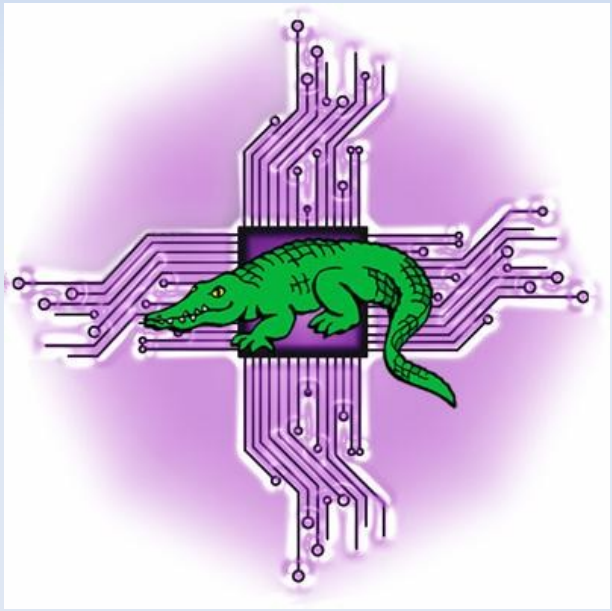
Teaching Scientists High Performance Computing

Presentation to the University of Otago Systems Research Group



February 22nd, 2013

http://levlafayette.com

About Your Presenter

HPC Systems Administrator at the Victorian Partnership for Advanced Computing in Australia for the last six years or so. Also .... Website Administration, Project Manager, Quality Assurance System Coordinator, and - importantly for this context - HPC and Scientific Applications Trainer.

Prior to that more experience in computer systems training; including community organisations, developing world (Ministry of Foreign Affairs, East Timor, UNDP publication Neon-Kompudator, 2003), and politicians and electorate officers for the Parliament of Victoria. Also, plenty of presentations (introductory and otherwise) to Linux Users of Victoria.

Not a graduate in computer science (so I don't want to teaching grandma to suck eggs)! But a "hobby interest" from the very early 80s led to a couple of undergraduate electives in discrete mathematics, introductory programming, and a little bit of computer architecture. Ended up graduating in politics, philosophy, and sociology. Have also completed a MBA (Tech Mngmnt), along with postgraduate studies in adult and higher education.

About VPAC and Teaching HPC

Registered research agency and non-profit organisation established in 2000 by a number of the universities in Victoria to provide HPC and advanced computing support. Smaller industry sector. Has had two computers in the Top 500 in that time. Latest system, Trifid, operates at a peak of 45.9 TFlops.

Several hundred users across the state. Three main courses conducted for researchers; introductory, intermediate, and advanced HPC using Linux. Each course runs for a day with a c65 page manual. First course provides an introduction to the Linux command line (c30% have next to no experience), environment modules, PBS and job submission, running jobs. Second course provides more detailed commands, regular expressions, scripting, job arrays and dependencies, program compilation. Third course assumes knowledge of C/C++, Fortran, introduces MPI programming concepts (communication between multiple cores, ordering of return values)

Invariably excellent responses from course surveys, also included as part of our ISO QA standard.

In the past we have also run specialist courses on particular applications or groups of applications (especially in the life sciences). Have made attempts to approach universities for the establishment of a Graduate Certificate (or equivalent) in High Performance Computing.

More at: http://vpac.org

What's The Issue?

Computer scientists are extremely aware of the benefits of high performance computing and the necessity for both unicore applications to be ported to a multicore system and for scientists in other disciplines (including social scientists) to be trained in their use (c.f., Multicore World).

Scientists outside of computer science do not necessarily understand the benefits of multicore; and why should they? A core motivation can be enticed (c.f., Greg Wilson of Software Carpentry) by explaining that use of HPC reduces the average time to produce a research paper from 8 to 6 months.

Nota bene: Software Carpentry (deliberately distinct from "Software Engineering") engages in a very similar project to VPAC in terms of training. It runs for a short period (two days) and provides the basics for a postgraduate researcher to use a computer system for scientific computer in a manner that enhances their productivity. Specifically, a little bit of command-line Linux, iPython programming, and SQL.

If HPC applications and MPI programming is so important however, where do you put it? The material - from a computer science perspective - is typically considered too simple for postgraduates and there's not enough "room" at an undergraduate level. At the moment it is assumed that the scientists will pick up the necessary knowledge along the way; this is not a viable proposition!

Four Options

The VPAC/Software Carpentry model. Short courses that provide a minimum level of understanding to the core concepts. Just enough to make use of the system, trying to increase usage of clusters (10 users = 50% of current usage, high percentage of unicore jobs, Wilson's unpublished paper "High Performance Computing Considered Harmful"). This model gives a larger number of scientists the knowledge to submit jobs and some conceptual level of HPC.

A significantly more detailed Graduate Certificate in High Performance Computing (nota bene: not meant as a computer science degree per se., but rather for graduates in disciplines other than computer science). This model would give a smaller number much greater knowledge of HPC and parallel programming; they would become the computational leaders of a scientific research team.

The Murdoch model; a single semester, multidisciplinary unit which provides a broad and sufficient base to the "faculty level" studies, as well as an introduction to "university life". Introducing HPC and parallel programming at this stage could only be done in the most minimal manner (e.g., like the VPAC/Software Carpentry model), but would also provide the possibility of early and inspiring interest.

Attempts to simplify job submission tools so the researcher can "just get their results" (e.g., xPBS, Hadoop, Grisu, Yabi, Galaxy). Would still require minimal conceptual and practical training, but may have a larger uptake.

Getting Back to the Context

Adult education is significantly different to childhood education and, within that, the needs of young undergraduates differs to that of older graduates; a distinction between training, teaching, and researching. Postgraduate and post-doctoral researchers are smart; they will "get it" when it's explained to them.

A core concept in adult and higher education is the idea of lifelong learning, with competing interests and motivations between proximal and distal goals. The individuals involved are part of the construction of knowledge; consider presage (requirements), process (their implementation), and product (subject matter). Consider the distinction between, for example, a conceptual knowledge of high performance computing and a practical knowledge of submitting HPC jobs etc.

The preferred result is a level of learner autonomy and self-efficacy, generated from regulation and direction. This is achieved from a number of intrinsic and extrinsic motivational factors (e.g., desire to publish, sense of mastery, accreditation, financial motivations, even work avoidance "lazy like a fox").

Implementation

Be careful of prescriptive notions of "learning styles", learning styles, specifically Fleming's NLP-derived VAK/VARK model. It is popular to assume that learners are either visual, auditory or tactile, the evidence again does not support these assumptions, either from a wide-ranging review of existing scientific literature and of the special public interest research investigation conducted by the Association for Psychological Science. Introduce different "learning styles" as part of a preference package, not as a solution.

An alternative model to consider is an effective elaboration of Vygotsky's notion of "proximal knowledge" into "apprenticeships" of communities of knowledge (c.f., Lakatos). Individual learners are assisted by models provided by recognised experts and "community leaders" in a research discipline. This can improve interest - not just motivation - in the subject area.

Learning, interest, motivation, and outcomes must "flow" (c.f., Csikszentimihalyi) into a development of a personal epistemology, resulting in a high level of immersion of the practioner. The state of flow is the an interested mastery of the subject matter.

Working Backwards from the Ideal

A situation where postgraduate and postdoctoral researchers make ample use of multicore scientific applications to produce cutting edge science in New Zealand prior to the rest of the world with the resultant economic benefits.

The postgraduate and postdoctoral researchers have an interest and motivation to engage in multicore scientific computing. They have an interest because they are part of a community of knowledge generators. They have the motivation because they are generating successes in this field.

The postgraduate and postdoctoral researchers have a deep conceptual understanding of the field because of the introduction of the core concepts at an early stage of their academic career, and have subsequently either acquired sufficient understanding to make practical use of the tools or have engage in a deeper, graduate certificate level, study of the subject to supplement their main research.

The four options previously presented, *in combination*, can significantly help to resolve the problem of HPC knowledge and usage.