



Price: € 1,880.00 excl. VAT

Duration: 3 consecutive days

Contact: training@hightechinstitute.nl, +31 85 401 3600

Methods

Lectures, discussions, and exercises. On the last day, "Effective Multicore" is an interactive session where we revisit the topics from a different angle. Course material: USB stick with VM and exercises; book with course notes, handouts, exercises, solutions, and cheat sheets.

Intended for

C++ software engineers, designers, and architects who need to design bug-free fast-performing code that utilizes its underlying hardware well.

Prerequisites:

- Experience in software development
- Basic understanding of operating systems
- Reasonable understanding of modern C++

Objective

After successful completion of the course, the participant thoroughly understands how parallelism is implemented in modern CPUs. The participant can explain how modern C++ and supporting libraries help to keep complexity under control. The participant knows about

- Hardware architecture and modern CPU bottlenecks
- C++11+ parallelism and concurrency
- Reduce locking complexity and suffer less performance loss due to locking
- Berkeley's parallel pattern library
- OpenMP and other programming libraries
- Achieving speedups through parallelism

Intro

Well-performing multithreaded code is still a mystery to many. This 3-day course teaches participants how to benefit from the power of modern multicore processors by understanding the ins-and-outs of parallelism, the parallel programming paradigms, applying parallel patterns and avoiding common pitfalls.

Trainers

Klaas van Gend MSc

Certification

Participants will receive a High Tech Institute certificate for attending this training.

Trainers

Klaas van Gend MSc

Program

Day 1: Modern multicore computer architecture and concurrency, threading extensions in C++11 and newer, locking done wrong and right, RAI and libguarded.

Day 2: OpenMP, task and data parallelism, how to recognize concurrency opportunities, the Berkeley parallel pattern language, how to recognize and remedy parallelism blockers and data corrupters, parallelism anti-patterns.

Day 3: Using Vectorization, threadsafe API design, threading libraries, “Effective Multicore”.

Every day has lectures and at least two larger supporting exercises.