Please fill out the following to the best of your ability. If you have multiple projects and/or codes represented by this case study, note this in the text and then fill out the table at the end using aggregate numbers where appropriate (e.g. total hours used) and maximum values elsewhere (e.g. number of compute cores used per job). If needed, you may include a different table for each major code. For reference, you can see a graph of historical HEP and NERSC usage at http://www.nersc.gov/science/requirements-workshops/case-study-faq/

# Case Study Title:

Principal Investigator:

Worksheet Author (if not PI):

NERSC Repositories:

# 1. Project Description

## 1.1 Overview and Context

Please give a high-level description of your research and the role played by high-end computing and storage. (1-3 paragraphs)

## 1.2 Scientific Objectives for 2017

What are your project’s scientific goals for 2017? Do not limit your answer to the computational aspect of the project. (1-3 paragraphs)

# 2. Computational Strategies (now and in 2017)

## 2.1 Approach

Give a short, high-level description of your computational problem and your strategies for solving it.

## 2.2 Codes and Algorithms

Please briefly describe your codes and the algorithms that characterize them.

# 3. HPC Resources Used Today

## 3.1 Computational Hours

How many hours on conventional cores (no GPUs) will your project use at NERSC in 2012? If you have significant allocations and usage elsewhere please describe it here.

## 3.2 Compute Cores

How many (conventional) compute cores do you typically use for production runs at NERSC today? What is the maximum number of cores that your codes could use for production runs today? If the typical number is less than the maximum, briefly explain why you use fewer cores than you could. Do you typically have multiple jobs running concurrently? If so, how many?

## 3.4 Shared Data

NERSC provides “project directories,” which are permanent, global, shared storage areas for collaboration. Does your project have a NERSC project directory? If so, what is its name?

## 3.5 Archival Data Storage

Archival data is accessible online, but may involve a delay in accessing it (e.g. data stored on HPSS tapes). How much data do you have stored on the NERSC HPSS data archive in 2012?

# 4. HPC Requirements in 2017

## 4.1 Computational Hours Needed

How many conventional compute hours will your project require in CY 2017? (“Conventional” refers to a machine like Hopper, Carver or PDSF. For GPUs, see section 4.7) Include all hours your project will need to reach the scientific goals you listed in 1.2 above. If you expect to receive significant allocations from sources other than NERSC, please list them here.

## 4.2 Number of Compute Cores

How many conventional (e.g., Hopper) compute cores will your code(s) typically use in 2017? What is the maximum that could be used?. (If you expect to be using accelerators or GPUs in 2017, please explain in 4.7 below.) Will you need more than one job running concurrently? If yes, how many?

## 4.3 Data and I/O

How much data will you need to read and write per run in 2017 (including checkpoint/restart data)? Please estimate your I/O bandwidth requirement (bandwidth = data read or written / time to read or write). What percentage of your total runtime are you willing to devote to I/O?

## 4.4 Shared Data

How much NERSC project directory space (globally accessible shared data) will you require in 2017?

## 4.5 Archival Data Storage

Archival data is accessible online, but may involve a delay in accessing it (e.g. data stored on HPSS tapes). How much archival storage will you need in 2017?

## 4.6 Memory Required

For NERSC to scope future systems, we need to know your memory requirements. Please describe your memory requirements for 2017 in terms of the minimum shared memory pool (node) and aggregate memory required for you to run.

## 4.7 Many-Core and/or GPU Architectures

It is expected that systems in the 2017 time frame will contain a significant number of “lightweight” cores and/or hardware accelerators (e.g., GPUs). Are your codes ready for this? If yes, please explain your strategy for exploiting these technologies. If not, what are your plans for dealing with such systems and what do you need from NERSC to help you successfully transition to them?

## 4.8 Software Applications and Tools

What software (applications/libraries/tools/compilers/languages) will you need to run on an HPC system at NERSC in 2017? Please make sure to include I/O-related software.

## 4.9 HPC Services

What NERSC services will you require in 2017? Possibilities include consulting or account support, data analytics and visualization, training, support servers, collaboration tools, web interfaces, federated authentication services, gateways, etc.

## 4.10 Time to Solution and Throughput

Will you have any special needs or constraints in 2017 regarding time to solution, throughput, turnaround, or job scheduling? If so, please elaborate.

## 4.11 Data Intensive Needs

Will you have additional needs we have not considered regarding data? These could be related to workflow, management, transfer, analysis, sharing or access, or visualization.

## 4.13 What Else?

Do you have any other services or facilities you would like NERSC to provide? Do you have present or future concerns you’d like to discuss? What features of an HPC system are most important to you? What the most important and useful services NERSC can provide?

# Requirements Summary Worksheet

Please try to fill out this worksheet, based on your answers above, to be best of your ability prior to the review.

|  |  |  |
| --- | --- | --- |
|  | **Used at NERSC in 2012** | **Needed at NERSC in 2017** |
| Computational Hours |  |  |
| Typical number of cores\* used for production runs |  |  |
| Maximum number of cores\* that can be used for production runs |  |  |
| Data read and written per run | TB | TB |
| Maximum I/O bandwidth | GB/sec | GB/sec |
| Percent of runtime for I/O |  |  |
| Shared filesystem space | TB | TB |
| Archival data | TB | TB |
| Memory per node | GB | GB |
| Aggregate memory | TB | TB |

\* “Conventional cores.” For GPUs and accelerators, please fill out section 4.7.