Dark software: Addressing the productivity challenges of extreme-scale science on-ramps and off-ramps

Ian Foster
Argonne National Laboratory and University of Chicago
foster@anl.gov

1. INTRODUCTION

Several decades of sustained progress in computational science and high-performance computing have ushered in a world in which extreme-scale computers are used to perform detailed simulations of great scientific importance. Software is at the heart of this transformation, encoding the algorithms that have not only increased performance in their own right but also allowed researchers to make effective use of increasingly complex hardware.

Yet scientific discovery is the result not of individual simulations but of complex end-to-end research processes. These processes frequently involve, for example, the ingest and analysis of simulation, experimental, and observational data; the invocation of simulations within larger design optimization and uncertainty quantification activities; validation through comparison of experimental and simulation data; and the dissemination of output data to communities for analysis. The software created and used by extreme scale scientists must address all such tasks—and the productivity of those scientists will be determined by the sum of the times taken for all tasks.

But while the software used to perform simulations on extreme-scale computers is often carefully engineered, the software used for other tasks in the end-to-end workflow is typically not. Indeed, it often involves ad-hoc scripts, one-off programs, and other non-scalable, non-shareable, and error-prone components. Scientists may not even think of this code as software, even though it consumes much time and energy. Thus, by analogy with dark matter in physics—the stuff that, while invisible, is hypothesized to account for a large part of the total mass in the universe [5]—I term this code **dark software**. (Bryan Heidorn uses the term dark data [4] to refer to a similar concept in the world of data.) I believe that dark software accounts for a substantial fraction of the total "mass" of an extreme-scale project as measured in lines of code developed by individual scientists—and the time spent with that code during a project's lifetime. I suggest that a program to improve software productivity for extreme scale science must address dark software if it is to be impactful. I discuss where dark software arises in research and propose a research program to address associated challenges.

2. END-TO-END DISCOVERY PROCESS

Figure 1 provides a simplistic, but I think useful, model of the scientific discovery process. In this model, a research project that involves extreme-scale computation first undertakes preparatory or *on-ramp* tasks (see Figure 1). For various reasons, these tasks are frequently more challenging and time-consuming in extreme-scale science. For example, large quantities of simulation, experimental, and observational data may need to be located, ingested, and analyzed; process models developed; new algorithms and codes implemented; submodels validated via comparison of experimental and simulation data; and simulations invoked within larger design optimization and uncertainty quantification cycles.

Once large-scale ultrascale computations have been performed, equally involved post-simulation or *off-ramp* tasks must frequently be undertaken. These tasks can range from validation of model output to dissemination of output data and derived products to large communities—who may engage in further analysis, generating further derived products. Any or all of the steps in Figure 1 may be invoked repeatedly during the discovery process.

3. DARK SOFTWARE CHALLENGES

Analysis of the discovery process in many disciplines reveals that on-ramp and off-ramp tasks often dominate elapsed time—and that, increasingly, these tasks are impeded by the diversity, complexity, and scale of their input and output data. For example, a single two-week run of the FLASH astrophysics code produced 74 million files [1]; off-ramp analysis tasks took two years. In climate science, the recent CMIP process ran for several years after simulation data was produced, with ultimately 700+ publications resulting. In both cases, I suspect (but cannot

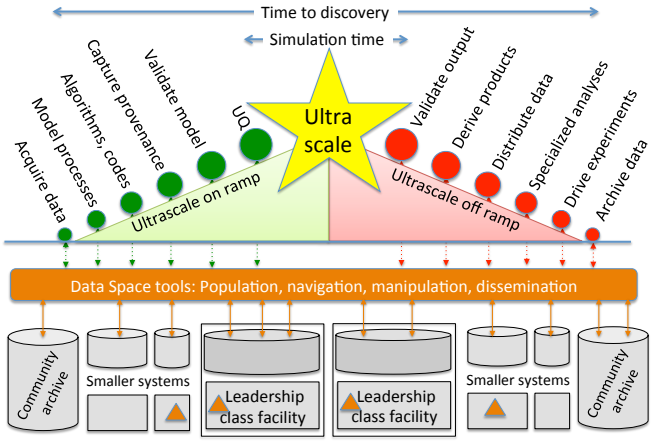

Figure 1: The extreme-scale computing discovery process

prove) that large quantities of dark software were produced in support of the ultimate science goals: software that mostly sits unused today, if it has not been lost altogether.

The following example illustrates other problems encountered in such scenarios: *A sustainable-energy analysis involves 200 ensemble runs of coupled global and regional energy, climate, and hydrology models. Each run consumes both historical and simulation data and also produces tens of thousands of output files—which, due to memory and I/O constraints (and error rates and resilience measures adopted during the run) may, after in-situ analysis and reduction, vary in composition relative to those from other runs. Subsequent ensemble-wide analysis must be able to identify and deal with such differences across the ~10M files generated by the study. In order to permit reuse, the processes used to create those files must also be recorded, so that others can assess data quality and suitability for other purposes.*

No project today has anything but ad-hoc solutions to the problems of capturing, mapping, analyzing, and managing such large and diffuse collections of information. Researchers may use diverse file formats, file naming strategies, and databases to record some information; much more often remains in their heads where it is neither long-term accessible nor easily sharable with others. Thus, tasks such as the following tasks become exceedingly difficult, and if encoded in software form (e.g., as scripts) are not easily re-used: comparing data with output from a run performed last year or from related applications; determining what parameter values and code versions were used for a run; determining which computations might need to be re-run to incorporate new experimental results; or identifying and locating special features across a range of runs. Scientific productivity is constrained by the lack of more structured support for such tasks.

4. ILLUMINATING DARK SOFTWARE

I attribute the large impact of dark software on scientific productivity to: (a) the complexity of the on-ramp and off-ramp tasks associated with end-to-end discovery; (b) an absence of general methods and tools for conducting those tasks, which leads to individual researchers repeatedly (re)inventing software to meet personal needs; and (c) a general lack of understanding around the software productivity challenges associated with those tasks.

I believe that a research program aimed at reducing the high costs associated with dark software has the potential for rapid progress due to the relatively primitive state of the art in the area. Such a program should aim to deliver transformative innovations rather than incremental progress—innovations that can be achieved by pursuing radically new conceptualizations of end-to-end discovery. We can ask, in particular, how discovery processes should be reconsidered in an era of massively computational and collaborative research. Software encodes process; processes may need to evolve along with technology.

The following is an idea that may be worthy of investigation. On-ramp and off-ramp tasks often proceed slowly due to difficulties associated with the tremendous diversity, volume, and distribution of input and output data—which are often scattered across multiple directories, file systems, and sites, in diverse formats, and with only ad hoc (or often no) organizing principles, provenance information, and other metadata to facilitate navigation and reuse. I hypothesize that we can accelerate such tasks by providing overlay structures that permit researchers to identify, navigate, manage, and manipulate the entirety of all relevant input and output data, and associated metadata and computational codes. Researchers can then always obtain up-to-date information on the location, organization, relationships, provenance, and other characteristics of all data and computations viewed as relevant to their project—and easily augment, manage, manipulate, and disseminate that data. Like a dataspace [3] but encompassing the entirety of a research project's data, software, and computations, such a construct may slash on-ramp and off-ramp times and thus the total cost of scientific discovery.

Delivering complex and evolving software to many people is a major software productivity challenge. Thus, another research area deserving greater effort concerns the efficient (in terms of both cost and researcher time) delivery of such software. Globus Online [2] has shown the value of software as a service (SaaS) methods for this purposes.

In these and other areas, a research program can usefully explore the numerous computer science challenges that must be overcome to provide such capabilities—and examine whether and how such capabilities can enable quantifiable acceleration of discovery.

ACKNOWLEDGMENTS

I thank Rachana Ananthakrishnan, Dan Katz, Kerstin Kleese-Van Dam, Galen Shipman, and Dean Williams for discussions of some of these ideas. This work was supported in part by grant DOE DE-AC02-06CH11357.

REFERENCES

1. Fisher, R.T., Kadanoff, L.P., Lamb, D.Q., Dubey, A., Plewa, T., Calder, A., Cattaneo, F., Constantin, P., Foster, I., Papka, M.E., Abarzhi, S.I., Asida, S.M., Rich, P.M., Glendening, C.C., Antypas, K., Sheeler, D.J., Reid, L.B., Gallagher, B. and Needham, S.G. Terascale Turbulence Computation on BG/L Using the FLASH3 Application Framework *IBM Journal of Research and Development*, 52(1/2):127-136, 2008.
2. Foster, I. Globus Online: Accelerating and democratizing science through cloud-based services. *IEEE Internet Computing*(May/June):70-73, 2011.
3. Franklin, M., Halevy, A. and Maier, D. From databases to dataspaces: a new abstraction for information management. *SIGMOD Rec.*, 34(4):27-33, 2005.
4. Heidorn, P.B. Shedding Light on the Dark Data in the Long Tail of Science. *Library Trends*, 57(2):280-299, 2008.
5. Trimble, V. Existence and nature of dark matter in the universe. *Annual review of astronomy and astrophysics*, 25(1):425-472, 1987.