

Star Midterm Product

Summary:

Team Star's solution is to integrate SequenceServer and WorkQueue to run workers on an HPC. To do this, we initialized an Atmosphere Virtual Machine Instance with CCTools to ensure that we could utilize WorkQueue. Team blastEasy provided a SeqInIt.py script which starts SequenceServer and initiates the WorkQueue master file. Once we adjusted the script to meet our needs, we were able to establish a connection between the Virtual Machine, which acts as the master for WorkQueue, and the HPC. From this, we were able to bring up the SequenceServer GUI in a browser using an IP address and port number.

Description:

Team blastEasy's integration of WorkQueue limits the amount of possible blast searches occurring at the same time. Team Star has taken this a step further and improved scalability by using the power of an HPC.

In our solution, the Team blastEasy sequencesserver2.0 Virtual Machine Instance is used as the Master. Simply, upon starting Sequencesserver, the Master is up and running, waiting for the workers to accept jobs.

At the same time, on an HPC, a flexible .pbs file allows the HPC to be populated with workers. This .pbs can be modified to scale depending on class size and required power. Once the .pbs is submitted and the workers are up and running, the Master VM will submit jobs to the Workers on the HPC.

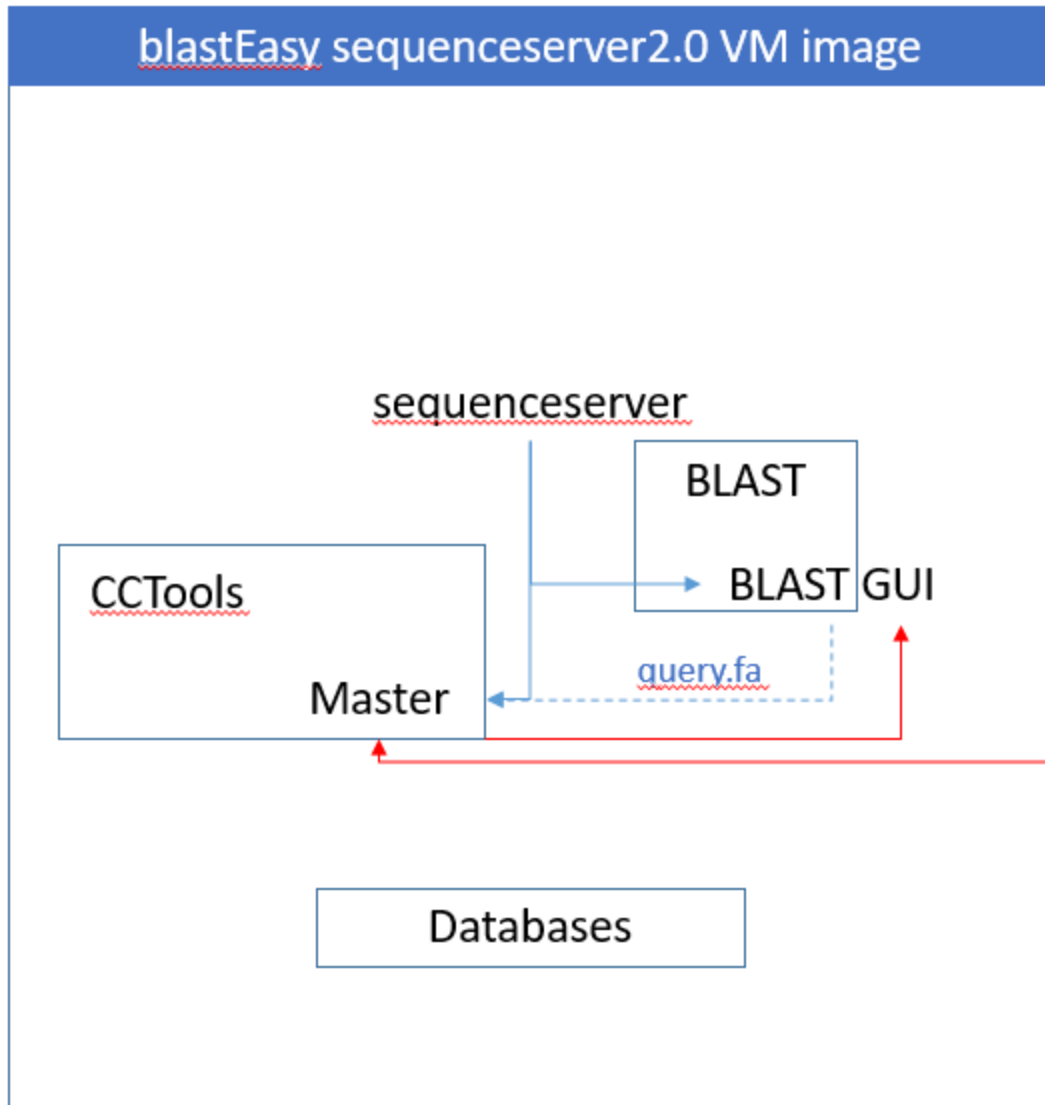
The beauty of this method is that it does not have limitations: although made to assist blast searches for large classrooms, it can be scaled down as one may prefer.

Team Star is thankful for the work of team blastEasy, the sequencesserver team, the CCTools team, and all the students and instructors of the UArizona ACIC course.

[For detailed instructions \(requirements, installation, running\) please visit our Github page.](#)

Concept Map:

blastEasy sequencesserver2.0 VM image



Presentation slides:

Team_star_midterm.pptx

Github:

<https://github.com/Midterm-Team-Star>

Benchmarking Data:

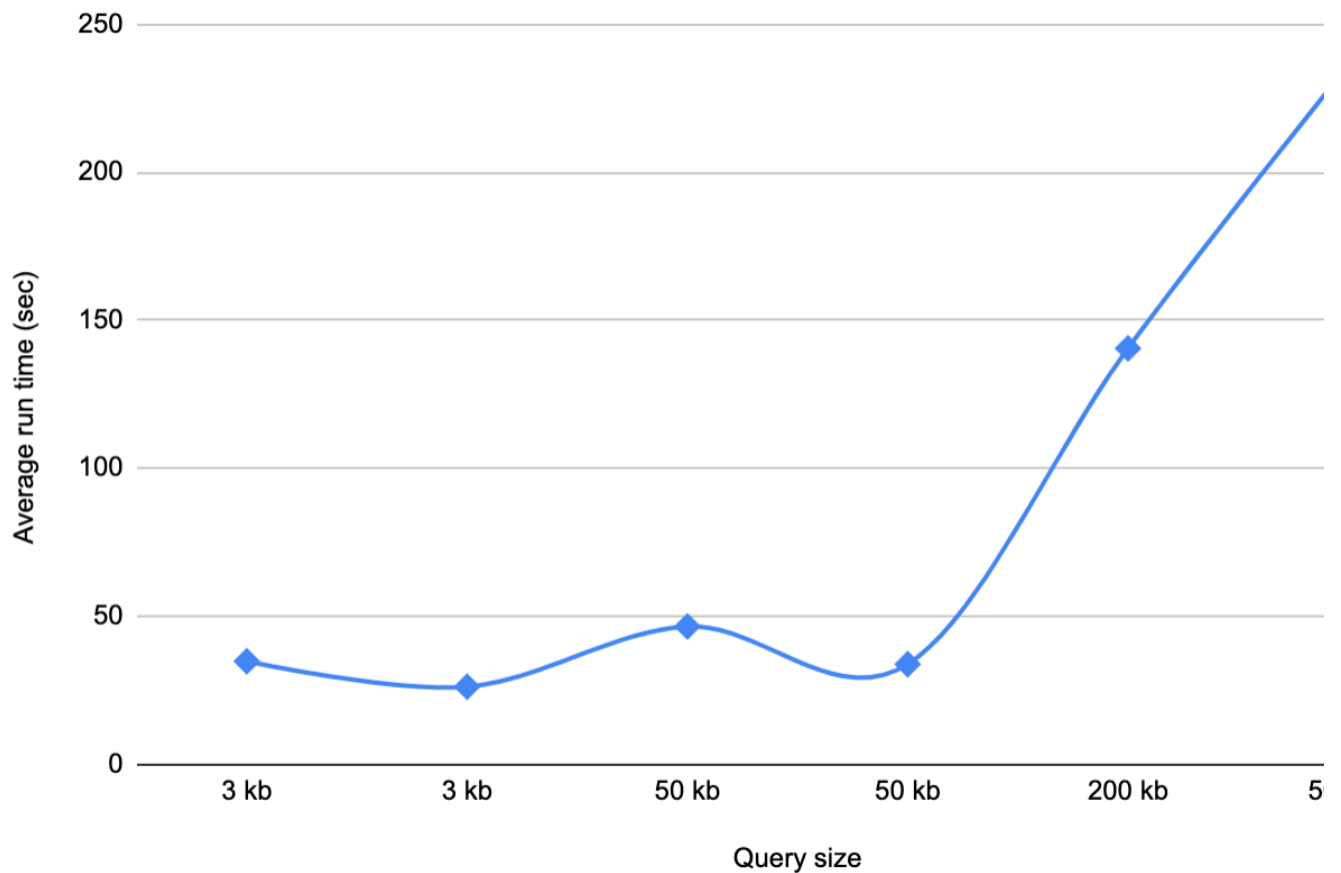
Benchmark was performed using the team blastEasy sequencesserver2.0 VM image as the Master and the HPC as Workers.

Database used: Mouse protein database (46 Mb), custom.

The Benchmarks were performed with 4 different computers simultaneously running the same query search. Each computer would have 4 tabs of sequencesserver and each query was submitted at the same time (therefore 16 simultaneous blasts). Trial times were recorded as when Sequencesserver would display the results.

Notes: more benchmarking could be made - scaling the workers available resources (cpus, ram, nodes) would have resulted in better data.

nodes = 1 , cpus = 6 , mem = 24gb						
	time (trial 1)	time (trial 2)	time (trial 3)	time (trial 4)	time (trial 5)	time (trial 6)
1	26.34	15.83	46.67	31.19	188.07	n/a
2	41.1	34.9	49.05	25.67	152.61	n/a
3	43.6	37.3	50.73	19.19	137.19	n/a
4	47.42	39.37	51.37	13.83	119.19	n/a
5	11.8	12.35	21.35	32.37	120.21	n/a
6	16.04	25.71	33.87	43.01	133.48	n/a
7	17.79	40.99	58.88	43.88	141.98	n/a
8	44.9	43.27	59.81	49.77	155.33	n/a
9	n/a	7.07	16.03	22.13	115.5	n/a
10	n/a	12.97	44.45	23.01	135.13	n/a
11	n/a	20.4	46.48	40.24	141.54	n/a
12	54.01	24.59	55.49	44.34	145.91	n/a
13	9.67	23.15	32.41	24.63	n/a	192.52
14	41.26	23.19	50.01	36.27	n/a	235.4
15	45.64	24.49	62	42.44	n/a	255
16	52.37	32.73	65.8	48.2	n/a	261.65
Average	34.764 61538	26.144 375	46.525	33.760 625	140.51 16667	236.14 25
S.D.	16.043 48394	11.024 75213	14.160 09228	11.272 47708	19.637 39005	31.143 28004
# Runs	16	16	16	16	12	4
Average Sec / # Runs	n/a	1.6340 23438	2.9078 125	2.1100 39063	11.709 30556	59.035 625
Query size	3,336	3,336	50,000	50,000	200,000	500,000
Q Size	3,336	3.336	50	50	200	500



Milestones:

- 1- successful implementation of the master-worker setup within the same machine (tested with work_queue_example.py)
- 2- successful implementation of the master-worker setup between two virtual machines (tested with work_queue_example.py)
- 3- successful implementation of the master-worker setup between a virtual machine and the HPC (tested with work_queue_example.py)
- 4- successfully ran blast search between master virtual machine and worker HPC
- 5- successful implementation of the master-worker setup between the blastEasy sequenceserver virtual machine and the HPC

Team member contributions:

Michele Cosi: Terminal, Documentation, PBS script

Emmanuel Gonzalez: Presentation, Terminal, PBS script

Anthony Dominguez: Coding, Terminal, PBS script

TJ Lippincott: Coding, Terminal, PBS script

Brandi Diesso: Presentation, Documentation

Test Sequence:

https://github.com/Midterm-Team-Star/SS_HPC_Integration_Scripts/blob/master/Test/Mouse_real_gene_3kb.fa

Post-Mortem Analysis:

The Good: pride in the final product; learning challenge.

The Bad: Communication. Need more communication between all parties.

The Ugly: Timing. Should have started earlier.

What to do different: Communicate. Listen. Start earlier.