Checkpointing Technology for Data Centers

CS5352 Course Project, Spring 2021

[This project accepts one student only.]

**Contact:**

Elham Hojati, [elham.hojati@ttu.edu](mailto:elham.hojati@ttu.edu)

**Description**

Parallel programming is one of the features that make high-performance computing environments more powerful. Usually, the process of running Big Data analysis or parallel programing applications takes a long time and a lot of resources such as CPU cores, memory, storage, and power. So failure could be costly. Also, some sensitive applications cannot tolerate failure at all. Therefore, using checkpointing technologies can be very useful. Checkpointing also gives us the ability to put a job in hibernate state whenever we need and resume it later. Parallel computing makes the process of creating a checkpoint, saving that, and recovering the process more complicated and time-consuming.

There are several existing checkpointing solutions, such as SCR [1], DMTCP [2], BLCR [3], IRIX CPR [4], Docker [6], Singularity [7], Mementos, and Idetic [8]. Each of them has pros and cons and could be useful for different types of applications.

This research aims to provide a survey of different checkpointing technologies for parallel computing and HPC Systems, find the advantages and disadvantages of each of them, and implement DMTCP checkpointing method.

**Steps:**

1. Providing a survey of checkpointing technologies for parallel computing and HPC Systems
2. Comparing those technologies
3. Implementing DMTCP checkpointing method

**Requirements:**

* Familiar with parallel computing
* Familiar with High-Performance Computing Systems.

**References:**

1- https://computing.llnl.gov/projects/scalable-checkpoint-restart-for-mpi

2- http://dmtcp.sourceforge.net/

1. https://crd.lbl.gov/departments/computer-science/CLaSS/research/BLCR/
2. http://nixdoc.net/man-pages/IRIX/man1/cpr.1.html
3. https://www.docker.com/
4. https://singularity.lbl.gov/
5. https://ieeexplore.ieee.org/document/6526735