

# Scaling Uncertainty Quantification Studies to Millions of Jobs

Tamara Dahlgren  
dahlgren1@llnl.gov

David Domyancic  
domyancic1@llnl.gov

Scott Brandon  
brandon1@llnl.gov

Todd Gamblin  
gamblin2@llnl.gov

John Gyllenhaal  
gyllenhaal1@llnl.gov

Rao Nimmakayala  
nimmakayala1@llnl.gov

Richard Klein  
klein4@llnl.gov

Lawrence Livermore National Laboratory

## ABSTRACT

Computer simulations for evaluating impacts of parameters and assessing the likelihood of outcomes typically involve hundreds to millions of executions on hundreds of processors each and can take months to complete using current resources. Reducing the turn-around time by scaling concurrent simulation ensemble runs to millions of processors is hindered by resource constraints. Our poster describes a preliminary investigation of mitigating the impacts of these limitations in the LLNL Uncertainty Quantification Pipeline (UQP) using CRAM. CRAM virtualizes MPI by splitting a program into separate groups, each with its own sub-communicator for running a simulation. We launched a single process problem on all 1.6 million Sequoia cores in under 40 minutes versus 4.5 days. The small problem size resulted in 400,000 concurrent ensemble simulations. Preparations are underway to demonstrate using CRAM to maximize the number of simulations we can run within our allocated partition for an ensemble using a multi-physics package.

## 1. INTRODUCTION

Uncertainty Quantification studies involve statistically analyzing the results of hundreds to millions of simulations. Results accuracy is generally tied to increasing the number of inputs (or sample points). This is a process that can take months to complete using current resources due to hardware and system constraints. Our goals are to reduce the turn-around time and improve machine utilization by performing more simulations concurrently. This poster describes our preliminary investigation into extending the LLNL Uncertainty Quantification Pipeline (UQP) with CRAM to launch ensembles at scale.

## 2. MOTIVATION

Current resource constraints preclude performing large scale, concurrent ensemble simulations. Finite resources and their management are key obstacles. Our experience using machines at LLNL and LANL revealed roadblocks to running more than a few thousand simultaneous UQ simulations even on one of the top three Supercomputers. Key system limits were process counts and front-end Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

SC '15, November 15-20, 2015, Austin, TX, USA  
© 2015 ACM. ISBN 978-1-4503-3723-6/15/11...\$15.00  
DOI: <http://dx.doi.org/10.1145/2807591.2807623>

node (FEN) memory. Many HPC scheduling systems top out at between 256 and 4 K simultaneous runs, given 16 GB and 256 GB of memory, respectively. A machine with 1.6 million cores would require about 96 TB of memory to launch one application per core. Increasing system limits and boosting memory solves part of the problem but the resource manager still limits the total number of concurrent jobs.

## 3. LLNL UQ PIPELINE

The UQP is a scientific workflow system for running and analyzing the uncertainty of concurrent simulations for UQ studies [5]. Using a simple, non-intrusive interface, the UQP provides the following capabilities:

- sampling high dimensional uncertainty spaces;
- generating ensembles of simulations;
- analyzing ensemble output;
- constructing surrogate models;
- performing sensitivity studies;
- incorporating observational data;
- performing statistical inferences; and
- estimating parameter values and probability distributions.

## 4. CRAM

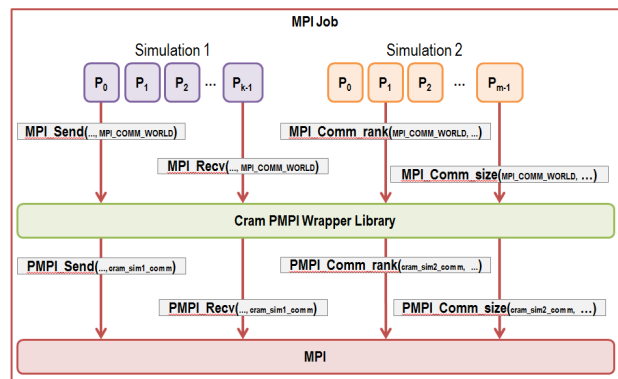
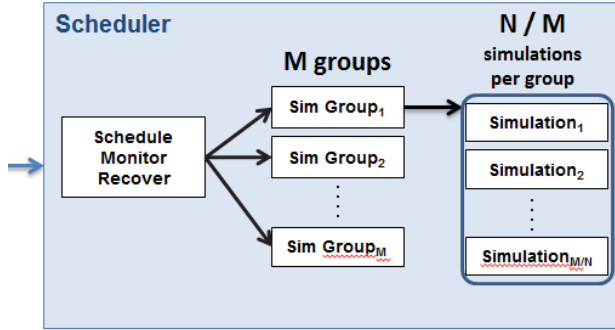


Figure 1: CRAM intercepts and translates MPI calls to allow many simulations to execute concurrently in a single run.

CRAM is a tool developed at LLNL for packing a large number of MPI-based simulations into a single simulation run [1, 3]. Command line arguments, environments, and working directories are compressed into an input file used by the CRAM library, which

must be linked with the application. Figure 1 illustrates CRAM intercepting and mapping MPI calls.

## 5. UQP WITH CRAM



**Figure 2: The UQP now packs  $N$  ensemble simulations into  $M$  execution groups with up to  $N/M$  runs per group.**

Sample point processing was modified to divide the ensemble into groups of runs, each packed into a CRAM input file, as shown in Figure 2. File system-related bottlenecks for managing working directories and collecting simulation results were also addressed through a C program that off-loaded the work onto the compute nodes.

## 6. MASSIVELY PARALLEL ENSEMBLES

Our proof-of-principle used a single-core, UQ-style example to demonstrate scaling to millions of jobs. We ran our ensemble on Sequoia, a 1.6 million core machine ranked in the top three on the Graph500 [2] and Top500 [4] lists. Sequoia was originally designed to support UQ jobs involving hundreds of large, concurrent simulations, each using thousands of processors. However, as this problem demonstrates, the UQ program evolved to require thousands to millions of smaller simulation runs.

The application estimates  $\pi$  using independent, randomly chosen points and the probability that a point lies within a circle circumscribed by a square with side  $2r$ :

$$\pi = \lim_{n \rightarrow \infty} \frac{4}{n} \sum_{\substack{i=1 \\ x_i, y_i \in [-r, r]}}^n \begin{cases} 1 & \text{if } \sqrt{x_i^2 + y_i^2} < r \\ 0 & \text{if } \sqrt{x_i^2 + y_i^2} \geq r \end{cases}$$

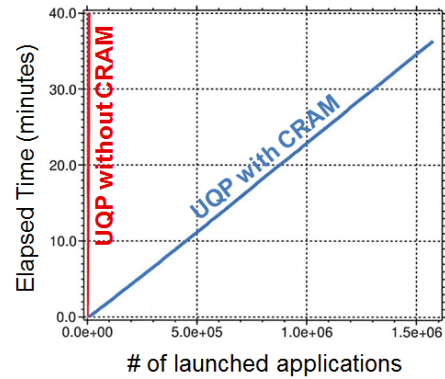
Sequoia was divided into 192 512-node partitions with CRAM packing 8K applications into each run. Figure 3 shows elapsed time as a function of application launches. All 1.6 M runs launched in under 40 minutes.

## 7. FUTURE WORK

Providing scientists with the ability to successfully complete ensemble simulations on modern HPC machines requires the ability to detect and recover from errors. More work is needed to automate the assessment, recovery, and rescheduling of individual simulation runs spawned using the CRAM extension.

## 8. SUMMARY

The UQP was extended to include CRAM, which provides an MPI wrapper library that automatically manages the execution of



**Figure 3: The UQP launched 715 applications per second (versus four) with the new technology.**

multiple simulations in a single MPI job. We demonstrated the ability to launch up to one concurrent UQ simulation run per core by reducing the resource management load on the system.

## Acknowledgments

We thank Don Lucas for his suggestions and feedback on this work and Livermore Computing for providing the resources and support for our dedicated use of Sequoia.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory, Contract DE-AC52-07NA27344. LLNL-ABS-676118.

## References

- [1] CRAM. <https://github.com/scalability-llnl/cram>, 2014.
- [2] D. Bader, P. Kogge, A. Lumsdaine, and R. Murphy. The Graph 500 List. <http://www.graph500.org>.
- [3] J. Gyllenhall, T. Gamblin, R. Bertsch, and R. Musselman. Enabling High Job Throughput for Uncertainty Quantification on BG/Q. *IBM HPC Systems Scientific Computing User Group (Scicomp'14)*, 2014.
- [4] H. Meuer, E. Strohmaier, J. Dongarra, and S. Horst. Top 500 Supercomputer Sites. <http://www.top500.org>.
- [5] J. Tannahill, D. Lucas, D. Domyancic, S. Brandon, and R. Klein. Data Intensive Uncertainty Quantification: Applications to Climate Modeling. In *Proceedings of the 2011 companion to High Performance Computing Networking, Storage and Analysis Companion*. ACM, 2011.