Towards Seamless Integration of Data Analytics into Existing HPC Infrastructures

Michael Gienger

High Performance Computing Center Stuttgart (HLRS), Germany

Redmond – May 11, 2017

Outline

- Introduction to HLRS
- Current challenges in HPC
- Data Analytics @ HLRS
  - Catalyst
  - Urika-GX
- Case study
  - Log file analysis for Cray XC series
- Summary

HLRS

High-Performance Computing Center | Stuttgart

High Performance Computing Center Stuttgart (HLRS)

- Member of the Gauss Centre for Supercomputing
- Basic and applied research
  - Publicly funded national and European projects
  - Focused industrial collaborations
- Consultancy and training activities
- Providing High Performance Computing services
  - Academia
  - Industry

Important HLRS systems

- ***Hazel Hen*** Cray XC40
  - 7.712 nodes
  - 185.088 cores Intel Haswell
  - 7.40 PFLOPS Peak performance
  - 1 PB main memory
  - 12 PB disk storage

- ***Gilgamesch*** & ***Enkidu*** Cray Urika-GX
  - 64 nodes
  - 2.400 cores
  - 33 TB main memory
  - 100 TB HDFS Storage

CURRENT CHALLENGES IN HPC

Challenges in HPC

- Customers tend to run more and more data-intensive applications resulting in vast amounts of output data
  - Single turbulence & acoustics simulation of an axial fan with just four rotations results in 80 TB of data
  - Domain experts are no longer able to analyze data manually in a timely manner
- Today's HPC centers are in need to provide **seamlessly integrated data analytics solutions** to process data ideally on the fly

When HPC meets Big Data

- Big Data Analytics has distinct requirements not met by current HPC architectures
  - Data colocation
  - Recurrent analysis
  - Ever-changing software zoo
  - Scheduling
  - Services
  - Sandboxing

Layer	HPC	Big Data
Programming	C/C++, Fortran Message Passing, Shared Memory	Java, Python Hadoop, Spark
Resource Manager	TORQUE, SLURM	YARN, Mesos, Marathon
File System	Lustre, GPFS, NFS	HDFS
Hardware	Tailored components (e.g. Xeon, InfiniBand)	Commodity components (e.g. 10 GbE)

DATA ANALYTICS @ HLRS

Catalyst

- Project established in 2016 to **evaluate and push the incorporation of data analytics for HPC**
- Cooperation with **Cray** and **Daimler**
  - Real-world case studies with partners from academia and industry
- Focus on the **engineering domain** in comparison to the general application of data analytics for natural sciences
- Integration and evaluation of **2 Cray Urika-GX systems** into the production environment of HLRS
  - Additional requirements concerning **multi-user support** and **security** arise

Urika-GX @ HLRS

	Gilgamesch	Enkidu
Nodes	48	16
Compute Nodes	41	9
CPU	2x Intel BDW 18-core, 2.1 GHz	
RAM	512 GB	
Local Storage	2 x 2 TB HDD; Intel DC P3608 SSD (1.6 TB)	
File System	Sonexion 900 with 240 TB 4.0 GB/s throughput	
Software	❑ YARN, Mesos, Marathon ❑ Hadoop, Spark, Cray Graph Engine, GNU R ❑ Apache Kafka ❑ Apache Hive ❑ …	

System integration

Lustre for HPC
(12 Petabytes)

2 Cray Urika-GX systems

Lustre dedicated to Urika-GX
(240 TB)

Hazel Hen Cray XC40

Integration challenges

- Usage model
  - Single versus multi-user operation
- Software
  - Each customer has specific requirements
- Security
  - Need to guarantee security compliance
- Accounting
  - Multiple resource managers complicate accounting and operation
- Data ingestion and storage
  - System located within the HLRS network

Case study

LOG FILE ANALYSIS CRAY XC40

Diana Moise, Cray Inc.

Motivation

- **Performance variability** on HPC platforms is a critical issue with serious implications for the users
  - **Irregular runtimes** prevent users from correctly assessing performance and from efficiently planning allocated machine time
  - Hundreds of applications concurrently sharing thousands of resources escalate the **complexity** of identifying the causes of runtime variations
- On production systems, implementing trial-and-error approaches is **practically impossible** !

Application interference

- What type of applications can **impact the performance** of other applications?
  - Victims
    - Applications that show high variability
  - Aggressors
    - Applications potentially causing the variability

- Understanding the nature of both types of applications is crucial for developing a meaningful **detection mechanism**

Detecting victims and aggressors

- Implementing **trial-and-error is not feasible**
- Use existing information without loading the system
  - Cray systems **collect large amounts of data** related to user applications
  - **Apply analytics tools** to use the data for identifying and understanding performance variability
- We have developed an Apache Spark based **tool for analyzing system logs** in order to **identify victims and aggressors**

Available input data

- Cray System Management Workstation (SMW) log files
  - Collected at HLRS on the Cray XC40 system
  - Performance data
  - Periods between two weeks and three months
- Job dataset (excerpt, anonymized)
  - Start time
  - End time
  - Elapsed time
  - Execution command
  - Allocated nodes for the execution

Analysis via Apache Spark

Step 1: Data filtering

- Minimum runtime (e.g. 60s)

Step 2: Victim detection

- Baseline approach
  - Average / minimum elapsed time
- Factorized approach
  - x times slower than baseline

Step 3: Aggressor detection

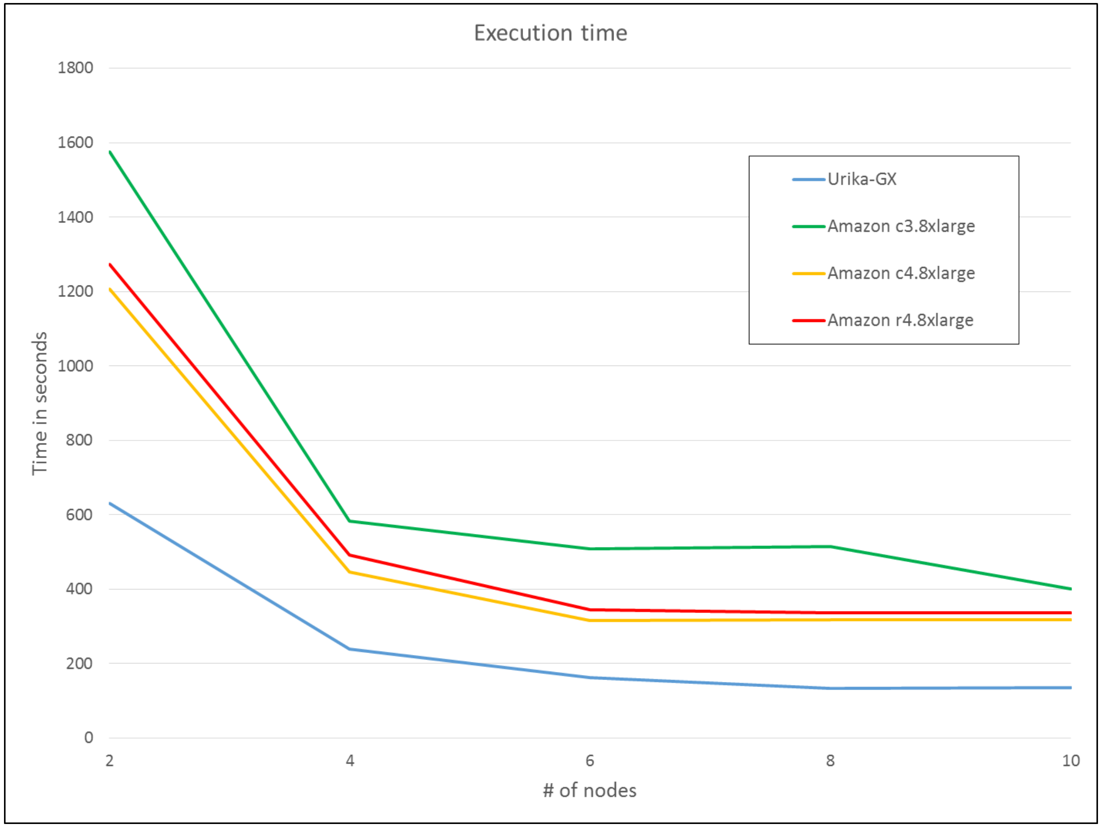
- Execution time overlap with victims
- Number of allocated nodes

Evaluation

- **Setup 1** (60s, 2x, average, #1000)
  - 3.215 victims
  - 67.908 potential aggressors
  - Spark configuration
    - 300 cores, 30 GB RAM
    - Runtime: 268s
- **Setup 2** (60s, 8x, average, #50)
  - 10 victims
  - 211 potential aggressors
  - Spark configuration
    - 15 cores, 15 GB RAM
    - Runtime: 17s
- Identification of common patterns of the most important aggressors was possible
  - **Recommended best practices to users**
  - **Implemented optimizations for system configuration**

Performance evaluation

- Urika-GX
  - Broadwell, 36 cores, PCIe SSD
- Amazon c3
  - Ivy Bridge, 32 cores, SATA SSD
  - Cost: 2.33 $ / hour
- Amazon c4
  - Haswell, 36 cores, no SSD
  - Cost: 2.40 $ / hour
- Amazon r4
  - Broadwell, 32 cores, no SSD
  - Cost: 2.83 $ / hour



SUMMARY

Take-away messages

- Data Analytics @ HLRS
  - Evaluation of Urika-GX in a real production environment
  - Multiple hurdles exist when integrating GX systems into existing infrastructures (e.g. security and accounting)
  - Focus on solutions for the engineering domain
  - Close collaboration with academia and industry
  - **Collaboration partners are always welcome**
- 1st case study on detecting jobs that potentially harm the overall system's performance
  - Next steps include increasing the confidence in identifying potential aggressors via machine learning mechanisms
- Second case study in the engineering domain already underway
  - More to come...

Thank you !

Questions ?

Michael Gienger
High Performance Computing Center Stuttgart
Nobelstrasse 19
70569 Stuttgart
Phone: +49-711-685-63824
Email: gienger@hlrs.de