

Survey

This survey is carried out within the scope of the article in preparation "**Programming Languages for Data-Intensive HPC Applications: a Systematic Mapping Study**", initiated by **Vasco Amaral** (Univ. Nova de Lisboa) and co-authored by the 19 contributors to the SLR/SMS study during the last 3 years.

For complementation and validation of the literature review results, we would like to compare with the honest estimations of **experts** in data-intensive high-performance computing (that is, **you**). Please help us in collecting a sufficiently large and broad statistical basis for this validation by answering this survey form now at the Las Palmas meeting. It only takes 2-3 minutes, and the collected data will enable us to complete the article, producing a tangible outcome of *cHiPSet*.

Please hand in the paper anonymously to **Christoph Kessler** or **Peter Kilpatrick** during the Las Palmas meeting. Many thanks in advance!

1. Were you involved in the SMS (Vasco's literature review program)?

☒ Yes ☐ No

2. How long have you been working in High Performance Computing?

☐ Not at all ☐ < 2 years ☐ 2 to 5 years ☐ 5 to 10 years ☒ > 10 years

3. In what areas of science or engineering have you worked?

(e.g., computer science, bioinformatics, material science, telecommunications ...)

High Energy Particle Physics, Computer Science

4. Do your High Performance Computing related activities consist primarily of

☒ *Developing* programming support tools, or ☐ *Using* existing programming tools?

5. How do you rate your level of technical knowledge about languages/frameworks for HPC?

☐ Very Poor ☐ Poor ☒ Neutral ☐ Good ☐ Excellent

6. Which programming languages do you *use* for High Performance Computing?

Java, C++, C

7. What are, in your view, the *key advantages* of these languages (in relation to the alternatives you know)? (this may include language properties, performance, programmability, etc.)

Due to its flexibility, it is adequate to non-repeatable/reusable software solutions and for exploration of alternatives when the technology is not well understood.

8. What *actually* made you use these languages? (if not already covered in 7.)

9. Which other programming frameworks (e.g., library-based) and tools do you *use* for HPC?

10. Which other HPC programming languages / frameworks / tools do you know about (but do not use)?

CUDA, Spark SQL and Scala