

## 8 Appendix

No. of jobs	1	1	1	1	1	10	10	10	10	10
Remote	5,59	5,7	5,66	5,68	5,64	21,92	21,92	21,3	21,95	22,54
Local	6,4	6,38	6,37	6,35	6,37	7,14	7,11	6,79	7,11	7,21
No. of jobs	25	25	25	25	25	50	50	50	50	50
Remote	42,59	42,76	42,42	42,28	43,16	77,33	84,18	77,78	79,26	73,04
Local	7,67	8,09	7,73	5,86	5,94	9,17	9,18	7,69	7,77	7,35
No. of jobs	100	100	100	100	100	250	250	250	250	250
Remote	159,37	159,52	164,44	153,63	158,56	342,49	348,44	348,29	344,29	346,85
Local	10,96	12,69	14,78	15,48	14,8	22,31	25,16	21,76	23,57	24,02
No. of jobs	500	500	500	500	500					
Remote	695,77	704,79	731,09	709,43	726,74					
Local	44,83	43,99	46,07	45,67	39,18					

Table 3: Testdata for number of jobs test with 1 MB transfer.

No.jobs	1	10	25	50	100	250	500
Remote	5,654	21,926	42,642	78,318	159,104	346,072	713,564
Local	6,374	7,072	7,058	8,232	13,742	23,364	43,948
Overhead in %	0%	347%	849%	1701%	1693%	2629%	2731%

Table 4: Average values and the overhead of using the remote solution for No. of jobs test with 1MB. Values are in seconds, except overhead which is in percent

Size in MB	1	1	1	1	1	5	5	5	5	5
Remote	5,59	5,7	5,66	5,68	5,64	6,12	5,97	5,9	5,88	5,93
Local	6,4	6,38	6,37	6,35	6,37	6,41	6,39	6,36	6,39	6,39
Size in MB	25	25	25	25	25	50	50	50	50	50
Remote	7,18	11,37	11,36	7,23	11,57	10,77	11,31	11,15	11,16	11,12
Local	6,52	6,45	6,51	6,48	6,5	6,61	6,62	6,64	6,62	6,6
Size in MB	100	100	100	100	100	500	500	500	500	500
Remote	12,13	12,1	11,94	11,99	11,94	47,18	46,25	46,82	41,91	52,05
Local	6,53	6,85	6,53	6,53	6,51	7,18	7,1	7,13	7,16	7,13
Size in MB	1000	1000	1000	1000	1000	5000	5000	5000	5000	5000
Remote	82,37	81,16	76,05	77,82	72,5	345,96	370,92	373,42	376,66	371,84
Local	6,04	6	5,99	6,22	6,03	17,05	14,01	15,2	19,76	14,58
Size in MB	10000	10000	10000	10000	10000					
Remote	772,91	774,44	753,95	761,29	769,04					
Local	31,69	31,23	32,77	36,18	36,22					

Table 5: All data points for file transfer test for a single job. Values are in seconds

Size in MB	1	10	25	50	100	500	1000	5000	10000
Remote	5,654	5,96	9,742	11,102	12,02	46,842	77,98	367,76	766,326
Local	6,374	6,388	6,492	6,618	6,59	7,14	6,056	16,12	33,618
Overhead in %	0%	347%	849%	1701%	1693%	2629%	2731%		

Table 6: Average values and the overhead of using the remote solution for increasing inputfile size test with a single job. Values are in seconds, except overhead which is in percent

No. Of Jobs	1	1	1	1	1	10	10	10	10	10
srun	5,83	5,72	5,83	5,69	5,67	20,87	22,45	21,06	22,71	22,57
sbatch	6	5,8	6,06	6,11	5,99	32,01	35,82	36,05	36,49	37,62
No. Of Jobs	50	50	50	50	50	100	100	100	100	100
srun	82,73	82,22	82,65	78,26	78,23	153,49	147,74	149,92	149,66	149,67
sbatch	156,95	158,63	157,59	158,25	157,68	305,3	321,12	303,51	309,87	305,67
No. Of Jobs	250	250	250	250	250					
srun	364,11	368,37	366,43	369,13	359,68					
sbatch	758,93	757,99	754,25	757,35	754,41					

Table 7: All data points for SLURM method comparison test for various number of jobs. Values are in seconds

No. of jobs	1	10	50	100	250
Avg. srun	5,748	21,932	80,818	150,096	365,544
Avg. sbatch	5,992	35,598	157,82	309,094	756,586
Avg. bash	5,654	21,926	78,318	159,104	346,072

Table 8: Average values for execution time in seconds for each SLURM method. The direct approach is included for comparison.

```
import os
import time
infile = "somewhere"+ os.path.sep +"particular"
outfile = "nowhere"+ os.path.sep +"particular"

with open(infile , 'r') as file:
    while True:
        s = str(file.read(104857600))
        if not s:
            break
result = "A"
os.makedirs(os.path.dirname(outfile), exist_ok=True)
```

```
with open(outfile , 'w') as file:  
    file.write(str(result))
```

Listing 1: Recipe used for all tests except the Sleep test.

```
import os  
import time  
time.sleep(1)  
outfile = "nowhere"+ os.path.sep +"particular"  
  
result = "A"  
os.makedirs(os.path.dirname(outfile), exist_ok=True)  
  
with open(outfile , 'w') as file:  
    file.write(str(result))
```

Listing 2: Recipe used for the Sleep test