NumPEx Application Demonstrator Form

This form must be completed by all proposals of application demonstrators (AD) seeking to be included in PC5 of the NumPEx project-see attached presentation.

Every AD must be focused on targeted developments addressing a scientific and engineering challenge problem, i.e., one that possesses a solution amenable to computational insight and that is intractable without exascale capabilities and/or capacities.

Please find below the list of points that will be used by NumPEx to evaluate the AD proposals and analyze them in order to identify a first set of meaningful cross-cutting algorithmic motifs , i.e, algorithmic methods that drive common patterns of computation and communication, including libraries and frameworks .

**Q1** Describe and quantify the science and engineering *Exascale challenges*.

* Why and how is it significant?
* What new Exascale capacity and/or capability does it require?
* The timeline and/or the roadmap of the demonstrator development

**Q2** Describe the *issues/barriers* and targeted development *needs in terms of Exascale challenges.*

* Physical models, algorithms, software components.
* Others, such as,
  + workflows execution and/or load balancing,
  + data logistics, in-situ data analysis and reduction, and efficient I/Os,
  + highly-accelerated architecture and technologies,
  + programming and execution environments.

**Q3** Describe and quantify the Application *team*.

* Human resources and expertise of the team.
* Application development methodologies of the team.
* Team involvement in the application demonstrator development.

**Q4** Describe and quantify the team’s experience in *leveraging HPC systems*.

* Use of GENCI, Euro-HPC and/or other international HPC resources.
* The current level of performance and capability of the application.

**Q5** Describe a *Figure of Merit* for your AD.

* Measure the rate of “science work” enabled by new performance and functionality.
* Identify possible dependencies and trade-offs between targeted developments and software components.
* Set base and stretch quantitative application development objectives.

AD Evaluation Procedure

For your information, here is the list of points that will be used by NumPEx to evaluate your Application Demonstrator.

### A. Significance

* Science and Engineering impact of the challenge problem.
* Challenge problem requires Exascale (capacity and/or capability).
* Aligned with the strategic priorities of the NumPEx stakeholders.
* Maps with national and European strategic priorities.

### B. Breadth

* Well-identified targeted developments and of their impact on the rate of “science work”.
* Maps with the expertise and the planned research and software technology developments in the NumPEx PCs.
* Foster collaborative developments across a number of NumPEx PC teams.

### C. Team experience and confidence in:

* Model, software development and deployment.
* Leveraging HPC systems.
* Addressing ambitious Exascale targets.
* Level of involvement of the applicative team in the development of the demonstrator.

### D. Impact on the Science and Engineering computational community

* Ability to deliver translatable solutions that impact the broader community.
* Synergies with national (e.g., ANR, PEPR, etc.) and or European projects (e.g., Euro-HPC CoE, Horizon-Europe, etc)
* Synergies with large international projects and/or consortia.

Algorithmic motifs

The goals of the algorithmic motifs are to accelerate the development and improve the portability of exascale applications and to reduce the development risk for the CSE application teams by :developing algorithmic and software components that embody the most common computation and communication patterns in and across the ADs; integrating and composing meaningful set of software components; investigating crucial performance trade-offs between software components.

An algorithmic motif can be instantiated in many ways depending of the ADs, each of them having unique requirements. It has important implications for both CSE software development methodologies and exascale systems, i.e., when designing and configuring programming models, libraries, system software, storage and communication systems, and other exascale system components. In other words, algorithm motifs require a co-design process to answer application relevant questions from different perspectives.,

Major factors informing the algorithmic motifs are the ability to: support a range of ADs, avoid imposing restrictions on how CSE application developers construct their algorithms and science-driven functionality; allow developers to interact with the software at different levels of abstraction.

**Possible examples of algorithmic motifs are:**

**Adaptive mesh refinement (AMR) algorithmic motifs,** in particular in the context of multi-physics and multi-scale applications, represent a higher-level abstraction that can be used to co-design a performance-portable software framework shared by many NumPeX multi-physics and multi-scale application demonstrators that solve systems of partial differential equations (PDEs) in simple and complex domains, and where possibly particles and/or particle mesh operations represent component physical processes: e.g., astrophysics and astronomy, multi-phase flow, combustion, plasmas physics and fusion modelling, accelerator design, additive manufacturing.

**Efficient exascale discretisation and communication motifs** for high-order finite element methods provide a high-level abstraction that can be used to co-design performance software, libraries and frameworks accelerating the development of a number of large-scale NumPeX applications — including coupled fluid flow and multi-physics simulations, cloud-resolving climate modelling, wave propagation in complex media, multi-scale coupled urban system — where practical efficiency is measured by the accuracy achieved per unit computational time and portability across exascale systems.

**Particle-based algorithmic motifs** enabling particle applications for exascale computing platforms through the development of co-designed libraries and light-weight frameworks that provide a similar portability layer as the AMR libraries. Such a high-level particle layer abstraction is shared by a number of NumPeX applications to describe physical systems, including: molecular dynamics; simulations using empirical models or the underlying quantum mechanics for particle interactions; cosmological simulations in which particles may represent an object or a cluster of objects and particle interaction is through gravity; and diverse plasma simulations on grids within a particle-in-cell framework to solve the interaction of particles with the electro-dynamic field.

**Online data analysis and reduction algorithmic motifs** focusing on process and data placements and communications areshared by many applications for which data movements are increasingly becoming the bottleneck and require high-rate flux and volume data generators (i.e., simulation and/or data acquisition streams) to be concurrently and information-based processed by cascades of complex data reduction and/or analysis operations pipelines, including information exchange via high-speed communication and data logistics. This includes multi-physics code coupling, large ensemble simulations to explore complex phase and parameter spaces, on-line information-based data analysis and adaptive sampling using machine learning, high-rate streaming data analysis and reduction using machine learning.

**Combinatorial algorithms in general and graph algorithmic motifs** is recognised as a rapidly growing area that will play a critical enabling role in numerous applications. The irregular memory access inherent to these algorithmic motifs makes them one of the hardest algorithmic kernels to implement on exascale systems. Core-elements include application demonstrators that drive the selection of key combinatorial kernels and integration of software tools developed, such as computational biology, computational chemistry, astronomy and astrophysics, high-energy particle physics and climate science; design and implementation of several variants of combinatorial kernels that play a crucial enabling role in those application areas, such as graph traversals, graph matching, graph colouring; sparse matrix ordering methods, graph clustering and graph partitioning with numerous applications in a variety of scientific computing contexts; and efficient implementations and portability on hierarchical distributed memory architectures representative of exascale platforms.

**Machine learning algorithmic motifs**, including physics-based and data-based methods, have profound implications for a number of computational, observational and experimental exascale science and engineering demonstrators, and thus are poised to have important implications for the design and use of exascale systems themselves. Core-elements include learning methods common across a number of NumPeX CSE application demonstrators, including deep neural networks of various types, kernel and tensor methods, decision trees, ensemble methods, graphical models and reinforcement learning methods.