Tool Issues at Scale

Working Group Report
CScADS '12

29 June 2012

Problem: Dimensions of Scale

- Application space/concurrency
    - size of input data
    - size of output data
    - # of processes/cores/GPUs etc.
    - execution time (also a factor in other dimensions)
    - environmental and resource issues

- Application code
    - binary size
    - mixed-language or mixed-paradigm
    - # of libraries
    - # of functions / symbols / templated classes
    - # of instructions / floating-point operations

- Tool space/concurrency
    - analysis workload
    - profiling data/output
    - presentation/visualization

Approaches to Tool Development

- Traditional paradigm
  - Develop for and test at small scale
  - Then scale up to large scale

- New paradigm
  - Develop for and test at large scale
  - Keep all scaling dimensions in mind

Techniques

- Processing collection data
  - "in-situ"/online filtering and analysis of profiling data
    - temperature monitoring example
    - feedback to tool for focused collection
    - global coordination points - refine collection process
  - look for patterns (local vs. global)
  - equivalence classes for profiling output

- Continuous monitoring
  - feedback to tool front-end or auto-tuning system
  - selecting interesting areas / areas of stress

- Environmental infrastructure integration
  - collaborate with vendors (example - CDTI)

Possible Solutions

- Frameworks
  - can provide deployment environment at scale
  - integration from other environmental resources (monitoring)
  - target scenarios or scale dimension metrics

- Virtualization/emulation for scaling tests
  - must have models for all affecting variables (FS contention, etc.)
  - can be hard to know without at-scale testing
  - chicken-and-egg problem

- Running on big-iron
  - phased approach to getting to this point