**• 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, include a different table for each major code.   
• For reference, review the graph of historical usage at   
http://www.nersc.gov/science/requirements-workshops/case-study-faq/**

# Case Study Title: (enter a title here)

**Principal Investigator:**

**Worksheet Author(s) (if not PI):**

**NERSC Repositories**:

# Project Description

## Overview and Context

**Please give a brief, high-level description of your research and the role that High Performance Computing (HPC) and storage play. (1-3 paragraphs)**

## 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)**

# Computational Strategies (now and in 2017)

## Approach

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

## Codes and Algorithms

**Please briefly describe the codes you use and algorithms that characterize them (1-2 sentences per).**

# HPC Resources Used Today

## Computational Hours

**How many hours on conventional cores (not GPUs) will your project(s) use at NERSC in 2013?   
If you have significant allocations and usage at other sites please describe it here.**

## Parallelism

**How many (conventional) compute cores do you typically use for production runs at NERSC today? (You can give a range.)**

**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 compute in High Throughput Computing mode, where you have multiple tasks running concurrently in a single job? If so, how many concurrent tasks are typical?**

**Which is more important to your project, strong scaling or weak scaling? Why? (Strong: you have a problem of a given size and you'd like to use parallel computing to solve it faster. Weak: you have a problem of a given size and you'd like to use parallel computing to solve a bigger problem in the same time.)**

## Scratch Data

**How much temporary disk space (space that can be purged) do you need to perform your runs?**

## 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? What is the primary reason you have this space?**

## Archival Data Storage

**How much data do you have stored on the NERSC HPSS data archive in 2013?**

# HPC Requirements in 2017

## Computational Hours Needed

**How many conventional compute hours will your project require in CY 2017? (“Conventional” refers to a machine like Edison, Hopper, Carver or PDSF.)** **Include all hours your project will need to reach the scientific goals you listed in 2.2 above.**

**If you expect to receive significant allocations from sources other than NERSC, please list them here.**   
  
  
  
**If you expect to need more compute hours in 2017 than you used at NERSC in 2013, what is the primary factor driving the need for more hours?**

## Parallelism

**How many conventional compute cores will your code(s) typically use in 2017?**

**What is the maximum that could be used?**

**Will you need more than one job running concurrently? If yes, how many?**

**If you compute in High Throughput Computing mode, where you have multiple tasks running concurrently in a single job, how many concurrent tasks will you typically have in 2017?**

## 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?**

## Scratch Data

**How much temporary disk space (space that can be purged) will you need for your runs in 2017?**   
  
**What is causing the growth in scratch space for your research?**

## Shared Data

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

**What is causing the growth in project directory space for your research?**

## Archival Data Storage

**How much data will you need to store on the NERSC HPSS data archive in 2017?**

**What is causing the growth in archival storage for your research?**

## Memory Required

**For NERSC to plan for future systems, we need to know your memory requirements. How much memory will your codes require per node? How much aggregate memory will be required?**

## Emerging Technologies and Programming Models

**Questions 4.1 – 4.2 and 5.1 – 5.2 (above) referred to systems with conventional processor technology such as on Hopper or Edison today. It is expected that systems in 2017 may contain processors using more energy-efficient technologies, with many “lightweight” and possibly heterogeneous cores along with specialized hardware (i.e., GPUs), all requiring significantly greater parallelism & threading.**

**Are your codes ready for this now? If yes, please explain your strategy for exploiting these technologies.**   
  
  
**If your codes are not ready now, what plans do you have for transitioning to such technologies and what help might you need from NERSC for this?**

## Software Applications and Tools

**What HPC software (applications / libraries / tools / compilers / languages / etc) will you need from NERSC in 2017? Be sure to include analytics applications and I/O software**.

## 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**.

## 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**.

## 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**.

**Are you satisfied with NERSC's HPSS system? If not, what improvements would you like to see to improve archival storage?**

**Do you already have a data management plan for your project and does it include archival storage?**

## What Else?

**Are there 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 2013** | **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 |  |  |
| Scratch File System space | TB | TB |
| 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.

# Additional Storage and I/O Questions

These questions are optional but your answers will provide additional useful data for NERSC. If you don't know the answer to any of these leave them blank.

**For Scratch data (like Question 5.4)**:

• **Is your I/O more serial or parallel?**

• **Is your I/O more single-node or multiple-node?**

• **Is your I/O more shared (N-to-1) or distributed (N-to-N)?**

•  **Is your I/O more small-file or large-file?**