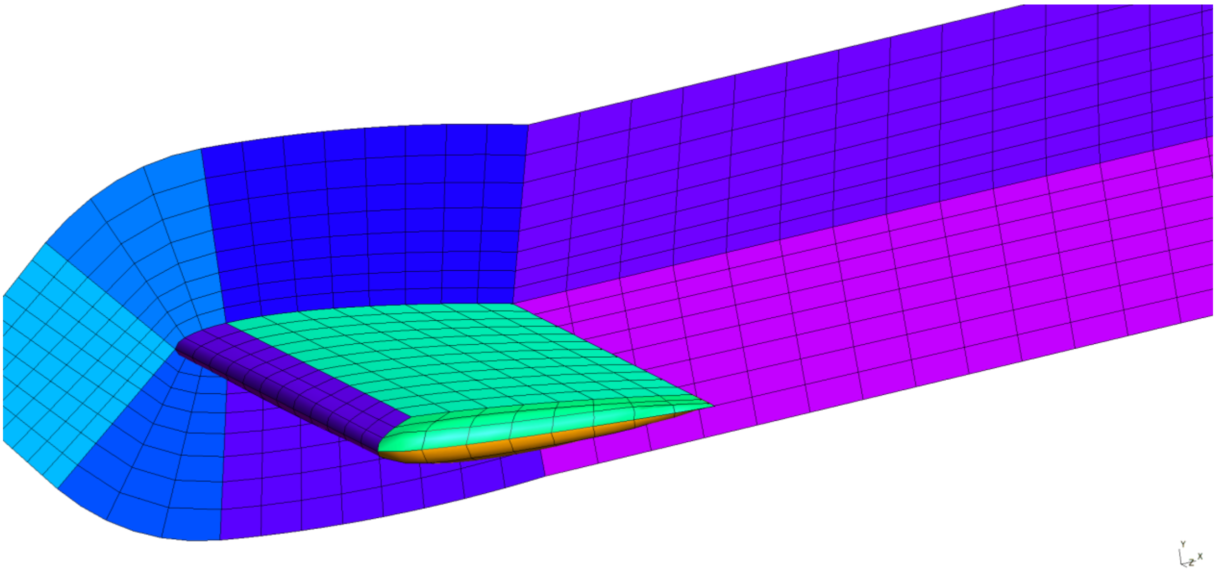


Figure 13: Structure of the mesh in the wing vicinity. Element borders are marked with black lines. Colours represent faces of different mesh volumes.

The overall workflow for mesh generation looks as follows: generation of inner and bounding box meshes with gmsh; exporting to the required format (order 2, version 2, not all elements saved); converting to Nek5000 format with gmsh2nek; smoothing inner mesh part with Nek5000 smoother [24]; merging inner and bounding box meshes with Nek5000 tools; smoothing the final mesh with Nek5000 smoother. This mesh was used to perform AMR test runs of NACA0012 aerofoil [6] use case with the Reynolds number equal 10000. The initial results are presented in Figure 14**.**

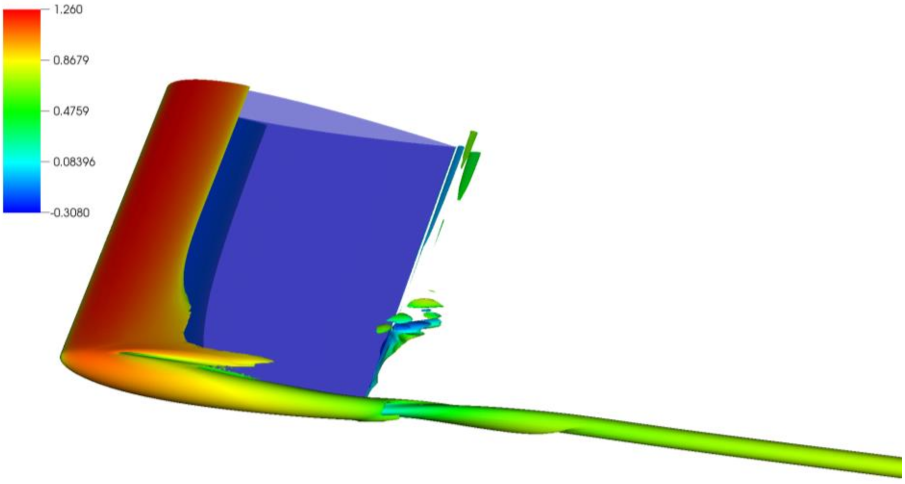


Figure 14: Vortical structure in the flow over NACA0012 aerofoil with rounded wing tip given by λ_2 criterion (isosurface). Colour gives value of streamwise velocity component. Hlf-transparent surface marks position of the wing tip. The tip vortex is clearly visible

5 Conclusion

The Evolution of EXCELLERAT's service portfolio is driven by two main objectives:

(1) To derive a service portfolio that covers to the largest possible extend all requirements and especially all user-groups targeting exascale engineering applications as represented by EXCELLERAT's use-cases.

(2) To identify the services with the largest impact on the engineering community by proper analysis, categorization, market assessment and community building activities.

This will help to further on refine the prioritisation for the implementation of the services by which the centre's sustainability can be achieved after its funding period.

In this deliverable the methodology applied to extend the originally envisaged service catalogue by service requests posed from different perspectives was described in section 2 along with the transfer towards EXCELLERAT's initial service layout which will be implemented via the EXCELLERAT portal. In summary nine services were identified in six service categories which will be accessible via four different service perspectives. The relation between services, service categories and service perspectives is described in section 3 and illustrated in Figure 2.

Section 4 was then left to describe the technical work conducted in the tasks of WP 4. With respect to this as already elaborated on in section 1, the conclusion has to be draw, that the report on the service portfolio has to be moved to WP 1. In exchange additional reports on "Best Practices and Tools for Visualization, Data Management and Analytics of Engineering Simulations at Exascale" will be introduced within the responsibility of WP 4. Additionally, it can be concluded that in the next project year the close integration of the use-cases with the technical developments done in WP 4 can be further extended based on the work already done. This will help to enable the efficient applicability of the engineering cycle on exascale HPC systems.

6 References

[1] The EXCELLERAT consortium, "EXCELLERAT grant agreement," EC, 2018.

[2] NASA, "NASA Common Research Model," 10 9 2019. [Online]. Available: https://commonresearchmodel.larc.nasa.gov/. [Accessed 29 11 2019].

[3] Y. Yokokawa, M. Murayama, T. Ito and K. Yamamoto, "Aerodynamics of High-Lift Configuration Civil Aircraft Model in JAXA," *Journal of the Japan Society for Aeoronautical and Space Sciences,* vol. 55647, pp. 563-571, 2007.

[4] T. Ménarda, S. Tanguyb and A. Berlemonta, "Coupling level set/VOF/ghost fluid methods: Validation and application to 3D simulation of the primary break-up of a liquid jet," *International Journal of Multiphase Flow,* vol. 33, no. 5, pp. 510-524, 5 2007.

[5] K. P. Geigle, R. Hadef and W. Meier, "Soot Formation and Flame Characterization of an Aero-Engine Model Combustor Burning Ethylene at Elevated Pressure," *J. Eng. Gas Turbines Power,* vol. 136, no. 2, 2 2014.

[6] U. A. A. Group, "UIUC Airfoil Coordinates Database," 2019. [Online]. Available: https://m-selig.ae.illinois.edu/ads/coord_database.html. [Accessed 29 11 2019].

[7] G. I. Taylor and A. E. Green, "Mechanism of the Production of Small Eddies from Large Ones," *Proceedings of the Royal Society A,* no. 158, p. 499–521, 1937.

[8] The Performance Optimisation and Productivity Centre of Excellence in Computing Applications, [Online]. Available: https://pop-coe.eu/.

[9] E-CAM, "E-CAM - Ecole Polytechnique Fédérale de Lausanne," [Online]. Available: https://www.e-cam2020.eu. [Accessed 29 11 2019].

[10] ETP4HPC, "ETP4HPC - European Technology Platform for High Performance Computing," [Online]. Available: https://www.etp4hpc.eu/. [Accessed 29 11 2019].

[11] Kitware, Inc., "Catalyst," [Online]. Available: https://www.paraview.org/in-situ.

[12] Kitware, Inc., "ParaView," [Online]. Available: https://www.paraview.org.

[13] Lawrence Livermore National Laboratory, "LibSim," [Online]. Available: https://www.visitusers.org/index.php?title=VisIt-tutorial-in-situ.

[14] Lawrence Livermore National Laboratory, "VisIt," [Online]. Available: https://visit.llnl.gov.

[15] "scikit-learn," [Online]. Available: https://scikit-learn.org/.

[16] "Tensorflow," [Online]. Available: https://www.tensorflow.org.

[17] "OpenFOAM," [Online]. Available: https://www.openfoam.com.

[18] "IPython," [Online]. Available: https://ipython.org.

[19] Fraunhofer, "Alliance for Big Data," [Online]. Available: https://www.bigdata.fraunhofer.de/de/datascientist/termine.html.

[20] Aerodynamisches Institut - RWTH Aachen, “Numerische Strömungsmechanik,” [Online]. Available: http://www.aia.rwth-aachen.de/index.php?id=31&L=0. [Accessed 29 11 2019].

[21] Helm community, “Helm 3: The package manager for Kubernetes,” [Online]. Available: https://helm.sh/. [Accessed 29 11 2019].

[22] C. Geuzaine and J.-F. Remacle, “Gmsh,” [Online]. Available: http://gmsh.info/. [Accessed 29 11 2019].

[23] OPEN CASCADE SAS, “Open CASCADE Technology,” [Online]. Available: https://dev.opencascade.org/. [Accessed 29 11 2019].

[24] K. Mittal and P. Fischer, “Mesh smoothing for the spectral element method,” *Journal of Scientific Computing,* vol. 78, no. 2, p. 22, 2018.