

# Mathematical Libraries and Application Software on JURECA and Booster JSC Training Course

June 2018 | I.Gutheil, D.Alvarez | JSC

# Outline

- Navigating modules
- Compiling for the Booster
- Sequential Libraries
- Parallel Libraries and Application Systems:
  - Threaded Libraries
  - MPI parallel Libraries
  - Application Software
- Software for Materials Science
- Software for Computational Engineering
- Further Information

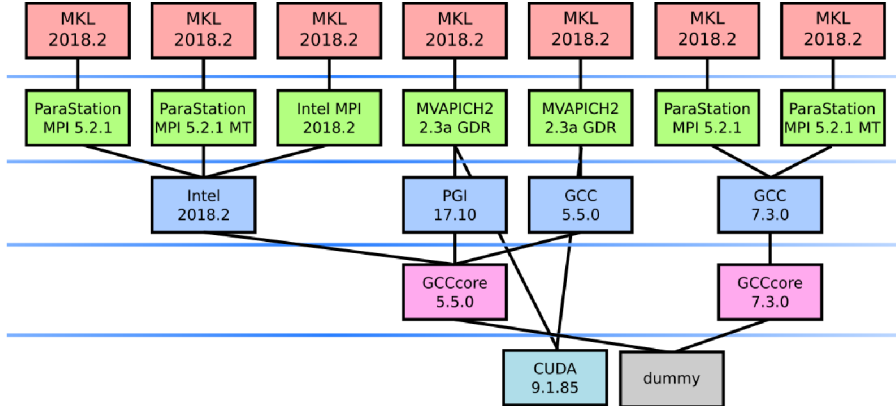


Figure: Current toolchain tree in JURECA

# Modules environment

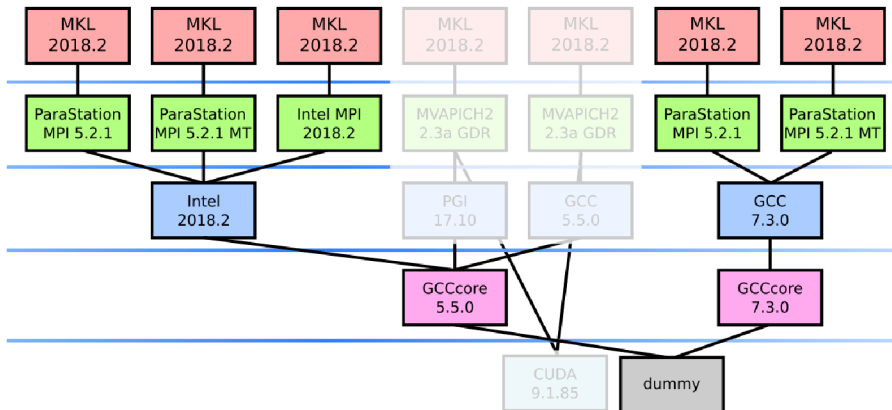


Figure: Current toolchain tree in Booster

# Modules environment

## Hierarchical modules

- GCCcore/.5.5.0 is preloaded, which enables a lot of base software
- For HPC software you have to load a compiler, to expand the module tree  
ml Intel
- Then you load an MPI version  
ml ParaStationMPI
- Then you can load any other packages  
ml QuantumESPRESSO/6.2

# Modules environment

## JURECA and Booster (I)

- `ml spider name` shows whether a library is available in the current stage and in which versions
- `ml spider name/version` shows which environment you have to load before you can load that version
- Many packages are hidden. To see them use `ml spider --show-hidden name`

# Modules environment

- For R, Python and Perl we use bundles
  - You might be looking for a software package that is part of a bigger module
- Use `ml key software`  
`ml key numpy` will suggest SciPy-Stack
  - You might be looking for a software package that is part of a bigger module
- You can use then `ml spider` to find out how to load the module

# Modules environment

## Stages

- The whole software stack in JURECA and the Booster is updated every 6 months
  - Right when there is an allocation for new projects
- Old stages are still accessible
- To check availability in other stages first type  
`ml use /usr/local/software/jureca[booster]/OtherStages`



# Compiling for the Booster (I)

- Cross-compilation on login nodes not recommended but technically possible
- `ml Architecture/KNL`
  - This should be the first module you load
  - Then you have the booster software stack available.
  - This doesn't work on the cluster compute nodes!
- Remember, cross-compiling is not recommended unless you know what you are doing

# Compiling for the Booster (II)

## Recommended way to compile for the Booster

- Start an interactive session, by  
`salloc --partition=develbooster`
- After the allocation is successful start a remote shell from within the `salloc` session and connect it to a pseudo terminal using  
`srtn --cpu_bind=none --nodes=1 --pty /bin/bash -i`
- In batch scripts add  
`module load Architecture/KNL`  
before all other module commands

# Sequential Libraries and Packages (I)

## Vendor specific Libraries

- MKL Intel® Math Kernel Library  
versions as mentioned in general informations,  
2018.2.199 on JURECA and Booster

# Sequential Libraries and Packages (II)

## Public domain Libraries

- LAPACK (Linear Algebra PACKage)
- ARPACK (Arnoldi PACKage)
- GSL (Gnu Scientific Library)
- GMP (Gnu Multiple Precision Arithmetic Library)

## Commercial library

NAG Fortran Library: JURECA only

# Contents of Intel® MKL

- BLAS, Sparse BLAS, CBLAS
- LAPACK
- Iterative Sparse Solvers, Trust Region Solver
- Vector Math Library
- Vector Statistical Library
- Fourier Transform Functions
- Trigonometric Transform Functions

# Contents of Intel® MKL

- GMP routines
- Poisson Library
- Interface for fftw

For more information see <http://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/SystemDependentLibraries/SystemDependentLibraries/MKL.html?nn=1035570>

# Usage of MKL (I)

- Can be loaded with Intel compiler or GCC
- MPI has to be loaded before imkl
- FORTRAN, C, and C++ callable
- Arrays FORTRAN like, i.e. column-first (except cblas)
- Compilation and linking of program name.f calling sequential MKL routines:

```
ifort name.f -o name -lmkl_intel_lp64 -lmkl_sequential  
-lmkl_core [-liomp5 -lpthread]
```

# Usage of MKL (II)

To use CBLAS include mkl.h into source code

Compilation and linking of program name.c calling sequential MKL

```
icc name.c -o name -lmkl_intel_lp64 -lmkl_sequential  
-lmkl_core [-liomp5 -lpthread -lifcore -lifport]
```



# LAPACK

- Part of MKL in libmkl\_core.a
- Can be loaded with Intel or GCC on JURECA and Booster
- In older Stages also part of OpenBLAS with GCC

## Arpack

- ARPACK-NG/3.5.0
- Iterative solver for sparse eigenvalue problems
- Reverse communication interface
- FORTRAN 77
- Calls LAPACK and BLAS routines, MKL necessary

# GSL – GNU Scientific Library

- `module load intel-para GSL/2.4`  
for icc version on JURECA
- `module load GCC/7.3.0 ParaStationMPI[/5.2.1-1] GSL/2.4`  
for gcc version on JURECA
- Provides a wide range of mathematical routines
- Not recommended for performance reasons
- Often used by configure scripts

# GMP- GNU Multiple Precision Library

version 6.1.2 on JURECA

## NAG Libraries

- NAG Fortran Mark 26 on JURECA only available with Intel compiler
- Please tell us if you really need it

# Parallel Libraries

## Threaded Parallelism I

- MKL (JURECA)
  - is multi-threaded
  - if OMP\_NUM\_THREADS not set, 48 threads used on JURECA
  - 68 threads on Booster
  - Usage:

```
ifort name.f -o name -lmkl_intel_lp64  
-lmkl_intel_thread -lmkl_core -liomp5 -lpthread
```

# Parallel Libraries

## Threaded Parallelism II

- FFTW 3.3 (Fastest Fourier Transform of the West)  
3.3.7 on JURECA Intel compiler and GCC  
MPI, OpenMP and threads version  
<http://www.fftw.org>

# Parallel Libraries

## MPI Parallelism

- ScaLAPACK (Scalable Linear Algebra PACKage)
- ELPA (Eigenvalue SoLvers for Petaflop-Applications)
- Elemental, C++ framework for parallel dense linear algebra
- FFTW (Fastest Fourier Transform of the West)
- MUMPS (MULTifrontal Massively Parallel sparse direct Solver)  
not yet on Booster
- ParMETIS (Parallel Graph Partitioning)
- Hypre (high performance preconditioners)

# MPI Parallelism (II)

- PARPACK (Parallel ARPACK), Eigensolver
- SPRNG (Scalable Parallel Random Number Generator)
- SUNDIALS (SUite of Nonlinear and Differential/ALgebraic equation Solvers)
- All three not yet on Booster

## Parallel Systems, MPI Parallelism

- PETSc, toolkit for partial differential equations

# ScaLAPACK

- part of MKL,
- Parallel BLAS 1-3, PBLAS Version 2
- Dense linear system solvers
- Banded linear system solvers
- Solvers for Linear Least Squares Problem
- Singular value decomposition
- Eigenvalues and eigenvectors of dense symmetric/hermitian matrices
- <http://www.netlib.org/scalapack/index.html>



# Usage on JURECA

Linking a program name.f calling routines from ScaLAPACK, default version, Intel compiler:

```
mpif77 name.f -lmkl_scalapack_lp64  
-lmkl_blacs_intelmpi_lp64 -lmkl_intel_lp64  
-lmkl_intel_thread[-lmkl_sequential]  
-lmkl_core -liomp5 -lpthread
```

# ELPA

## Eigenvalue SoLvers for Petaflop-Applications

ELPA uses ScaLAPACK, must be linked together with scalapack

- FORTRAN 95, same data-distribution as ScaLAPACK
- [http://elpa.rzg.mpg.de/elpa-english?set\\_language=en](http://elpa.rzg.mpg.de/elpa-english?set_language=en)
- JURECA pure MPI and hybrid version 2017.11.001
- Booster special version with AVX512 kernels for ELPA2

# Elemental

- C++ framework, two-dimensional data distribution element by element
- <http://libelemental.org/about/>
- 0.87.7

# MUMPS

## Multifrontal Massively Parallel sparse direct Solver

- Solution of linear systems with symmetric positive definite matrices, general symmetric matrices, general unsymmetric matrices
- Real or Complex
- Parallel factorization and solve phase, iterative refinement and backward error analysis
- F90 with MPI and OpenMP since 5.1.1
- current version 5.1.2
- <http://graal.ens-lyon.fr/MUMPS/>

# ParMETIS

Parallel Graph Partitioning and Fill-reducing Matrix Ordering  
developed in Karypis Lab at the University of Minnesota  
4.0.3 on JURECA

<http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview>

## Hypre

High performance preconditioners  
Version 2.14.0 on JURECA, also version with bigint,  
<http://www.llnl.gov/CASC/hypre/software.html>

# FFTW

3.3.7 on JURECA, (Intel and GCC modules)  
all with MPI parallel version

## PARPACK

- PARPACK MPI-Version, part of ARPACK-NG/3.5.0
- Must be linked with LAPACK and BLAS
- Reverse communication interface, user has to supply parallel matrix-vector multiplication

<https://github.com/opencollab/arpack-ng>

[http://www.caam.rice.edu/~kristyn/parpack\\_home.html](http://www.caam.rice.edu/~kristyn/parpack_home.html)

# SPRNG

The Scalable Parallel Random Number

not yet on Booster

Generators Library for ASCI Monte Carlo Computations

version 5.0:

various random number generators in one library

Version 1.0 separate library for each random number generator

<http://sprng.cs.fsu.edu/>

## Sundials (CVODE)

Package for the solution of ordinary differential equations, Version

3.1.0

not yet on Booster

<https://computation.llnl.gov/casc/sundials/main.html>

# PETSc

- Portable, Extensible Toolkit for Scientific Computation
- Numerical solution of partial differential equations
- version 3.9.0
- with several other packages included
- complex version and version with 8-Byte integer
- debug versions in Devel Stages only
- <http://www.mcs.anl.gov/petsc/>
- `ml spider petsc`



# Software for Materials Science

| Package         | JURECA |
|-----------------|--------|
| Abinit          | yes    |
| ADF             | yes    |
| Amber           | yes    |
| CP2K            | yes    |
| CPMD            | yes    |
| Gromacs         | yes    |
| LAMMPS          | yes    |
| Molpro          | yes    |
| NAMD            | yes    |
| NWChem          | yes    |
| QuantumEspresso | yes    |
| TURBOMOLE       | yes    |

# Software for Computational Engineering

- JURECA only
- CFD Package **OpenFOAM**
  - Version 4.1 in Stages/2017b and some older versions in older stages
  - and OpenFOAM-Extend 3.1 and 3.2, only in older stages
- Commercial **FEM Software**
  - **ANSYS, LS-DYNA , COMSOL** are technically maintained on **JURECA**
  - **Licenses** must be provided by the **User** !

# Further information and JSC-people

<http://www.fz-juelich.de/ias/jsc/jureca>

[http://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/\\_node.html](http://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Software/_node.html)

## **Mailto**

Supercomputer support:

[sc@fz-juelich.de](mailto:sc@fz-juelich.de)

I. Gutheil: Parallel mathematical Libraries

[i.gutheil@fz-juelich.de](mailto:i.gutheil@fz-juelich.de)

D. Alvarez: Software Manager

[d.alvarez@fz-juelich.de](mailto:d.alvarez@fz-juelich.de)

B. Körfgen: Physics and Engineering software

[b.koerfgen@fz-juelich.de](mailto:b.koerfgen@fz-juelich.de)