VORTECH

High Performance Computing

If your software needs to have a higher performance

When doing large calculations, you will easily run into the computer's speed and size limits. This can be remedied by changing your computational software to use the available hardware more intelligently, or by adapting the software for more powerful hardware systems. VORtech offers a range of services to assist your organization in benefiting from High Performance Computing.

scientific software engineers

The advantages

High Performance Computing (HPC) provides the following advantages:

- **Faster results** – Not only vital for operational predictions, it improves the work flow by reducing waiting time.
- **Higher accuracy** – A more detailed calculation is possible in less time.
- **Larger problems** – Bigger calculations that include additional features can be realized, thereby increasing the realism of your simulations.
- **More alternatives** – Many more calculations can be performed within the same time period, allowing for further optimization of product design by considering more variants.

In three steps to high performance software

- **The first step** in getting high performance computational software is to use the best solution methods available. This requires expert knowledge in numerical mathematics and algorithms.
- **The second step** is to assess the potential of introducing HPC optimizations. An extensive analysis of the algorithms and the code by experienced scientific software developers pinpoints the computationally intensive parts and identifies the best approaches for performance improvement. The analysis answers questions like: Where are the quick wins? What is the cost-to-benefit ratio of different strategies? The result is an advice for the best approach and a development plan for the implementation.
- **The third step** is implementing the HPC optimizations, which can consist of software and hardware. Depending on the chosen strategy and on the state of the code, the software adaptations vary from a few lines for compiler hints to a rewrite of the computational core routines. See the table below for an overview of HPC improvement strategies and their gains and required investments. In the Tech Section on page 4 these strategies are further explained.

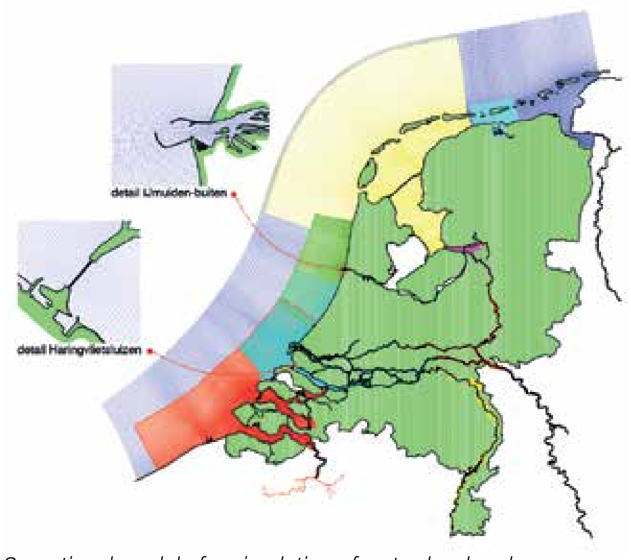
	Potential gains	Software investment	Hardware investment
Optimizing memory access	2-10x	●●	●
Improve single core performance	2-8x	●	●
Shared memory parallelization	2-20x	●	●●
Distributed computing	5-100x	●●●	●●●●
Massively parallel accelerators	10-100x	●●●●	●●●

Table: Performance improvement for computational core routines that can be achieved by applying different HPC improvement strategies, and required software/hardware investments. See also the Tech Section on page 4.

VORtech: experts in performance-aware scientific software engineering

Ever since its foundation, VORtech has been an expert company in the field of HPC. Some of the best practices that we apply:

- From our extensive experience with scientific software engineering we understand how important it is to have maintainable software. We maintain a high level of quality in software development, delivering thoroughly reviewed, tested and documented code. If possible, we adopt principles of modularity and object-oriented design. This keeps the software manageable by your in-house software developers. We prefer working in phases with intermediate deliveries, and in close collaboration with your developers to ensure that the result is in agreement with your expectations.
- For a parallelization strategy, the computational work needs to be split into smaller pieces. In some cases this splitting is trivial, for example when multiple scenarios are independently evaluated. The only point requiring attention here is job scheduling and related infrastructure regarding running the software. For other cases, splitting of the work can be done in a variety of levels and approaches. VORtech has longstanding experience as well as up-to-date knowledge in parallelization. We've experienced the advantages and disadvantages of the various approaches.
- Sometimes it is possible to use a standard computational library for the most intensive part of the work. Such libraries contain heavily optimized and parallelized code suited for large-scale computations, and they are actively being developed and updated. This reduces the cost of software development. We have experience with a variety of scientific computational libraries, both proprietary and open source.



Operational models for simulation of water level and flow of the North Sea.

What VORtech can do for you

Based on our extensive experience in HPC, scientific software engineering and mathematical modeling, we provide a wide range of services.

- **The Model Scan** – Full code analysis and advice on the best strategy for improvement, including a development plan and cost-to-benefit ratio estimate.
- **Software development** – Implementation of the code improvements for any of the HPC optimization strategies. Deployment, porting and tuning of your software for a specific platform.
- **Hardware consultation and mediation** – Advice on HPC hardware requirements for your application, and mediation with our hardware, middleware and HPC cloud partners.
- **Training** – Courses in scientific software engineering and performance-aware programming.

Please contact us for more information on any of these services.

Tech Section: HPC performance boosting strategies

There are several approaches to optimize scientific software; all of them relate to the interplay between the software and the system it uses. Different approaches can be combined to achieve a cumulative speedup, allowing for a roadmap towards fully optimized software.

1. **Optimization of memory access patterns**
   In modern hardware architectures, memory latency and bandwidth are as important for the performance of software as is computational power itself. If the software is adapted to benefit from the cache memory in the processor, speedups of 2-10x can be achieved. This often involves changes to the layout of data structures and the order in which the data is processed.
2. **Improvement of single core performance**
   A variety of factors determines the performance of a single CPU core. These include pipelining, branch prediction and vectorization. By vectorizing the code, the same operation can be done on multiple numbers simultaneously. Often, the compiler will do this automatically. However, the compiler cannot always do this by itself, and some code rearrangement or hints from the programmer are required to achieve well vectorized code.
3. **Shared memory parallelization**
   Taking advantage of all cores in today's processors requires a multi-threaded approach. This is done by subdividing the problem into smaller tasks so that each core computes a part of the whole solution. This requires careful coordination as the memory space is shared, but usually the impact on the code is relatively modest.
4. **Distributed computing**
   When one needs more computational power than a single machine can provide, distributed computing is the way to go. This can be anything from several desktop PCs in an office network to a supercomputing cluster. The distributed parallel evaluation of the problem on all computers requires synchronization and communication. In general, this approach requires more substantial changes to the software than the previous strategies. It may, however, also speed up your software with a factor equal to the number of computers in the cluster.
5. **Offloading to accelerator boards**
   Computing on massively parallel architectures like graphics cards (GPGPU) and Many Integrated Cores (MIC) offers distinct advantages over the conventional CPU-based approach. Spectacular speedups for modest hardware investments are possible, but the code may need to undergo substantial rewriting. Furthermore, it strongly depends on the algorithms whether they are suitable for computation on these architectures. However, if the application allows it, the computations could be executed by a standard computational library, thus reducing the cost of rewriting the algorithms.

Interested?

For more information about the possibilities of high performance computing for your business, you can contact Koos Huijssen at:

VORtech
email: koos.huijssen@vortech.nl
phone: +31(0)15 - 285 01 25
web: www.vortech.nl/HPC