Problem: Dimensions of Scale

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

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

• Tool space/concurrency
  o analysis workload
  o profiling data/output
  o presentation/visualization
Approaches to Tool Development

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

• New paradigm
  o Develop for and test at large scale
  o Keep all scaling dimensions in mind
Techniques

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

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

• Environmental infrastructure integration
  o collaborate with vendors (example - CDTI)
Possible Solutions

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

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

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