## Intel Compilers

This document gives a short overview of the Intel Compilers on the Linux-based HPC systems at LRZ. These are the mainline compilers that should be used to obtain the best possible performance.

### Versions

The Intel Fortran and C/C++ Compilers are available on all HPC platforms at LRZ. The packages also include a gdb-based debugger.

Presently available versions are listed in the following table.

<table>
<thead>
<tr>
<th>Product</th>
<th>Version</th>
<th>OpenMP standard support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fortran / C / C++ Compiler</td>
<td>16.0</td>
<td>extended (but not complete) 4.5 support</td>
</tr>
<tr>
<td>Fortran / C / C++ Compiler</td>
<td>17.0 (Default)</td>
<td>extended (but not complete) 4.5 support</td>
</tr>
<tr>
<td>Fortran / C / C++ Compiler</td>
<td>18.0</td>
<td>full 4.5 support, some features from 5.0 preview</td>
</tr>
<tr>
<td>Fortran / C / C++ Compiler</td>
<td>19.0</td>
<td>additional 5.0 features</td>
</tr>
</tbody>
</table>

The extent of compliance with the language standards depends on the version of the compiler.

- The extent of Fortran standards support is indicated in the linked document for Fortran on the Intel web site. Note that in particular, the parallel coarray programming model is supported; the linked document supplies extra information on the additional setup steps needed for general coarray use.
- The current default version of the Intel Fortran Compiler provides a full implementation of the Fortran 2003 standard, plus a significant set of Fortran 2008 and Fortran 2018 features as well as vendor-specific extensions.
- The extent of C++17 standards support is indicated in the linked document for C++17 on the Intel web site. That document also contains references to support for earlier C++ standards.
- The Intel C/C++ Compilers provide ISO/ANSI conforming implementations of the C and C++ languages, respectively.
- Furthermore, the OpenMP - shared memory and device parallelism - paradigm is supported in all languages.

### Usage

#### Environment Settings

The environment modules package provides default versions of the compilers automatically. Issuing

```bash
module list
```

should indicate “intel” among the loaded modules. Otherwise, issue `module load intel` to get access to the compiler suite.

Note that non-default releases can be selected by loading an explicit version after unloading the default one. Example:

```bash
module unload intel
module load intel/18.0
```

### Calling the compilers

The compiler call for Fortran 77/90/95 is `ifort`, where either a suitable switch for the language standard to be supported must be provided, or the file extension indicates this (e.g., .f for Fortran 77, .f90 for Fortran 90 free source).

The C compiler is called via `icc`, and the C++ compiler must be called via `icpc`.

### The GDB-based debugger

The debugger is a command-line tool that is based on gdb. Please invoke `gdb-ia` on a program to start up the debugger.

### Documentation

- Intel Compiler product page (has links to details as well as documentation via the support pages)
- Software Forums for the Intel Development products. Click the links appropriate for the language and platform you are working with.

### Support
For details on compiler optimization etc. please either consult the vendor documentation.
If you find any problems with the compilers, please first check out the FAQ and Troubleshooting.
If this does not provide a solution please contact Support for HPC and Big Data.