Job-Arrays

In PBSPro gibt es die Möglichkeit sogenannte Job-Arrays zu starten. Diese ermöglichen es dem Cluster gleichartige Jobs effizient und schnell zu verarbeiten.

Dabei wird nur ein Job abgeschickt, welcher jedoch mehrfach mit einem fortlaufenden Index gestartet wird. Dadurch ist es möglich bei jedem Lauf andere Eingabe-Parameter zu verwenden.

Jeder Job bekommt dazu die Variable $PBS_ARRAY_INDEX gesetzt und kann dadurch unterschiedliche Programme starten.

Beim Abschicken der Jobs muss man dafür jedoch den Range für $PBS_ARRAY_INDEX mit:

```
qsub -A Project -J 0-3 job.file
```

Beispiel 1: Parameter im Job-File

```
#!/bin/bash
#PBS -l select=1:ncpus=1:mem=2gb:arch=ivybridge
#PBS -l walltime=72:00:00
#PBS -N Bachelorarbeit
#PBS -A Benchmarking
#PBS -r y

me=`basename $0`
LOGFILE=$PBS_O_WORKDIR/$PBS_JOBNAME"."$PBS_JOBID"_"$PBS_ARRAY_INDEX".log"

#SCRATCHDIR=/gpfs/scratch/$USER/$PBS_JOBID/
#mkdir -p "$SCRATCHDIR"

cd $PBS_O_WORKDIR

echo "$PBS_JOBID ($PBS_JOBNAME) @ `hostname` at `date` in "$RUNDIR" START" > $LOGFILE
echo "`date +"%d.%m.%Y-%T"`" >> $LOGFILE

echo >> $LOGFILE
echo "GLOBAL PARAMETERS" >> $LOGFILE
echo "--------------------------" >> $LOGFILE
echo "Node      : "`hostname` >> $LOGFILE
echo "RunDir    : "$PBS_O_WORKDIR >> $LOGFILE
#echo "ScratchDir: "$SCRATCHDIR >> $LOGFILE
echo "# CPUs    : "$NCPUS >> $LOGFILE
echo "# Threads : "$OMP_NUM_THREADS >> $LOGFILE

shopt -s extglob
job[1]='java -jar $PBS_O_WORKDIR/Code/elki/out/artifacts/elki_jar/elki.jar KDDCLIApplication  -evaluator
NoAutomaticEvaluation -resulthandler ResultWriter -out.silentoverwrite true -dbc.in $PBS_O_WORKDIR/Datasets
/half_rel_160d/half_rel_160d.csv -parser.labelIndices 161 -algorithm clustering.kmeans.KMeansLloyd -kmeans.
initialization SampleKMeansInitialization -kmeans.algorithm KMeansLloyd -kmeans.samplesize 0.1 -kmeans.k 113  -
dbc.filter transform.GlobalPrincipalComponentAnalysisTransform -out $PBS_O_WORKDIR/Datasets/half_rel_160d/output
/pca/113'
job[2]='java -jar $PBS_O_WORKDIR/Code/elki/out/artifacts/elki_jar/elki.jar KDDCLIApplication  -evaluator
NoAutomaticEvaluation -resulthandler ResultWriter -out.silentoverwrite true -dbc.in $PBS_O_WORKDIR/Datasets
/half_rel_160d/half_rel_160d.csv -parser.labelIndices 161 -algorithm clustering.kmeans.KMeansLloyd -kmeans.
initialization SampleKMeansInitialization -kmeans.algorithm KMeansLloyd -kmeans.samplesize 0.1 -kmeans.k 114  -
dbc.filter transform.GlobalPrincipalComponentAnalysisTransform -out $PBS_O_WORKDIR/Datasets/half_rel_160d/output
/pca/114'
job[3]='java -jar $PBS_O_WORKDIR/Code/elki/out/artifacts/elki_jar/elki.jar KDDCLIApplication  -evaluator
NoAutomaticEvaluation -resulthandler ResultWriter -out.silentoverwrite true -dbc.in $PBS_O_WORKDIR/Datasets
/half_rel_160d/half_rel_160d.csv -parser.labelIndices 161 -algorithm clustering.kmeans.KMeansLloyd -kmeans.
initialization SampleKMeansInitialization -kmeans.algorithm KMeansLloyd -kmeans.samplesize 0.1 -kmeans.k 115  -
dbc.filter transform.GlobalPrincipalComponentAnalysisTransform -out $PBS_O_WORKDIR/Datasets/half_rel_160d/output
/pca/115'

# Select current call
run=${job[$PBS_ARRAY_INDEX]}

#cp -r $PBS_O_WORKDIR/* $SCRATCHDIR/.
#cd $SCRATCHDIR

echo >> $LOGFILE
echo "STARTING..." >> $LOGFILE
echo "--------------------------" >> $LOGFILE

eval $run

#cp -r "$SCRATCHDIR"/* $PBS_O_WORKDIR/.
cd $PBS_O_WORKDIR

echo >> $LOGFILE
qstat -f $PBS_JOBID >> $LOGFILE

echo "$PBS_JOBID ($PBS_JOBNAME) @ `hostname` at `date` in "$RUNDIR" END" >> $LOGFILE
echo "`date +"%d.%m.%Y-%T"`" >> $LOGFILE
```

In der Variable $job ist in diesem Beispiel ein Array definiert welches jeweils den kompletten Prozess-Aufruf beinhaltet. Dieser wird dann per eval ausgeführt.

Alternativ könnte man auch eine Datei mit zeilenweise jeweils einem Parameter-Satz anlegen und die passende Zeile aus der Datei laden.

Beispiel 2: Parameter in Datei

```
#!/bin/bash
#PBS -l select=1:ncpus=1:mem=2gb:arch=ivybridge
#PBS -l walltime=72:00:00
#PBS -N Bachelorarbeit
#PBS -A Benchmarking
#PBS -r y

me=`basename $0`
LOGFILE=$PBS_O_WORKDIR/$PBS_JOBNAME"."$PBS_JOBID"_"$PBS_ARRAY_INDEX".log"

#SCRATCHDIR=/scratch_gs/$USER/$PBS_JOBID/
#mkdir -p "$SCRATCHDIR"

cd $PBS_O_WORKDIR

echo "$PBS_JOBID ($PBS_JOBNAME) @ `hostname` at `date` in "$RUNDIR" START" > $LOGFILE
echo "`date +"%d.%m.%Y-%T"`" >> $LOGFILE

echo >> $LOGFILE
echo "GLOBAL PARAMETERS" >> $LOGFILE
echo "--------------------------" >> $LOGFILE
echo "Node      : "`hostname` >> $LOGFILE
echo "RunDir    : "$PBS_O_WORKDIR >> $LOGFILE
#echo "ScratchDir: "$SCRATCHDIR >> $LOGFILE
echo "# CPUs    : "$NCPUS >> $LOGFILE
echo "# Threads : "$OMP_NUM_THREADS >> $LOGFILE

shopt -s extglob

parameterFile="parameter_per_job.input"

# Select current call
run=$(sed "${$PBS_ARRAY_INDEX}q;d" $parameterFile}

#cp -r $PBS_O_WORKDIR/* $SCRATCHDIR/.
#cd $SCRATCHDIR

echo >> $LOGFILE
echo "STARTING..." >> $LOGFILE
echo "--------------------------" >> $LOGFILE

eval $run

#cp -r "$SCRATCHDIR"/* $PBS_O_WORKDIR/.
cd $PBS_O_WORKDIR

echo >> $LOGFILE
qstat -f $PBS_JOBID >> $LOGFILE

echo "$PBS_JOBID ($PBS_JOBNAME) @ `hostname` at `date` in "$RUNDIR" END" >> $LOGFILE
echo "`date +"%d.%m.%Y-%T"`" >> $LOGFILE
```

Dabei sieht dann die Datei `parameter_per_job.input` wie folgt aus:

```
java -jar $PBS_O_WORKDIR/Code/elki/out/artifacts/elki_jar/elki.jar KDDCLIApplication  -evaluator
NoAutomaticEvaluation -resulthandler ResultWriter -out.silentoverwrite true -dbc.in $PBS_O_WORKDIR/Datasets
/half_rel_160d/half_rel_160d.csv -parser.labelIndices 161 -algorithm clustering.kmeans.KMeansLloyd -kmeans.
initialization SampleKMeansInitialization -kmeans.algorithm KMeansLloyd -kmeans.samplesize 0.1 -kmeans.k 113  -
dbc.filter transform.GlobalPrincipalComponentAnalysisTransform -out $PBS_O_WORKDIR/Datasets/half_rel_160d/output
/pca/113
java -jar $PBS_O_WORKDIR/Code/elki/out/artifacts/elki_jar/elki.jar KDDCLIApplication  -evaluator
NoAutomaticEvaluation -resulthandler ResultWriter -out.silentoverwrite true -dbc.in $PBS_O_WORKDIR/Datasets
/half_rel_160d/half_rel_160d.csv -parser.labelIndices 161 -algorithm clustering.kmeans.KMeansLloyd -kmeans.
initialization SampleKMeansInitialization -kmeans.algorithm KMeansLloyd -kmeans.samplesize 0.1 -kmeans.k 114  -
dbc.filter transform.GlobalPrincipalComponentAnalysisTransform -out $PBS_O_WORKDIR/Datasets/half_rel_160d/output
/pca/114
java -jar $PBS_O_WORKDIR/Code/elki/out/artifacts/elki_jar/elki.jar KDDCLIApplication  -evaluator
NoAutomaticEvaluation -resulthandler ResultWriter -out.silentoverwrite true -dbc.in $PBS_O_WORKDIR/Datasets
/half_rel_160d/half_rel_160d.csv -parser.labelIndices 161 -algorithm clustering.kmeans.KMeansLloyd -kmeans.
initialization SampleKMeansInitialization -kmeans.algorithm KMeansLloyd -kmeans.samplesize 0.1 -kmeans.k 115  -
dbc.filter transform.GlobalPrincipalComponentAnalysisTransform -out $PBS_O_WORKDIR/Datasets/half_rel_160d/output
/pca/115
```

Tipp: Für weitere Details siehe auch User Manuals PBSPro User´s Guide Kapitel 8 "Job Arrays".