ARCHER
Software Carpentry workshop

Mario Antonioletti, Alistair Grant, Mike Jackson, Arno Proeme
ARCHER CSE Team
mario@epcc.ed.ac.uk
agrant3@epcc.ed.ac.uk
michaelj@epcc.ed.ac.uk
aproeme@epcc.ed.ac.uk

Reusing this material

http://www.training.prace-ri.eu

ARCHER in a nutshell

- UK National Supercomputing Service
  - Replacement for HECToR (they overlapped by ~4 months)
- Cray XC30 Hardware
  - Nodes based on 2×Intel Ivy Bridge 12-core processors
  - 64GB (or 128GB) memory per node
  - 3008 nodes in total (72162 cores)
  - Linked by Cray Aries interconnect (dragonfly topology)
- Cray Application Development Environment
  - Cray, Intel, GNU Compilers
  - Cray Parallel Libraries (MPI, SHMEM, PGAS)
  - DDT Debugger, Cray Performance Analysis Tools

Performance

- HECToR #50 in top 500 with 830 TFlop/s
- ARCHER
  - Designed to provide 3-4 times scientific throughput of HECToR
  - #25 in June 2014 top 500 list with 1.65 PFlop/s
- Extract the best performance possible from the hardware

Optimization is risky

- “More computing sins are committed in the name of efficiency (without necessarily achieving it) than for any other single reason ..
- …Including blind stupidity” (Wulf)
- "We should forget about small efficiencies, say about 97% of the time:
- …Premature optimization is the root of all evil.“ (Knuth 1974)

Optimization

- What is preferable?
  - An optimized code that produces incorrect results?
  - A non-optimized code that produces correct results?
  - Which is the most efficient use of resources?
- How do we prevent ourselves introducing bugs when we’re optimizing and parallelizing?
- Isn’t our time more valuable than a computer’s time?
- Hardware life < software life < our life

Three rules of optimization

- Don't – optimized code is not readable or maintainable code
- Don't....yet – designing code is not the same as optimizing code
- Profile first – anything else is just a guess!

archer | software carpentry | epcc | The University of Edinburgh

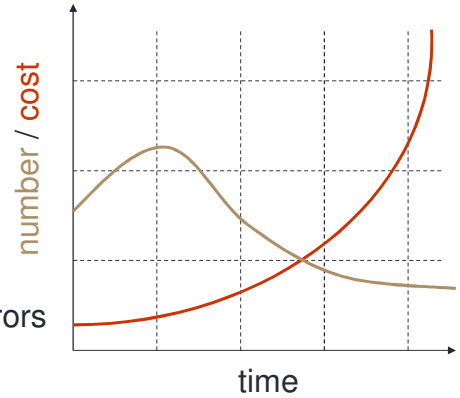
Optimize ourselves

50%-80% of errors
in
15%-20% of modules
(Davis 1995 quoting Endres 1975, Weinberg 1992)

50% of modules are error free
(Boehm and Basili 2001)

design errors >> # coding errors
(Boehm et al. 1975)

Technical debt
1-10-100 rule



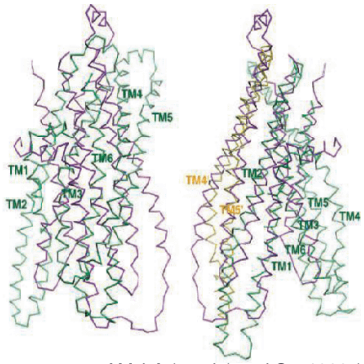
archer | software carpentry | epcc | The University of Edinburgh

Programming language or programmer?

- "The number of lines of code a programmer can write in a fixed period of time is the same independent of the language used" – Corbato's Law
- "Regardless of whether one is dealing with assembly language or compiler language, the number of debugged lines of source code per day is about the same!" (Corbato 1969)
- "performance variability that derives from differences among programmers of the same language … is on average as large or larger than the variability found among the different languages." (Prechelt 2000)

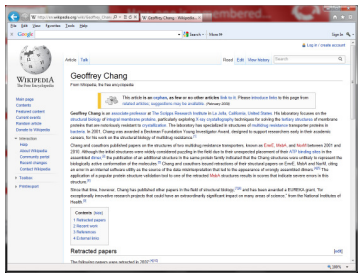
archer | software carpentry | epcc | The University of Edinburgh

What about correctness?



The structures of MsbA (purple) and Sav1866 (green) overlap little (*left*) until MsbA is inverted (*right*)
Courtesy of Dawson, R.J.P and Locher, K.P, Nature 443, 180, 2006

"Chang was horrified to discover that a homemade data-analysis program had flipped two columns of data"



Miller, G. "A Scientist's Nightmare: Software Problem Leads to Five Retractions", Science 314(5807), pp1856-1857. DOI: 10.1126/science.314.5807.1856

archer | software carpentry | epcc | The University of Edinburgh

Behind every great piece of science…

```
nd then each downstream SNP of interest
r(my $y = $x+1; $y < scalar @pos; $y++)
#if SNPs within our chosen distance (500kb) and both present in the haplotypes file
if((!($trait[$x] eq $trait[$y])) && (abs($pos[$x] - $pos[$y]) <= 500000) && (exists($legArrayPos{$pos[$x]})) && (exists($legArrayPos
{
    my $snp1ArrayPos = "";
    my $snp2ArrayPos = "";
    my $snp1All = "";
    my $snp2All = "";

    #create output file for this SNP pair
    my $filename = "ConditionedResults2/$chr[$x].$pos[$x]-$pos[$y].EHH.GBR.2.txt";
    print "$filename\n";
    unless (-e $filename) {
        open(OUT, ">$filename");

        ####################CHANGE THESE IF NOT FOCUSING ON SECOND SNP#########################
        my $start = $pos[$y]-500000;
        if ($start < 1) {
            $start = 1;
        }
        my $end = $pos[$y]+500000;
        if ($end > $chrLengths{$chr[$x]}) {
            $end = $chrLengths{$chr[$x]};
```

Courtesy of Carole Goble

archer | software carpentry | epcc | The University of Edinburgh

Important stuff

Images courtesy of (clockwise from top left) one village initiative, Bobbymond, Bajanda, Anton Novojilov

archer | software carpentry | epcc | The University of Edinburgh

How to ask for help

- Flag down a helper

- Stick up your sticky note
  - Red – I'm stuck, help!
  - Green – I'm fine