Initial proposal for joint access to NHR@GÖTTINGEN and NHR@ZIB

Replace the headline above with your official project title as given in the online portal.

Requested start date

1.1 or 1.4 or 1.7 or 1.10. 202X (The project duration is always one year.)

Type of proposal

Initial proposal

Whitelist

NO or

DFG / BMBF / NHR / GCS / EU project (*grant number #, start - end of funding*)

Whitelisting is only possible if the preceding proposal discusses numerical methods and explicitly describes HPC resource requirements. Acknowledged institutions are DFG, BMBF, NHR, GCS, and EU projects. Please provide the grant number (if assigned) and state the funding time frame. As a reference, you need to upload the reviewed proposal and its review report to the online portal.

Principal investigator

Name, affiliation, contact data

Project manager and contact

Name, affiliation, contact data

DFG classification (RB-Nr.)

313-02 Oceanography

Please provide the five-digit DFG classification of your project according to `www.dfg.de/en/dfg_profile/statutory_bodies/review_boards/subject_areas`. *Here, click on 'Show all Subject Areas' to see the last two digits.*

Short abstract

Your abstract should be generally understandable. It should be limited to this first page and written in one paragraph (without figures).

All italic text (such as this one) shall be removed/replaced appropriately by you!

The typical length of an initial proposal is about 13 pages (including the title and content page).
If whitelisted, the typical length of an initial proposal reduces to 6 total pages.
The maximal length of initial proposals is restricted to 18 pages (font 11 pt).

Initial proposals corresponding to whitelisted projects require processing only the technical part since their associated projects are scientifically reviewed already.

Contents

1 Scientific part *(initial proposal, not on the whitelist)*

In the case of whitelist status, you can delete and ignore all content/subsections of this section.

1.1 State-of-the-art

Give a short outline of the state of research, including references.
(about 0.5 to 1 page)

1.2 Preliminary work

Provide a brief summary of your preliminary work in connection with the proposed project, including references.
(about 0.5 to 2 pages)

1.3 Project objectives, impact, and novelty

(1 to 2 pages) What scientific questions do you want to tackle? What are the overall scientific goals of the project? It is important to describe the innovative aspects, impact, and topicality of the proposal.

- *Objectives and questions you want to address*
- *Scientific and technical innovation potential, and topicality*
- *Expected impact on the research area*
- *Expected outcome and progress beyond the state-of-the-art*
- *Planned exploitation of the results: publication etc.*

1.4 Detailed project description and work schedule

(maximal 3 pages in total) Describe your research project in detail, structured in work packages. Describe any critical risks, related to the work packages or overall project goals. Detail any risk mitigation measures. Provide tables for a clear presentation of the work schedule:

Work package	2022 Q4			2023 Q1			2023 Q2			2023 Q3		
	10	11	12	1	2	3	4	5	6	7	8	9
WP1 Finding bugs	■	■							■	■		
WP2 Squashing bugs			■	■	■	■						
WP3 Producing results on the supercomputer							■	■	■	■	■	
WP4 Dissemination												■

After the one-year term of your computing time project, you have access to your data for a further six months to download. Extensions are possible on request.

Work package 1

...

Work package 2

(about 0.25 to 1 page per work-package)

1.5 Model, numerical methodology, and implementation

(about 0.5 to 2 pages) If applicable, present your model, its assumptions, strengths, and shortcomings. Also, explain the equations or algorithms and their implementation underlying the software modules you intend to use.

1.6 Bibliography

(maximal half a page) Provide recent/most important bibliographic references [1, 2] that are relevant to the project. Mark authors who are also project participants with a bold font.

[1] **A. Momo**, B. Mimi, and C. Mama, *Study of Blibli*, Journal of Blibli, **15**, 43–62, 1965.

[2] A. Toto, B. Titi, and **C. Tutu**, *Effect of blibli on blublu*, Journal of Blabla, **468**, 77–105, 2002.

2 Technical part

2.1 Software description

(about 0.5 to 1 page) Describe all codes, packages, or libraries that you need to undertake the project, and how these will enable the research to be achieved. Include for each code to be used information about

- *Which code will be used*
- *On which hardware the code will be used (CPUs, GPUs, etc. or combinations, if applicable)*
- *How the code is parallelized (pure MPI, mixed MPI/OpenMP, Pthreads, CUDA, etc.)*
- *The amount of memory necessary (per core, per node, and in total)*
- *Current job profile (independent jobs, chained jobs, workflow, etc.)*
- *Describe requirements concerning the High-Performance network, and I/O requirements*

Important: please consider the corresponding technical guidelines and requirements (memory restrictions, etc.) of the chosen machine(s)!

If you use third-party codes, include

- *Name, version, licensing model, and conditions*
- *Web page and other references*
- *Contact information of the code developers.*
- *Your relationship to the code (developer, collaborator to main developers, end-user, etc.)*

2.2 Software performance and HPC suitability

(about 0.5 to 1 page) Proof that your code is ready for the specific HPC system(s)/module(s) for which you are applying. Typically, this includes

- *Description of the application performance in general and of the parallelization concept in particular*
- *Scaling plots or tables with speedup results with respect to parameter sets and problem size should be created. These runs should be representative of the jobs in the planned project, e.g. including data I/O. General benchmark results are not sufficient. Describe architecture, machine/system name, and problem size used for the scaling plots. Ideally, your scaling test should be executed -*

during a preceding 'Test-project' - on the same HPC architecture which is requested in this proposal.

Here we give an example plot for presenting scaling information. The scaling should range from a single core to the maximal possible number of cores. If this is not doable, the presented scaling should range from the lowest possible number to the maximal possible number of cores for your case.

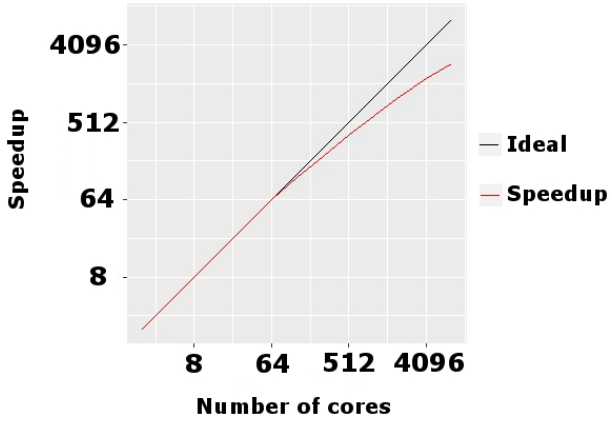

Figure 1: *Strong scaling test of* `code` *on* `architecture and system` *at* `location`*. This data was obtained with a problem size of* `size`*.*

2.3 The necessity to use an HPC system

(a quarter page) Discuss why the implementation of the project is not possible without the use of an HPC system: e.g. explain which hardware properties of your local university cluster are insufficient regarding XYZ of your code...

2.4 Requested compute resources

(0.5 to 3 pages) Justify the number of resources you request for the current granting period. Based on physical or stability reasons or otherwise, derive why exactly X steps/core-hours/storage requirements are necessary and why less is not possible.
Tabulate the following information for each type of requested resource (e.g. CPUs, GPUs, etc.) separately. Always use the unit core hours (core-h).

Table 1: *The following CPU-system resources are requested*

Work-package	Type of run	# runs	# steps/ run	Wall time/ step [h]	# cores/ run	Total [core-h]
WP1	Preprocessing	R1	S1	W1	C1	R1·S1·W1·C1
	Production	R2	S2	W2	C2	R2·S2·W2·C2
...				...		
TOTAL						sum of above

Work-package	Type of run	Problem size	Memory/ node [GB]	Work (+tmp) storage [GB]	Backup tape [TB]
WP1	Preprocessing	P1	M1	T1	B1
	Production	P2	M2	T2	B2
...			...		

Table 2: *The following GPU-system resources are requested*

Work-package	Type of run	# runs	# steps/ run	Wall time/ step [h]	# equivalent cores[1]/ run	Total [core-h]
WP1	Postprocessing	R3	S3	W3	C3	R3·S3·W3·C3
	Visualization	R4	S4	W4	C4	R4·S4·W4·C4
...				...		
TOTAL						sum of above

Work-package	Type of run	Problem size	Memory/ node [GB]	Work (+tmp) storage [GB]	Backup tape [TB]
WP1	Postprocessing	P3	M3	T3	B3
	Visualization	P4	M4	T4	B4
...			...		

2.4.1 Usage schedule

Break down the core hour consumption according to the work packages and quarters of the project. (half a page)

[1]For GPUs, please specify the resources in terms of CPU core hours: for this purpose, one of the available NVIDIA A100 GPUs is equivalent to 150 cores. Consequently, the usage of one entire 4-GPU node is equivalent to 600 cores.

Core-h usage schedule	2022 Q4	2023 Q1	2023 Q2	2023 Q3
WP1 Finding bugs	10^3		10^3	10^3
WP2 Squashing bugs	10^3	10^3		
WP3 Producing results on the supercomputer			5×10^6	5×10^6
WP4 Dissemination				
Total core-h demand per quarter	2000	1000	5001000	5001000

Compute hours are granted quarterly. Hence, all core hours need to be distributed over the four quarters of the one-year project period. On request, you can shift the quota from quarter to quarter during the one-year project term.