



DE LA RECHERCHE À L'INDUSTRIE

How will legacy codes survive the next generations of systems?

2019/05/16

Guillaume Colin de Verdière

- Part 1
 - Talk 1 : Setting the landscape

- Part 2
 - Discussions with the audience

- Definition: a legacy code (LC) is a code
 - That is older than 10 years
 - Has seen at least two system generations
 - Sometimes more than 6 !
 - That is vital to the entity
 - That needs to be ported to newer systems



- Large Research Entities (LRE)
 - E.g. CEA
 - Some codes are older than 20 years

- ISVs
 - Some code roots are coming from LREs

- SMEs?
 - Too often no

- “Old” implementations
 - Older versions of standards
 - F77
- “Old” data structures
 - “à la CRAY”
- Parallelism somewhat limited
 - MPI only, seldom GPU versions or OpenMP
- Relies on “old” third party libraries
 - E.g: IMSL to manipulate strings in Fortran

- Original team not available anymore
 - Need to “educate” new generations of developers

- Large source code
 - >> 1M SLOC
 - Intimate knowledge is lost

- Funding to maintain the code

- Funding to port to the next generations of machines

- Establish a quantitative measure of LC potential of evolution
 - “Code Viscosity”
 - Survey to fill
 - Pointers to contributors needed ([x.y@z.t](#))
 - Assumption: Exascale+ systems will be accelerated

- Goal: demonstrate to the commission
 - The importance of LC
 - The richness of the European portfolio
 - The associated costs and needs of funding.

$$V = \frac{A * L}{TC * P}$$

- A = age of code
- L = SLOC
- TC = TEAM confidence to develop accelerated codes
 - [0.001, 1]
- P = % of acceleration
 - [0.001, 1]

Discussions

Guillaume.colin-de-verdiere@cea.fr

- Importance of legacy codes in your institution?
- Are you ready / staffed / funded / ... to do the port?
- Is a complete rewrite envisioned / feasible / ...?
- Evolution of programming languages foreseen?
- Which tools are you lacking to move forward?
- What would be your key messages to the commission?



THANKS