

Thu 06 Apr

Noon

Evening

Unix Tools, Scripting, and Parallel Computing

Sat 08 Apr

Noon

Evening

Why do we still need the shell? (Psst... the command line is 50 years old)

- ▶ Scripting and automation
- ▶ Everything you can do in a GUI, you can do in a CLI
- ▶ Very old, very new, or very simple programs
- ▶ all designed to be chained together
- ▶ low-overhead, no-frills system access
- ▶ It's super fast!



To follow along interactively git clone
the workshop repository

```
git clone https://github.com/lpipes/unix_workshop.git
```

```
brew install parallel
```



GitHub

Exercise I a: Renaming files



Exercise I a: Renaming files

DOG_125.jpg	DOG_182.jpg	DOG_239.jpg	DOG_296.jpg	DOG_352.jpg	DOG_409.jpg	DOG_466.jpg
DOG_126.jpg	DOG_183.jpg	DOG_24.jpg	DOG_297.jpg	DOG_353.jpg	DOG_41.jpg	DOG_467.jpg
DOG_127.jpg	DOG_184.jpg	DOG_240.jpg	DOG_298.jpg	DOG_354.jpg	DOG_410.jpg	DOG_468.jpg
DOG_128.jpg	DOG_185.jpg	DOG_241.jpg	DOG_299.jpg	DOG_355.jpg	DOG_411.jpg	DOG_469.jpg
DOG_129.jpg	DOG_186.jpg	DOG_242.jpg	DOG_3.jpg	DOG_356.jpg	DOG_412.jpg	DOG_47.jpg
DOG_13.jpg	DOG_187.jpg	DOG_243.jpg	DOG_30.jpg	DOG_357.jpg	DOG_413.jpg	DOG_470.jpg
DOG_130.jpg	DOG_188.jpg	DOG_244.jpg	DOG_300.jpg	DOG_358.jpg	DOG_414.jpg	DOG_471.jpg
DOG_131.jpg	DOG_189.jpg	DOG_245.jpg	DOG_301.jpg	DOG_359.jpg	DOG_415.jpg	DOG_472.jpg
DOG_132.jpg	DOG_19.jpg	DOG_246.jpg	DOG_302.jpg	DOG_36.jpg	DOG_416.jpg	DOG_473.jpg
DOG_133.jpg	DOG_190.jpg	DOG_247.jpg	DOG_303.jpg	DOG_360.jpg	DOG_417.jpg	DOG_474.jpg
DOG_134.jpg	DOG_191.jpg	DOG_248.jpg	DOG_304.jpg	DOG_361.jpg	DOG_418.jpg	DOG_475.jpg
DOG_135.jpg	DOG_192.jpg	DOG_249.jpg	DOG_305.jpg	DOG_362.jpg	DOG_419.jpg	DOG_476.jpg
DOG_136.jpg	DOG_193.jpg	DOG_25.jpg	DOG_306.jpg	DOG_363.jpg	DOG_42.jpg	DOG_477.jpg
DOG_137.jpg	DOG_194.jpg	DOG_250.jpg	DOG_307.jpg	DOG_364.jpg	DOG_420.jpg	DOG_478.jpg
DOG_138.jpg	DOG_195.jpg	DOG_251.jpg	DOG_308.jpg	DOG_365.jpg	DOG_421.jpg	DOG_479.jpg
DOG_139.jpg	DOG_196.jpg	DOG_252.jpg	DOG_309.jpg	DOG_366.jpg	DOG_422.jpg	DOG_48.jpg
DOG_14.jpg	DOG_197.jpg	DOG_253.jpg	DOG_31.jpg	DOG_367.jpg	DOG_423.jpg	DOG_480.jpg
DOG_140.jpg	DOG_198.jpg	DOG_254.jpg	DOG_310.jpg	DOG_368.jpg	DOG_424.jpg	DOG_481.jpg
DOG_141.jpg	DOG_199.jpg	DOG_255.jpg	DOG_311.jpg	DOG_369.jpg	DOG_425.jpg	DOG_482.jpg
DOG_142.jpg	DOG_2.jpg	DOG_256.jpg	DOG_312.jpg	DOG_37.jpg	DOG_426.jpg	DOG_483.jpg
DOG_143.jpg	DOG_20.jpg	DOG_257.jpg	DOG_313.jpg	DOG_370.jpg	DOG_427.jpg	DOG_484.jpg
DOG_144.jpg	DOG_200.jpg	DOG_258.jpg	DOG_314.jpg	DOG_371.jpg	DOG_428.jpg	DOG_485.jpg
DOG_145.jpg	DOG_201.jpg	DOG_259.jpg	DOG_315.jpg	DOG_372.jpg	DOG_429.jpg	DOG_486.jpg
DOG_146.jpg	DOG_202.jpg	DOG_26.jpg	DOG_316.jpg	DOG_373.jpg	DOG_43.jpg	DOG_487.jpg
DOG_147.jpg	DOG_203.jpg	DOG_260.jpg	DOG_317.jpg	DOG_374.jpg	DOG_430.jpg	DOG_488.jpg
DOG_148.jpg	DOG_204.jpg	DOG_261.jpg	DOG_318.jpg	DOG_375.jpg	DOG_431.jpg	DOG_489.jpg
DOG_149.jpg	DOG_205.jpg	DOG_262.jpg	DOG_319.jpg	DOG_376.jpg	DOG_432.jpg	DOG_49.jpg
DOG_15.jpg	DOG_206.jpg	DOG_263.jpg	DOG_32.jpg	DOG_377.jpg	DOG_433.jpg	DOG_490.jpg
DOG_150.jpg	DOG_207.jpg	DOG_264.jpg	DOG_320.jpg	DOG_378.jpg	DOG_434.jpg	DOG_491.jpg
DOG_151.jpg	DOG_208.jpg	DOG_265.jpg	DOG_321.jpg	DOG_379.jpg	DOG_435.jpg	DOG_492.jpg
DOG_152.jpg	DOG_209.jpg	DOG_266.jpg	DOG_322.jpg	DOG_38.jpg	DOG_436.jpg	DOG_493.jpg

Exercise I a: Renaming files

```
cd Exercise_1a
```

```
for i in {1..500}
do
mv DOG_${i}.jpg RAMBO_${i}.jpg
done
```

Exercise 1b: Renaming files (in parallel)

```
cd Exercise_1b
```

```
for i in {1..8}  
do  
mv DOG_${i}.jpg RAMBO_${i}.jpg &  
done
```

Exercise | c: For loops

16S_K1354-G5-S47_R1.fastq	16S_K1481-T9-S76_R1.fastq	16S_WK1355-C3-S4_R1.fastq	16S_WK1482-K6-S23_R1.fastq
16S_K1354-G5-S47_R2.fastq	16S_K1481-T9-S76_R2.fastq	16S_WK1355-C3-S4_R2.fastq	16S_WK1482-K6-S23_R2.fastq
16S_K1354-K6-S50_R1.fastq	16S_K1482-A1-S86_R1.fastq	16S_WK1355-E4-S19_R1.fastq	16S_WK1482-M8-S11_R1.fastq
16S_K1354-K6-S50_R2.fastq	16S_K1482-A1-S86_R2.fastq	16S_WK1355-E4-S19_R2.fastq	16S_WK1482-M8-S11_R2.fastq
16S_K1354-M8-S63_R1.fastq	16S_K1482-B2-S79_R1.fastq	16S_WK1355-G5-S28_R1.fastq	16S_WK1482-T9-S12_R1.fastq
16S_K1354-M8-S63_R2.fastq	16S_K1482-B2-S79_R2.fastq	16S_WK1355-G5-S28_R2.fastq	16S_WK1482-T9-S12_R2.fastq
16S_K1355-A1-S58_R1.fastq	16S_K1482-C3-S85_R1.fastq	16S_WK1355-K6-S32_R1.fastq	16S_WK1495-A1-S36_R1.fastq
16S_K1355-A1-S58_R2.fastq	16S_K1482-C3-S85_R2.fastq	16S_WK1355-K6-S32_R2.fastq	16S_WK1495-A1-S36_R2.fastq
16S_K1355-C3-S84_R1.fastq	16S_K1482-E4-S74_R1.fastq	16S_WK1355-L7-S37_R1.fastq	16S_WK1495-B2-S24_R1.fastq
16S_K1355-C3-S84_R2.fastq	16S_K1482-E4-S74_R2.fastq	16S_WK1355-L7-S37_R2.fastq	16S_WK1495-B2-S24_R2.fastq
16S_K1355-E4-S78_R1.fastq	16S_K1482-K6-S65_R1.fastq	16S_WK1355-M8-S31_R1.fastq	16S_WK1495-E4-S30_R1.fastq
16S_K1355-E4-S78_R2.fastq	16S_K1482-K6-S65_R2.fastq	16S_WK1355-M8-S31_R2.fastq	16S_WK1495-E4-S30_R2.fastq
16S_K1355-G5-S71_R1.fastq	16S_K1482-M8-S82_R1.fastq	16S_WK1355-T9-S3_R1.fastq	16S_WK1495-G5-S16_R1.fastq
16S_K1355-G5-S71_R2.fastq	16S_K1482-M8-S82_R2.fastq	16S_WK1355-T9-S3_R2.fastq	16S_WK1495-G5-S16_R2.fastq
16S_K1355-K6-S56_R1.fastq	16S_K1482-T9-S81_R1.fastq	16S_WK1356-A1-S7_R1.fastq	16S_WK1495-K6-S26_R1.fastq
16S_K1355-K6-S56_R2.fastq	16S_K1482-T9-S81_R2.fastq	16S_WK1356-A1-S7_R2.fastq	16S_WK1495-K6-S26_R2.fastq
16S_K1355-L7-S59_R1.fastq	16S_K1495-A1-S54_R1.fastq	16S_WK1356-B2-S1_R1.fastq	16S_blank-1-S5_R1.fastq
16S_K1355-L7-S59_R2.fastq	16S_K1495-A1-S54_R2.fastq	16S_WK1356-B2-S1_R2.fastq	16S_blank-1-S5_R2.fastq
16S_K1355-M8-S66_R1.fastq	16S_K1495-B2-S72_R1.fastq	16S_WK1356-E4-S33_R1.fastq	16S_blank-2-S17_R1.fastq
16S_K1355-M8-S66_R2.fastq	16S_K1495-B2-S72_R2.fastq	16S_WK1356-E4-S33_R2.fastq	16S_blank-2-S17_R2.fastq
16S_K1355-T9-S48_R1.fastq	16S_K1495-C3-S67_R1.fastq	16S_WK1356-G5-S35_R1.fastq	16S_blank-3-S44_R1.fastq
16S_K1355-T9-S48_R2.fastq	16S_K1495-C3-S67_R2.fastq	16S_WK1356-G5-S35_R2.fastq	16S_blank-3-S44_R2.fastq
16S_K1356-A1-S80_R1.fastq	16S_K1495-E4-S64_R1.fastq	16S_WK1356-K6-S2_R1.fastq	16S_blank-4-S53_R1.fastq
16S_K1356-A1-S80_R2.fastq	16S_K1495-E4-S64_R2.fastq	16S_WK1356-K6-S2_R2.fastq	16S_blank-4-S53_R2.fastq
16S_K1356-B2-S62_R1.fastq	16S_K1495-G5-S77_R1.fastq	16S_WK1356-L7-S25_R1.fastq	16S_blank-5-S75_R1.fastq
16S_K1356-B2-S62_R2.fastq	16S_K1495-G5-S77_R2.fastq	16S_WK1356-L7-S25_R2.fastq	16S_blank-5-S75_R2.fastq
16S_K1356-C3-S61_R1.fastq	16S_K1495-K6-S70_R1.fastq	16S_WK1356-M8-S34_R1.fastq	16S_blank-6-S89_R1.fastq
16S_K1356-C3-S61_R2.fastq	16S_K1495-K6-S70_R2.fastq	16S_WK1356-M8-S34_R2.fastq	16S_blank-6-S89_R2.fastq
16S_K1356-E4-S51_R1.fastq	16S_WK1159-C3-S21_R1.fastq	16S_WK1481-L7-S9_R1.fastq	16S_blank-7-S90_R1.fastq
16S_K1356-E4-S51_R2.fastq	16S_WK1159-C3-S21_R2.fastq	16S_WK1481-L7-S9_R2.fastq	16S_blank-7-S90_R2.fastq
16S_K1356-G5-S55_R1.fastq	16S_WK1160-E4-S22_R1.fastq	16S_WK1481-M8-S10_R1.fastq	16S_blank-8-S91_R1.fastq
16S_K1356-G5-S55_R2.fastq	16S_WK1160-E4-S22_R2.fastq	16S_WK1481-M8-S10_R2.fastq	16S_blank-8-S91_R2.fastq
16S_K1356-K6-S60_R1.fastq	16S_WK1354-B2-S39_R1.fastq	16S_WK1481-T9-S18_R1.fastq	16S_pcr-blank-1-S92_R1.fastq
16S_K1356-K6-S60_R2.fastq	16S_WK1354-B2-S39_R2.fastq	16S_WK1481-T9-S18_R2.fastq	16S_pcr-blank-1-S92_R2.fastq

Exercise 1c: For loops

```
cd Exercise_1c
```

```
for file in *.fastq
do
gzip ${file}
done
```

Exercise 1c: Using `find`

```
find . -name “*.fastq” | xargs -I {}  
gzip {}
```

Exercise 1c: Using `find` and `parallel`

```
find . -name "*.fastq" | parallel -j 4  
"gzip {}"
```

```
find . -name "*.fastq" | sed 's/\.\///g'  
| parallel -j 4 "gzip {}"
```

```
find . -name "*.fastq" | sed 's/\.\///g'  
| sed 's/\.\.fastq//g' | parallel -j 4  
"gzip {}.fastq &> {}_resultslog"
```

Exercise 1d: Writing a bash script (Exercise_1c.sh) for benchmarking

```
#!/bin/bash

file=$1
name=`basename ${file} . fastq`
logFile="$name"_runlog"
timeLog=Exercise_1d/"$name"_time.out"
cp Exercise_1c/${file} Exercise_1d
/usr/bin/time -o ${timeLog} -p bash -c
" gzip Exercise_1d/${file} &>
Exercise_1d/${logFile}"
```

Exercise 1d: Wrapping up your bash script

```
#!/bin/bash

find Exercise_1c -name "*.fastq" | sed
's/Exercise_1c\///g' | parallel -j 4
"./Exercise_1c.sh {}"
```

Exercise 1e: Multiple arguments using `parallel`

`Exercise_1e.txt` (made with `change_case.py`):

16S_K1356-L7-S69_R2.fastq	k1356-l7-s69_r2
16S_K1482-A1-S86_R2.fastq	k1482-a1-s86_r2
16S_WK1481-M8-S10_R2.fastq	wk1481-m8-s10_r2
16S_WK1354-E4-S6_R1.fastq	wk1354-e4-s6_r1
16S_pcr-blank-1-S92_R2.fastq	pcr-blank-1-s92_r2
16S_K1354-C3-S52_R1.fastq	k1354-c3-s52_r1
16S_K1355-A1-S58_R2.fastq	k1355-a1-s58_r2
16S_WK1482-C3-S20_R1.fastq	wk1482-c3-s20_r1
16S_WK1354-M8-S42_R1.fastq	wk1354-m8-s42_r1
16S_K1354-E4-S57_R1.fastq	k1354-e4-s57_r1

Exercise 1e: Multiple arguments using `parallel`

```
parallel -j 4 --colsep '\t' -a  
Exercise_1e.txt ./Exercise_1e.sh {1}  
{2}
```

Now that you can run parallel jobs on your computer, let's move on to running parallel jobs on an HPC cluster



Logging on to Savio

```
ssh [USERNAME]@hpc.brc.berkeley.edu  
sftp [USERNAME]@dtn.brc.berkeley.edu
```

Scratch directory

```
cd /global/scratch/users/lpipes
```

Writing a slurm script (test.slurm)

```
#!/bin/bash
#SBATCH --job-name=[TEST]
#SBATCH --account=[ACCOUNT_NAME]
#SBATCH --partition=[PARTITION_NAME]
#SBATCH --time=00:00:30

echo "hello world"
```

Savio

partitions

([https://docs-](https://docs-research-it.berkeley.edu/services/high-performance-computing/user-guide/hardware-config/)

research-

[it.berkeley.edu/](https://docs-research-it.berkeley.edu/services/high-performance-computing/user-guide/hardware-config/)

services/high-

performance-

computing/

user-guide/

hardware-

config/)

Partition	Nodes	Node Features	Nodes shared?	SU/core hour ratio
savio	132	savio	exclusive	0.75
savio_bigmem	4	savio_bigmem or savio_m512	exclusive	1.67
savio2	163	savio2 or savio2_c24 or savio2_c28	exclusive	1.00
savio2_bigmem	36	savio2_bigmem or savio2_m128	exclusive	1.20
savio2_htc	20	savio2_htc	shared	1.20
savio2_gpu	17	savio2_gpu	shared	2.67 (5.12 / GPU)
savio2_1080ti	8	savio2_1080ti	shared	1.67 (3.34 / GPU)
savio2_knl	28	savio2_knl	exclusive	0.40
savio3	184	savio3 or savio3_c40	exclusive	1.00
savio3_bigmem	20	savio3_bigmem or savio3_m384	exclusive	2.67
savio3_htc	24	savio3_htc or savio3_c40	shared	2.67
savio3_xlmem	4	savio3_xlmem or savio3_c52	exclusive	TBD
savio3_gpu	2	savio3_gpu (2x V100)	shared	TBD
savio3_gpu	9	4rtx (4x GTX)	shared	TBD
savio3_gpu	10	8rtx, 8a5k (8 GPU)	shared	TBD
savio3_gpu	16	a40 (2 GPU)	shared	TBD
savio4_htc	32	savio4_m256 or savio4_m512	shared	TBD

Writing a slurm script (test.slurm) with more options

```
#!/bin/bash
#SBATCH --job-name=[TEST]
#SBATCH --account=[ACCOUNT_NAME]
#SBATCH --partition=[PARTITION_NAME]
#SBATCH --time=00:00:30
#SBATCH --nodes=[# NODES]
#SBATCH --ntasks-per-node=[# TASKS/NODE]
#SBATCH --cpus-per-task=[# CPUS/TASK]
#SBATCH --mail-type=START,END,FAIL
#SBATCH --mail-user=[YOUR EMAIL]

echo "hello world"
```

Thank you!

Please email me if you have any
feedback or have any questions:
lpipes@berkeley.edu