FabSim3 for Automation and Tool Integrations in VECMAtk

FabSim3

VECMA

Verified Exascale Computing for Multiscale Applications

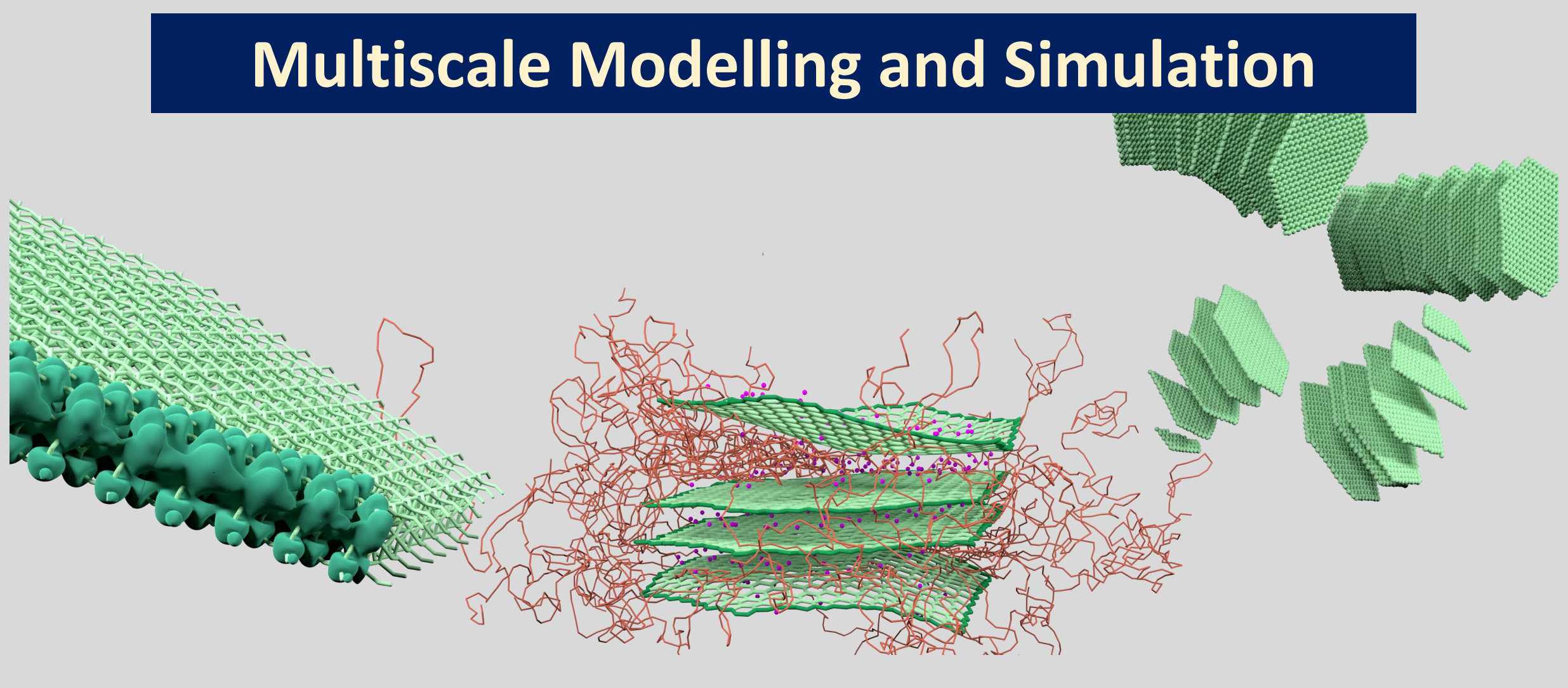
Project central objective is, for a diverse set of multiscale computing applications, to automate the stages of VVUQ by developing generic algorithms and approaches into an open source toolkit, exploiting the computational power offered by existing petascale and emerging exascale computing environments.

vecma.eu @VECMA4 VECMA FET-HPC

High-Performance Computing (HPC)

- High-performance computing has unlocked a new dimension in scientific research and simulation-based decision making.
- With this ever-growing forefront of computational power, we can simulate increasingly complex systems of interest.
- Exascale computing systems can perform > 1 exaFLOPS, or a quintillion calculations per second.
- We currently sit on the brink of the exascale.

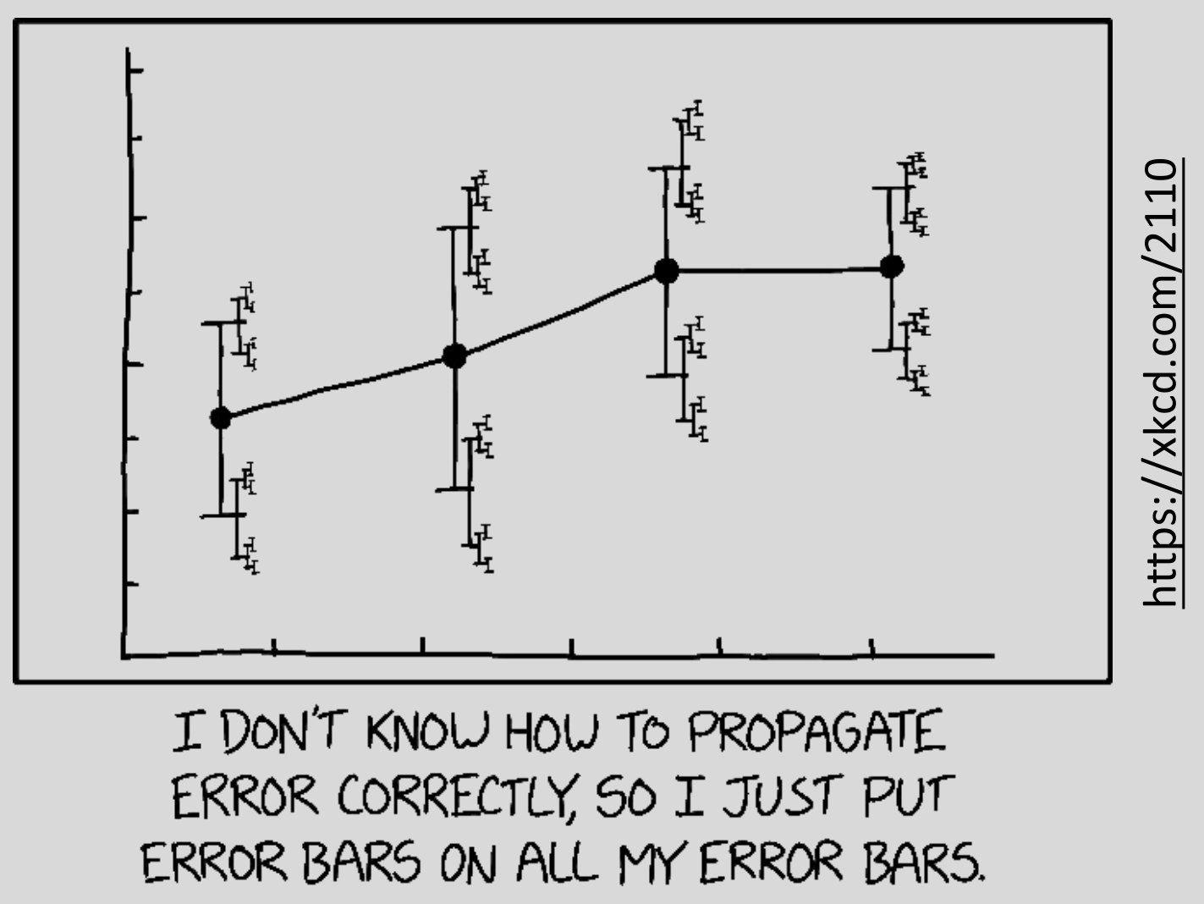
Multiscale Modelling and Simulation



- Systems of interest can be examined at a wide range of physical and temporal scales using high performance computing.
- Multiscale modelling and simulation combines different scales in order to make new discoveries and inform critical decisions.
- It is widely applied in fields ranging from the physical sciences, engineering, and the life science domain.
- VECMA project deals with the following domains: Climate Modelling, Migration Prediction, Material Science, Fusion Energy and Biomedicine.

VVUQ

- **VVUQ** - Validation, Verification, and Uncertainty Quantification;
- **Verification -** Determines if the computational model fits the mathematical description.
- **Validation -** Determine if the model accurately represents the real-world application.
- **Uncertainty Quantification** - Determine how variations in the numerical and physical parameters affect simulation outcomes.


I DON'T KNOW HOW TO PROPAGATE ERROR CORRECTLY, SO I JUST PUT ERROR BARS ON ALL MY ERROR BARS.

https://xkcd.com/2110

VECMA Toolkit

VECMAtk: An open-source Toolkit for multiscale VVUQ based on generic multiscale VV and UQ patterns (software solutions to reoccurring problems).

D. Groen, et al., "VECMAtk: A Scalable Verification, Validation and Uncertainty Quantification Toolkit for Scientific Simulations", Philosophical Transactions of the Royal Society A. 2020, https://arxiv.org/abs/2010.03923v2

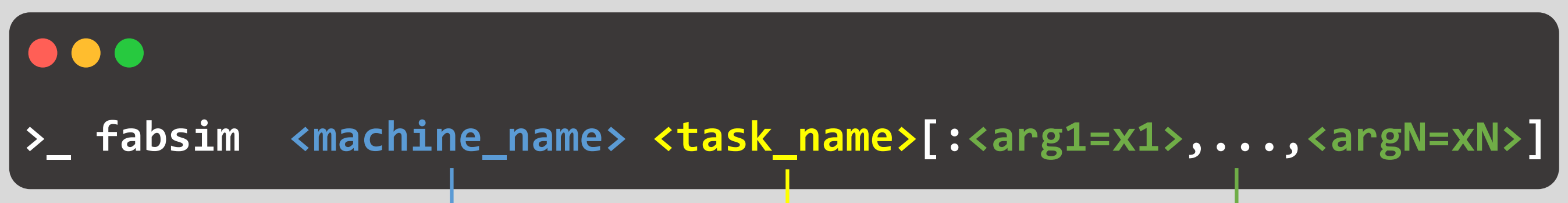
VECMAtk

Components:

- **EasyVVUQ** for VVUQ definitions;
- **MUSCLE3** for model coupling;
- **FabSim3** for automation and tool integrations;
- **QCG Tools** to manage and execute application workflow on HPC infrastructures.

FabSim3 Architecture

FabSim3 is designed to support to support automation of large scale simulation workflow from data preparation step to results analysis in such a way as to reduce the burden on application developers. The tool is generic and is oriented towards developers from different research disciplines and with at least basic programming experience. To enable users to rapidly prototype and evolve their domain-specific workflows, FabSim3 supports the development of application-specific plugins. Once developed, these plugins can then be shared with the wider community, eliminating the need to duplicate machine configurations, workflow definitions or deployment instructions.

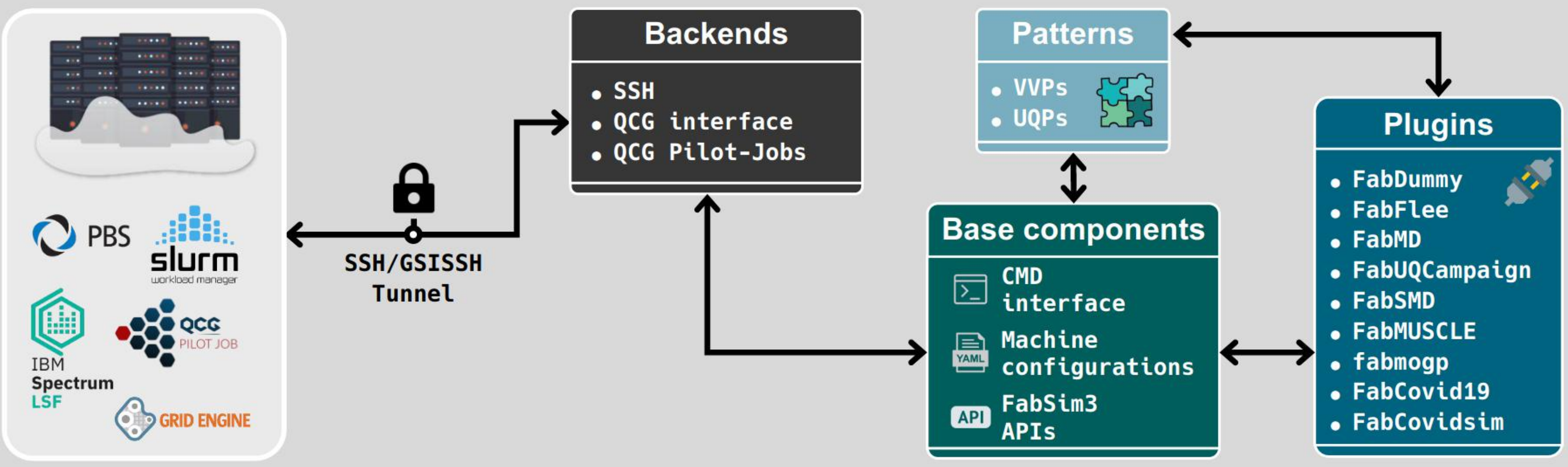


```
>_ fabsim  <machine_name> <task_name>[:<arg1=x1>,...,<argN=xN>]
```

The name of machine, i.e., `localhost` or `remote machine`,to be used for executing the simulation runs.

The name of FabSim3 API or plugin function.

Input arguments for the tasks or machines configuration variable.



FabSim3 Plugins

By abstracting the low-level required activities, FabSim3 features a flexible plugin system that allows users to have a customizable version of their own application, in a modular and lightweight manner, to be employed for automating simulation runs and performing tasks such as search, store, transfer, visualize, or any other analysis on the inputs/outputs data. This helps developers to curate complex and dynamic ensemble workflows in an easier way, and also make their simulations more robust and reproducible.

FabSim3 plugin API philosophy follows these three main goals:

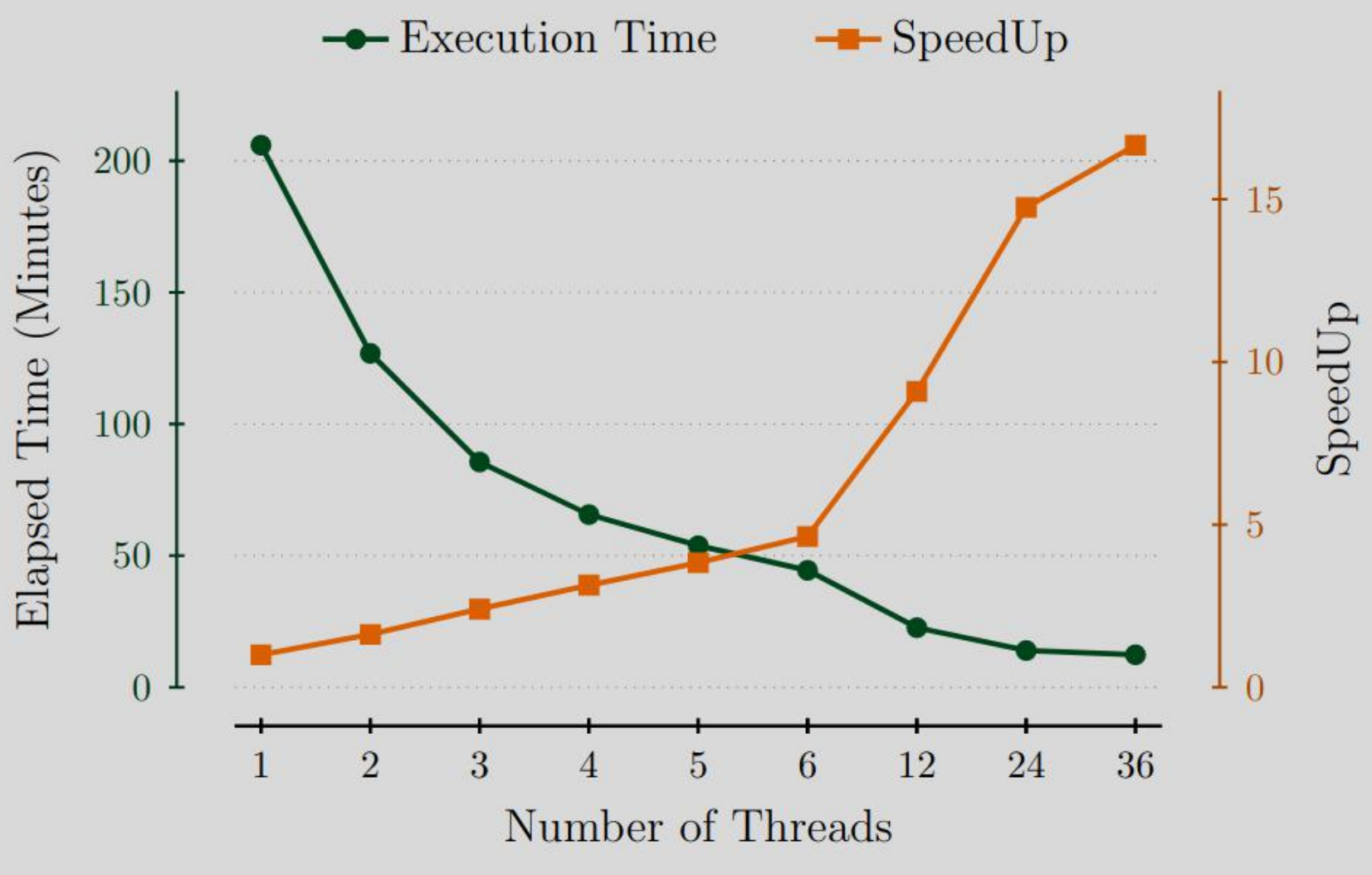
a) Stability:
   i. should not require any changes in the application core development, i.e., it should be only used as a wrapper functionality around the application.
   ii. the installation and usage should be applicable across devices and users.

b) Ease of Development:
   i. should be easy enough to develop and support.
   ii. should have a sufficient level of simplicity and generality for an average users.

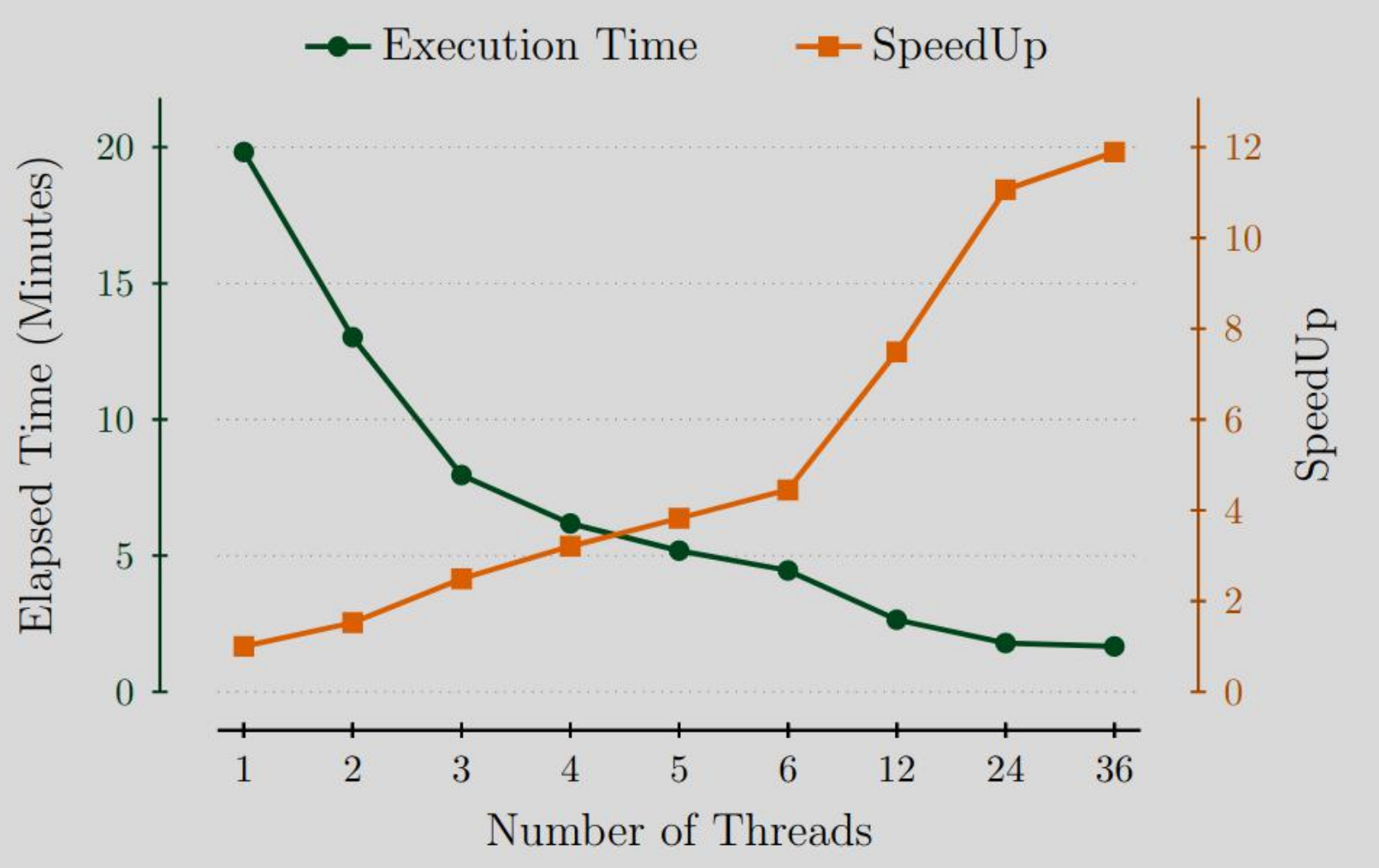
c) Performance:
   i. should provides a range of functionalities for common use cases.

FabSim3 Performance and Scalability

Time required to submit 15121/4865 jobs with FabSim3 (with/without QCG-PJ) relative to the number of job submission threads used. Graph is made using average of 10 repetition of each ensemble size. Please note that, here we only measure the job submission overhead, so, queuing time and job execution on computing nodes are not considered in our test.



ensemble size = 15121, QCG-PilotJob = True



ensemble size = 4865, QCG-PilotJob = False

Future Work

- Improving the total job submission;
- Supporting more VV and UQ patterns;
- Close integration of FabSim3 with QCG-PJ, EasyVVUQ and MUSCLE3.

The VECMA Consortium

UCL Brunel University London IPP CBK UNIVERSITY OF AMSTERDAM Atos CWI Centrum Wiskunde & Informatica PSNC lrz

VECMA Join us as an Associate Partner

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement ID: **800925**.

Coordinator: Prof. Peter Coveney, UCL, p.v.coveney@ucl.ac.uk
Project manager: Xuanye Gu, UCL, xuanye.gu@ucl.ac.uk

Runtime of the project: 15/06/2018 to 14/12/2021

Authors: Hamid Arabnejad[1], Diana Suleimenova[1], Derek Groen[1], Erwan Raffin[2], Paul Karlshoefer[2], Bartosz Bosak[3], Piotr Kopta[3] and Tomasz Piontek[3]

[1]Brunel Univesity London
{Hamid.Arabnejad,Diana.Suleimenova,Derek.Groen}@brunel.ac.uk

[2]Center for Excellence in Performance Programming ATOS
{Erwan.Raffin,Paul.Karlshoefer}@atos.net

[3]Poznań Supercomputing and Networking Center
{bbosak, pkopta, piontek}@man.poznan.pl