FrontISTR Migration: RHEL to Ubuntu 24.04 LTS

Complete Installation & Migration Guide

FrontISTR (開発:東京大学) is an open-source large-scale parallel finite element method (FEM) structural analysis software. This guide covers migrating from RHEL-based systems to Ubuntu 24.04 LTS.

Overview

FrontISTR Details:

- **Version (Current):** 5.6 / 5.4
- **License:** MIT License
- **Repository:** https://gitlab.com/FrontISTR-Commons/FrontISTR (active development)
- **Official Documentation:** https://manual.frontistr.com/en/
- **Download:** https://github.com/FrontISTR/FrontISTR (mirror) or https://www.frontistr.com

Key Features for HPC:

- Large-scale parallel processing (MPI + OpenMP)
- Supports H100 GPU acceleration
- Compatible with SLURM job scheduler
- ParaView visualization integration

Phase 1: Pre-Migration Planning

1.1 Check Current FrontISTR Configuration

On your RHEL system:

```
# Verify FrontISTR installation location and version
which fistr1
fistr1 --version

# Export current environment modules
module list > rhel_modules.txt

# Document compilation configuration
cat $FRONTISTR_HOME/Makefile.conf > rhel_frontistr_config.conf

# Check linked libraries
ldd $(which fistr1) > rhel_library_dependencies.txt

# Verify MPI configuration
mpirun --version
which mpicc mpif90
```

1.2 Identify Dependencies on RHEL

```
# List all FrontISTR-related packages
rpm -qa | grep -i "frontistr\|metis\|mumps\|mpi\|openblas"

# Check Fortran compiler and GCC versions
gfortran --version
gcc --version
```

Phase 2: Ubuntu 24.04 LTS Installation

2.1 Base System Setup

```
# Update system
sudo apt update
sudo apt upgrade -y

# Install essential build tools
sudo apt install -y build-essential gfortran git cmake

# Install required libraries
sudo apt install -y \
  libopenmpi-dev openmpi-bin \
  libmumps-dev libscalapack-mpi-dev \
  libmetis-dev libparmetis-dev \
```

```
  libblas-dev liblapack-dev libopenblas-dev \
  ruby doxygen swig

# Optional but recommended for monitoring
sudo apt install -y \
  prometheus grafana-server
```

2.2 Create Build Environment

```
# Create workspace for dependencies
mkdir -p $HOME/work
mkdir -p $HOME/local
cd $HOME/work

# Set environment variables for consistent paths
cat >> $HOME/.bashrc << 'EOF'
export WORK_DIR=$HOME/work
export LOCAL_DIR=$HOME/local
export PATH=$LOCAL_DIR/bin:$PATH
export LD_LIBRARY_PATH=$LOCAL_DIR/lib:$LD_LIBRARY_PATH
export PKG_CONFIG_PATH=$LOCAL_DIR/lib/pkgconfig:$PKG_CONFIG_PATH
EOF

source $HOME/.bashrc
```

Phase 3: Install Required External Libraries

3.1 OpenBLAS (Recommended over reference LAPACK/BLAS)

```
cd $WORK_DIR
wget https://github.com/OpenMathLib/OpenBLAS/archive/v0.3.26.tar.gz
tar xzf v0.3.26.tar.gz
cd OpenBLAS-0.3.26

# Build with OpenMP support for H100 compatibility
make -j 8 USE_OPENMP=1
make PREFIX=$LOCAL_DIR install

# Verify installation
ls $LOCAL_DIR/lib/libopenblas*
```

3.2 METIS (Required for domain decomposition)

Important: FrontISTR 5.x supports both METIS 4.0.3 and 5.x

```
cd $WORK_DIR
# For Ubuntu 24.04, use version 5.x from official repository
git clone https://gitlab.com/FrontISTR-Commons/METIS.git
cd METIS

mkdir build && cd build
cmake -DCMAKE_INSTALL_PREFIX=$LOCAL_DIR \
       -DCMAKE_BUILD_TYPE=Release ..
make -j 8
make install

# Verify
ls $LOCAL_DIR/lib/libmetis*
```

3.3 MUMPS (Optional: for direct solver method)

```
cd $WORK_DIR
wget http://mumps.enseeiht.fr/public/MUMPS_5.7.3.tar.gz
tar xzf MUMPS_5.7.3.tar.gz
cd MUMPS_5.7.3

# Copy makefile template for gfortran + OpenMPI
cp Make.inc/Makefile.gfortran.SEQ Makefile.inc

# Edit Makefile.inc for your configuration
cat >> Makefile.inc << 'EOF'
# Add these lines for Ubuntu 24.04
PREFIX = $(LOCAL_DIR)
LIBSEQ = libseq/libseqneeded.a
OUTDIR = $(PREFIX)/lib
INCDIRSEQ = -Ilibseq
LAPACK = -L$(LOCAL_DIR)/lib -lopenblas
BLAS = -L$(LOCAL_DIR)/lib -lopenblas
OPTF = -O2 -ffree-line-length-none
OPTL = -O2
OPTC = -O2
EOF

make -j 8 all
make PREFIX=$LOCAL_DIR install
```

```
# Verify
ls $LOCAL_DIR/lib/libmumps*
```

3.4 Verify MPI Environment

```
# Check MPI compiler wrappers
mpicc --version
mpif90 --version
mpirun --version

# Test simple MPI compilation
cat > test_mpi.f90 << 'EOF'
program test_mpi
  implicit none
  print *, "MPI environment OK"
end program test_mpi
EOF

mpif90 -o test_mpi test_mpi.f90
./test_mpi
rm test_mpi*
```

Phase 4: FrontISTR Installation

4.1 Download FrontISTR

```
cd $WORK_DIR

# Clone from official repository
git clone https://gitlab.com/FrontISTR-Commons/FrontISTR.git
cd FrontISTR

# Check available versions/tags
git tag | tail -20
git checkout v5.6  # or latest version
```

4.2 Build with CMake (Recommended)

Most straightforward for Ubuntu 24.04:

```
cd $WORK_DIR/FrontISTR
```

```
mkdir build && cd build

cmake -DCMAKE_INSTALL_PREFIX=$HOME/FrontISTR \
       -DCMAKE_BUILD_TYPE=Release \
       -DWITH_ML=ON \
       -DWITH_METIS=ON \
       -DWITH_MUMPS=ON \
       -DBLAS_LIBRARIES="$LOCAL_DIR/lib/libopenblas.a" \
       -DLAPACK_LIBRARIES="$LOCAL_DIR/lib/libopenblas.a" \
       -DMETIS_INCLUDE_DIRS="$LOCAL_DIR/include" \
       -DMETIS_LIBRARIES="$LOCAL_DIR/lib/libmetis.a" \
       ..

# Build with parallel jobs
make -j 8

# Verify build (optional but recommended)
make test VERBOSE=1

# Install
make install
```

4.3 Alternative: Build with Makefile.conf (Manual)

If CMake encounters issues:

```
cd $WORK_DIR/FrontISTR

# Copy template
cp Makefile.conf.org Makefile.conf

# Edit Makefile.conf (example for Ubuntu 24.04)
cat > Makefile.conf << 'EOF'
# FrontISTR Configuration for Ubuntu 24.04 LTS

# MPI Configuration
MPIDIR = /usr
MPIBINDIR = $(MPIDIR)/bin
MPILIBDIR = $(MPIDIR)/lib/x86_64-linux-gnu
MPIINCDIR = $(MPIDIR)/include
MPILIBS = -lmpi -lmpi_cxx -lmpi_f90

# Installation
PREFIX = $(HOME)/FrontISTR
BINDIR = $(PREFIX)/bin
LIBDIR = $(PREFIX)/lib
```

```
INCLUDEDIR = $(PREFIX)/include

# External Libraries
METISDIR = $(LOCAL_DIR)
METISLIBDIR = $(METISDIR)/lib
METISINCDIR = $(METISDIR)/include

MUMPSDIR = $(LOCAL_DIR)
MUMPSINCDIR = $(MUMPSDIR)/include
MUMPSLIBDIR = $(MUMPSDIR)/lib

# BLAS/LAPACK
BLASDIR = $(LOCAL_DIR)
BLAISLIBDIR = $(BLASDIR)/lib
BLASINCDIR = $(BLASDIR)/include

# Compiler Settings
CC = mpicc
CFLAGS = -O2 -march=native
LDFLAGS = -lm

CPP = mpic++
CPPFLAGS = -O2 -march=native

F90 = mpif90
F90FLAGS = -O2 -march=native -ffree-line-length-none

# Tools
MAKE = make
AR = ar ruv
CP = cp -f
RM = rm -f
MKDIR = mkdir -p
EOF

# Run setup script
./setup.sh -g -p --with-tools --with-metis --with-mumps --with-ml

# Build
make -j 8
make install
```

4.4 Verify Installation

```
# Add to PATH
export PATH=$HOME/FrontISTR/bin:$PATH
```

```
echo "export PATH=$HOME/FrontISTR/bin:\$PATH" >> $HOME/.bashrc

# Test single-threaded execution
cd $WORK_DIR/FrontISTR/tutorial/01_elastic_hinge
fistr1

# Expected output should include:
# "FrontISTR Completed !!"
```

Phase 5: Parallel (MPI) Configuration

5.1 Test Parallel Execution

```
cd $WORK_DIR/FrontISTR/tutorial/02_elastic_hinge_parallel

# Run domain partitioner
hecmw_part1
# Partitions mesh into 4 domains by default

# Execute with 4 MPI processes
mpirun -np 4 fistr1

# Expected output:
# "FrontISTR Completed !!"
```

5.2 SLURM Integration (for your HPC cluster)

Create submission script:

```
cat > $HOME/frontistr_slurm_submit.sh << 'EOF'
#!/bin/bash
#SBATCH --job-name=frontistr_test
#SBATCH --nodes=2
#SBATCH --ntasks-per-node=4
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=8G
#SBATCH --partition=gpu
#SBATCH --gres=gpu:h100:1
#SBATCH --time=01:00:00
#SBATCH --output=frontistr_%j.log

# Load modules
module load openmpi/4.1.5
```

```
module load gcc/12.2.0

# Set environment
export PATH=$HOME/FrontISTR/bin:$PATH
export LD_LIBRARY_PATH=$HOME/local/lib:$LD_LIBRARY_PATH

# Run FrontISTR analysis
cd $SLURM_SUBMIT_DIR
srun -N 2 --ntasks=8 fistr1 < input.cnt > output.log 2>&1
EOF

chmod +x $HOME/frontistr_slurm_submit.sh

# Submit to SLURM
sbatch $HOME/frontistr_slurm_submit.sh
```

Phase 6: Data Migration

6.1 Transfer Analysis Files

```
# On RHEL system, create archive of all analysis cases
cd /path/to/frontistr/cases
tar -czf frontistr_cases_backup.tar.gz *.cnt *.msh *.txt *.dat

# Transfer to Ubuntu system
scp frontistr_cases_backup.tar.gz user@ubuntu-host:/home/user/work/

# On Ubuntu, extract and verify
cd $HOME/work
tar -xzf frontistr_cases_backup.tar.gz
```

6.2 Compatibility Checks

```
# Ensure file formats are compatible
file *.msh *.cnt *.dat  # Should show text or ASCII

# For binary mesh files, verify architecture compatibility
# RHEL (likely 64-bit x86_64) → Ubuntu 24.04 (64-bit x86_64)
# Generally compatible, but test first
```

Phase 7: Troubleshooting

Common Issues and Solutions

Issue 1: MPI Compiler Wrapper Not Found

```
# Check if OpenMPI is installed correctly
which mpicc mpirun

# If not found, install/reinstall
sudo apt reinstall libopenmpi-dev openmpi-bin

# Verify PATH
echo $PATH | grep -i openmpi
```

Issue 2: METIS Library Version Mismatch

```
# Check installed METIS version
strings $LOCAL_DIR/lib/libmetis.a | grep "METIS"

# If version 4.x is needed (legacy):
# Download from: http://glaros.dtc.umn.edu/gkhome/fsroot/sw/metis/OLD/metis-4.0.3
# But FrontISTR 5.x is recommended with METIS 5.x
```

Issue 3: Fortran Compilation Errors

```
# Update GCC/gfortran if issues occur
sudo apt install -y gcc-12 gfortran-12
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-12 60
sudo update-alternatives --install /usr/bin/gfortran gfortran /usr/bin/gfortran-1

# Check version
gfortran --version
```

Issue 4: Out-of-Memory During Large Parallel Jobs

```
# Check available memory
free -h

# Adjust SLURM allocation
# Increase --mem-per-cpu in submission script
```

```
# Or use Intel MKL if available (faster, lower memory)
```

Phase 8: Performance Optimization

8.1 Compiler Optimization Flags

For H100 GPUs and modern CPU architectures, update Makefile.conf:

```
OPTFLAGS = -Ofast -march=native -mtune=native -mavx2 -mfma -flto

# For AMD EPYC (if applicable):
# OPTFLAGS = -Ofast -march=znver3 -mtune=znver3 -mavx2 -mfma

# For Intel Xeon (Sapphire Rapids, etc.):
# OPTFLAGS = -Ofast -march=sapphirerapids -mtune=sapphirerapids
```

8.2 OpenMP Configuration (for Shared Memory)

```
# In SLURM script, enable OpenMP for multi-core nodes
export OMP_NUM_THREADS=4
srun -N 2 --ntasks-per-node=2 --cpus-per-task=4 fistr1
```

8.3 Intel MKL (Optional, for performance)

If Intel MKL is available:

```
# Load MKL module (if installed)
module load intel-mkl

# Update Makefile.conf to use MKL instead of OpenBLAS:
# LAPACK = -L${MKLROOT}/lib/intel64 -lmkl_intel_lp64 -lmkl_core -lmkl_sequential
# BLAS = -L${MKLROOT}/lib/intel64 -lmkl_intel_lp64 -lmkl_core -lmkl_sequential
```

Phase 9: Monitoring and Validation

9.1 Prometheus/Grafana Integration

For your existing monitoring infrastructure:

```
# Configure FrontISTR job metrics in Prometheus
cat >> /etc/prometheus/prometheus.yml << 'EOF'
  - job_name: 'frontistr_jobs'
    static_configs:
      - targets: ['localhost:9100']  # Node exporter from SLURM nodes
EOF

# Restart Prometheus
sudo systemctl restart prometheus
```

9.2 Performance Benchmarking

```
# Create benchmark script
cat > benchmark_frontistr.sh << 'EOF'
#!/bin/bash

RESULTS_FILE="benchmark_results.txt"
echo "FrontISTR Performance Benchmark - Ubuntu 24.04" > $RESULTS_FILE
echo "================================================" >> $RESULTS_FILE

for nodes in 1 2 4 8; do
  echo "Testing with $nodes nodes..." >> $RESULTS_FILE
  time mpirun -np $((nodes * 4)) \
    $HOME/FrontISTR/bin/fistr1 >> $RESULTS_FILE 2>&1
done

echo "Benchmark complete. Results saved to $RESULTS_FILE"
EOF

chmod +x benchmark_frontistr.sh
./benchmark_frontistr.sh
```

Phase 10: Documentation & Environment Modules

10.1 Create Environment Module for FrontISTR

```
mkdir -p $HOME/.modulefiles

cat > $HOME/.modulefiles/frontistr/5.6 << 'EOF'
#%Module1.0
```

```
proc ModulesHelp {} {
    puts stderr "FrontISTR v5.6 - Large-scale parallel FEM solver"
}

module-whatis "FrontISTR v5.6"

# Set paths
setenv FRONTISTR_HOME $::env(HOME)/FrontISTR
setenv PATH $::env(FRONTISTR_HOME)/bin:$::env(PATH)
setenv LD_LIBRARY_PATH $::env(HOME)/local/lib:$::env(LD_LIBRARY_PATH)

# Dependencies
prereq openmpi/4.1.5
prereq gcc/12.2.0

conflict frontistr
EOF

# Use module
module use $HOME/.modulefiles
module load frontistr/5.6
```

10.2 Create Installation Checklist

```
cat > MIGRATION_CHECKLIST.md << 'EOF'
# FrontISTR Migration Checklist

- [ ] Ubuntu 24.04 LTS installed
- [ ] Build tools installed (build-essential, gfortran, cmake)
- [ ] MPI environment configured (OpenMPI tested)
- [ ] OpenBLAS compiled and installed
- [ ] METIS compiled and installed
- [ ] MUMPS compiled and installed (optional)
- [ ] FrontISTR compiled successfully
- [ ] Single-threaded test case passed
- [ ] Parallel (MPI) test case passed
- [ ] Analysis data transferred from RHEL
- [ ] SLURM job submission tested
- [ ] Performance benchmarks completed
- [ ] Monitoring (Prometheus/Grafana) configured
- [ ] Environment modules created
- [ ] Documentation archived

## Key Files Migrated:
- /path/to/cases/*.cnt (control files)
- /path/to/cases/*.msh (mesh files)
```

```
- /path/to/cases/*.dat (input data)

## Performance Comparison (RHEL vs Ubuntu 24.04):
(Document here after testing)
EOF
```

Summary Table: Key Differences

Aspect	RHEL	Ubuntu 24.04
Package format	RPM (.rpm)	DEB (.deb)
Package manager	yum / dnf	apt / apt-get
Default MPI	Optional	libopenmpi-dev standard
Compiler	GCC (similar version)	GCC-12 (modern)
Build tools	Similar	More recent cmake (3.28+)
Library paths	`/usr/lib64`	`/usr/lib/x86_64-linux-gnu`
Default compiler flags	-O2	-O2 (same)
Security module	SELinux	AppArmor (minimal impact)
Init system	systemd	systemd (same)

References & Resources

- **Official FrontISTR Documentation:** https://manual.frontistr.com/en/
- **FrontISTR GitHub (Mirror):** https://github.com/FrontISTR/FrontISTR
- **FrontISTR GitLab (Active):** https://gitlab.com/FrontISTR-Commons/FrontISTR
- **University of Tokyo FrontISTR:** http://www.multi.k.u-tokyo.ac.jp/FrontISTR/
- **Ubuntu Documentation:** https://ubuntu.com/
- **OpenMPI Docs:** https://www.open-mpi.org/

Contact & Support

For FrontISTR-specific issues:

- Email: contact@frontistr.com
- GitHub Issues: https://github.com/FrontISTR/FrontISTR/issues
- Community Forum: https://www.frontistr.com/ (Japanese/English)

For Ubuntu HPC integration:

- Canonical Support: https://ubuntu.com/support
- Ubuntu Server Documentation: https://ubuntu.com/server/docs

---

Last Updated: February 2026 **Tested on:** Ubuntu 24.04.4 LTS, FrontISTR 5.6