JUNO performance evaluation and optimization on virtual platform

Xiaomei Zhang Zhentai Ma Ying Lv Ce Yang
CHEP2016, San Francisco
Content

- Introduction
- Evaluation Plan
- Results and optimizations
- Automation tool
- Summary
IHEP Computing and Virtual Platform

- Local resources in IHEP
  - ~12000 CPU cores, ~10PB disk and tape

- More HEP experiments are coming, need to manage twice or more servers as today
  - JUNO, HXMT, LAHHSO, CEPC……

- IHEP Virtual Platform is being built up to meet future challenge
  - IHEPCloud on OpenStack
  - A large scale usage through cluster and distributed computing
Jiangmen Underground Neutrino Observatory (JUNO)

- A multi-purpose neutrino experiment designed to measure the neutrino mass hierarchy and mixing parameters
  - Start to build in 2014, operational in 2019, located at Guangzhou province
  - 20 kt Liquid Scintillator detector, 700m deep underground
  - 2-3% energy resolution
  - Rich physics opportunities
  - Estimated to produce **2PB data/year** for 10 years
JUNO Data Processing

- **Software**
  - Three parts: Framework, Offline physics packages, External Libraries
    - [https://indico.cern.ch/event/505613/contributions/2230858/](https://indico.cern.ch/event/505613/contributions/2230858/)

- **Type of Data Processing**
  - Physics Generator (PhyGen) and Detector Simulation (DetSim) -- CPU bound
  - Electronics Simulation (EleSim)
    -- Memory and I/O bound (~20% I/O with local disk)
  - PMT Reconstruction (PmtRec) -- CPU bound
  - Event Reconstruction (EvtRec) -- CPU bound
Evaluation Plan

Indicators
- Basic benchmark
- JUNO software

Automation tool

Analysis Plots

Scenarios
- VM size
- hardware
- OS
- Data access
- KVM configuration

Analysis
Plots
Evaluation Environment

❖ **Hardware**

- **H1**: Intel(R) Xeon(R) CPU E5-2630L v2@2.40GHz (6 core/CPU)
  - disk: HP, MM1000FBFVR
- **H2**: Intel(R) Xeon(R) CPU X5650@ 2.67GHz (6 core/CPU)
  - disk: ST31000340NS
- **H3**: Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz (8 core/CPU)
  - disk: ProLiant BL460c Gen9
- Mem: 4GB/core
- File system: ext4
- KVM: libvirt 0.10.2, image :qcow2

❖ **Software**

- **HOST OS**: Scientific Linux 6.5
- **Guest OS**: Scientific Linux 6.5
- **JUNO software version**: J16v2r1-Pre2
## Basic benchmark

| Tool    | Perf Loss (H1) | PerfLoss (H2) | PerfLoss (H3) |
|---------|----------------|---------------|---------------|
| CPU     | SPECCPU 2006   | 17.68%        | 6.25%         | 8.83%         |
| I/O     | IOZone         | 10%~16%       | 13%~36%       | 4~9%          |
| Memory  | STREAM         | 4.6%          | 6.5%          | 2.7%          |
| Network | NetPerf        | <1%           | <1%           | <1%           |

- KVM Memory consumption is about 667~1810 MB with different VM size
- For IOZone, the loss for Read is less than that of Write
- Different hardware has quite different loss

* Use Virtio-net mode
JUNO software benchmark(1)

- With the increase of event number, performance loss decreases and tend to be stable
  - Initial part caused higher penalty than event processing part
  - Initial part includes loading libraries and necessary parameters, etc

- The effect can be ignored with more than 300 events in single process case, but not for multi-process case
JUNO software benchmark(2)

|        | PerfLoss(H1) | PerfLoss(H2) | PerfLoss(H3) |
|--------|--------------|--------------|--------------|
| DetSim | 9.25%        | 5.5%         | 5.1%         |
| ElecSim| 14.1%        | 18.5%        | 7.5%         |
| ElecSim* (no out) | 3.4%        | 1.8%         | 2.8%         |
| PmtRec | 1.2%         | 0.6%         | 9.5%         |
| EvtRec | 1.4%         | 1.8%         | 1.3%         |

- Three hardware has different loss, H3 has the best I/O performance, H2 has better CPU performance.
- DetSim based on Geant4 has more CPU loss than Rec.
- ElecSim with more I/O has higher penalty than others, which is proved by *ElecSim without output test.
- Most of loss is under 5%, a few need concern.
Findings with DetSim(1)

- Full processes in different size of VMs with H2
  - Profile different VM size
    - 1vm with 12 cores, 2vm with 6cores….., 12vm/1core
  - With increase of VM number, the loss increases from 5% to 20%
  - The worst point is 12 processes, each VM with 1 core
  - For Rec, the loss ranges from 1.8% to 6%

Simulation performance loss in Multi-VMs

reconstruction performance loss
Findings with DetSim(2)

- Investigations in 12 VM case show that initial part is a quite I/O intensive processes. 17 minutes have been spent for the initial part.
  - Only 2 minutes in physical machine.
- Initial part has more than 300% loss than that of 1pm with 12 processes.
Findings with DetSim(3)

- Tests also found that performance loss can be greatly reduced to 6.21%, if initial part of processes avoided to do at the same time.
- With local pre-cache of CVMFS files in 12VM case, the performance loss is reduced to 7%.

![cvmfs pre-cache effects on performance loss](chart.png)
Findings with ElecSim(1)

- Sparse space allocation in Qcow2 cause serious penalty with I/O intensive ElecSim
  - Preallocation=off penalty >100%
    - For new write, OS needs to lookup and allocate a new block to the virtual image incurring a performance penalty
  - Preallocation=full penalty <20%
    - Image is not a sparse file
  - Space growth with growing events incur significant penalty
Findings with ElecSim(2)

- Writeback cache mode is ~10% better than the default writethrough cache mode in ElecSim case
Optimizations with KVM set-up(1)

- Most of Loss is < 5%, methods taken to reduce serious loss
  - “CPU feature adopting from Host” help reduce 3%~5% of CPU loss
    - With option “-cpu host”
  - Disk preallocation avoid sparse space allocation to gain ~80%
  - Writeback cache mode can help gain ~10%

|               | H1(CPUhost)   | H3(CPUhost) |
|---------------|--------------|-------------|
| DetSim        | 9.25%->6.54% | None        |
| PmtSim        | None         | 9.5%->4.5%  |
Optimizations with KVM set-up(2)

- Network file system to hold data instead of local disk can reduce loss by 10%
- Others has slight effects on performance
  - CPU pinning(~1%), KSM(Kernel SamePage Merging), EPT, THP
The above experience has shown that many factors influence performance.

These factors keep changing in future usage:
- New machines and hardware are in
- OS upgraded
- Cloud manager upgraded or have some changes
- JUNO Applications upgraded to new versions

Need to make sure these changes don’t have adverse effects on performance:
- Automatic tool to make it easy
Functions

- Define and manage test cases and processes
- Auto start testing and retrieve back results
- Auto analyze test results

Features

- Modularize indicator tests for easy extension
- Standardize output for later comparisons
- Use DB to keep records for more analysis
Summary

- JUNO Evaluations showed
  - CPU-bound processes are suitable to run on virtualization form
  - I/O penalty is still a key issue in I/O intensive processes
  - Simulation has bigger CPU loss than Reconstruction

- Many factors influence penalty, including hardware, application, KVM parameters, OS….
  - Tuning can achieve certain improvements

- Automatic test and monitoring tool needed to keep watch on performance issues in various scenarios and changing environment