Things that have worked

- Performance risk analysis.
- Source management.
- Agile practices:
  - Pair programming:
    - For training
    - Highly technical programming.
  - Test-first.
  - Tight customer interaction.
  - Incremental release.
- Domain specific frameworks/libraries (designed by SW engineers:
  - Abstracts away parallel machine.

Things that worked 2

- Generalization to high level constructs that abstract away the platform specifics.
- Code reviews.

Things we need

- Design for performance from the beginning. Can’t retrofit performance.
- Performance has to come before functionality.
- Ideas for SW architecture.
- Expert testimony that standard SW lifecycle models are not adequate for HPC.
- Tools that we need:
  - Light-weight documentation.
  - Correctness testing. Help in design for testability.
  - Designed for scientists.

Things we need

- Follow the design process that you promote:
  - Gather requirements, etc.

Things that we don’t need

- Grid:
  - Big design up front.
  - Doesn’t work.
  - What do you want to do? Wait 10 years and the grid will be there.
- Full-blown RUP SPMP.
  - Takes a lot of time.
  - Gets stale before it is done.
  - Projects with high technical risk can be managed by changing requirements.
- Or: Big design up front before we mitigate the technical risks.