Dissonance Reduction Tooling for HPC Pedagogy

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

3rd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

4th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

5th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

6th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

***Abstract*—We describe a general work-flow which scales intuitively to high-performance computing clusters for disparate domains of scientific computation. We demonstrate our methodology with a radial distribution function calculation in three languages; FORTRAN, C++ and Python. We show that incorporating appropriate tooling, namely Git, CMake, TORQUE/SLURM, and nix-language expressions for packaging into the pedagogical practice allows for high-performance, platform-independent, reproducible scientific software. For domain specific algorithms, we show that there is a language-independent pedagogical methodology which may be leveraged to ensure best practices for the scientific HPC (high-performance computing) community with minimal cognitive dissonance for practitioners and students.**

***Index Terms*—pedagogy, best-practices, tooling, methodology, reproducible-research**