

Intel Parallel Studio

The Anselm cluster provides following elements of the Intel Parallel Studio XE

Intel Parallel Studio XE
Intel Compilers
Intel Debugger
Intel MKL Library
Intel Integrated Performance Primitives Library
Intel Threading Building Blocks Library

Intel compilers

The Intel compilers version 13.1.3 are available, via module intel. The compilers include the icc C and C++ compiler and the ifort fortran 77/90/95 compiler.

```
$ module load intel
$ icc -v
$ ifort -v
```

Read more at the Intel Compilers page.

Intel debugger

The intel debugger version 13.0 is available, via module intel. The debugger works for applications compiled with C and C++ compiler and the ifort fortran 77/90/95 compiler. The debugger provides java GUI environment. Use X display for running the GUI.

```
$ module load intel
$ idb
```

Read more at the Intel Debugger page.

Intel Math Kernel Library

Intel Math Kernel Library (Intel MKL) is a library of math kernel subroutines, extensively threaded and optimized for maximum performance. Intel MKL unites and provides these basic components: BLAS, LAPACK, ScaLapack, PARDISO, FFT, VML, VSL, Data fitting, Feast Eigensolver and many more.

```
$ module load mkl
```

Read more at the Intel MKL page.

Intel Integrated Performance Primitives

Intel Integrated Performance Primitives, version 7.1.1, compiled for AVX is available, via module `ipp`. The IPP is a library of highly optimized algorithmic building blocks for media and data applications. This includes signal, image and frame processing algorithms, such as FFT, FIR, Convolution, Optical Flow, Hough transform, Sum, MinMax and many more.

```
$ module load ipp
```

Read more at the Intel IPP page.

Intel Threading Building Blocks

Intel Threading Building Blocks (Intel TBB) is a library that supports scalable parallel programming using standard ISO C++ code. It does not require special languages or compilers. It is designed to promote scalable data parallel programming. Additionally, it fully supports nested parallelism, so you can build larger parallel components from smaller parallel components. To use the library, you specify tasks, not threads, and let the library map tasks onto threads in an efficient manner.

```
$ module load tbb
```

Read more at the Intel TBB page.