

## Tools/itac/example\_rank\_league

### Example: Intel Trace Analyzer on benchmark rank\_league

- Prepare environment

```
module purge
```

```
# Load Intel compiler environment
```

```
source /software/all/toolkit/Intel_OneAPI/compiler/latest/env/vars.sh
```

```
# Load Intel MPI environment
```

```
source /software/all/toolkit/Intel_OneAPI/mpi/latest/env/vars.sh
```

```
# Load Intel Trace Analyzer and Collector environment
```

```
source /software/all/toolkit/Intel_OneAPI/itac/latest/env/vars.sh
```

- Build rank\_league benchmark with instrumentation

```
export MPICC="mpiicc"
```

```
export CFLAGS="-Ofast -xHost"
```

```
# -tcollect
```

```
#           inserts instrumentation probes calling the Intel(R) Trace Collector  
#           API.
```

```
# -trace
```

```
#           Use the -t or -trace option to link the resulting executable file  
#           against the Intel(R) Trace Collector library.
```

```
# -g
```

```
#           Produce symbolic debug information.
```

```
CFLAGS+=" -tcollect -trace -g"
```

```
${MPICC} ${CFLAGS} rank_league.blocked.c -o rank_league
```

- Create job script file rank\_league.jobscript

```
#!/usr/bin/bash
#SBATCH --partition=<...>
#SBATCH --nodes=4
#SBATCH --tasks-per-node=1
#SBATCH --time=20

# Prepare environment
module purge

# Load Intel compiler environment
source /software/all/toolkit/Intel_OneAPI/compiler/latest/env/vars.sh

# Load Intel MPI environment
source /software/all/toolkit/Intel_OneAPI/mpi/latest/env/vars.sh

# Load Intel Trace Analyzer and Collector environment
source /software/all/toolkit/Intel_OneAPI/itac/latest/env/vars.sh

# Configure trace collector
export VT_LOGFILE_FORMAT=SINGLESTF      # Create only one trace file
export VT_LOGFILE_NAME=rank_league.stf # Name of trace file

# Configure mpirun
export I_MPI_HYDRA_BOOTSTRAP="slurm"
export I_MPI_HYDRA_BRANCH_COUNT="-1"

# Execute rank_league benchmark
# test_type:  b - bandwidth (default)
# output_type: s - statistics per rank - average, min, max
# loop_num:   number of loops per every round (default is 200)
mpirun \
    -print-rank-map -binding domain=core \
    ./rank_league -t=b -o=s -l=200
```

- Execute instrumented rank\_league benchmark and create trace file rank\_league.stf

```
sbatch < rank_league.jobscript

(hkn0805:0)
(hkn0807:1)
(hkn0809:2)
(hkn0811:3)

***** Running bandwidth test *****
Total number of rounds:      3
Total number of loops per round: 200
Message size:                100000
```

\*\*\*\*\*

Round number 3

\*\*\*\*\*

RANK	MIN RESULT	RANK	MAX RESULT	RANK	AVERAGE
0	11372.02	1	15648.48	3	12829.07
1	11713.28	2	15348.30	3	13923.02
2	11580.73	1	15384.44	3	13892.53
3	11436.90	2	15006.74	0	12655.29

Global statistics:

MIN 11372.02 between 1 and 0

MAX 15648.48 between 0 and 3

AVERAGE 13324.98

[0] Intel(R) Trace Collector INFO: Writing tracefile rank\_league.stf in <...>/rank\_league

- Analyze trace file

traceanalyzer rank\_league.stf