Date: 1/29/2020

To: James Ahrens

cc: David Pugmire, David Rogers, Hank Childs, Berk Geveci

From: Kenneth Moreland, WBS 2.3.4.13 / ECP/VTK-m

Milestone Deliverable - STDA05-51

Milestone Due Date: January 31, 2020

Milestone Completion Date: January 22, 2020

Description of Milestone:
The STDA05-51 milestone comprises the following tasks.

- Visualization for WDM application: Consult with WDM project to determine codes and analysis requirements.
- Advanced Flow Techniques (ORNL): Engage with an ECP applications to determine needs for flow analysis and visualization.
- Deployment in VisIt: Identify priority operations in VisIt that can improved with VTK-m support.
- Data files for tests: Design of data file collection framework.
- Volume Fractions: Construct initial design of module that uses volume fraction information to construct material boundaries, as well as initial design of representation of volume fractions in VTK-m. Confirm that module will meet stakeholder needs.
- Higher order mesh support: Create initial design of VTK-m higher order data model.
- ArrayHandle thread safety: Draft initial design for ArrayHandle thread safety.
- Perf Regression Testing Infrastructure: Draft initial design of the performance regression testing infrastructure.
- Structured Grid Contouring optimization: Determine which structured grid algorithm(s) such as Flying Edges that is suitable for implementation into VTK-m.
- Resampling/Statistics/Random Numbers: Evaluate requirements for these capabilities, in collaboration with ALPINE team, including parallel random number generator (RNG).
- Test image comparison: Design of test image comparison framework.
- VTK-m Maintenance (Kitware): Level of effort.
- VTK-m Maintenance (SNL): Level of effort.

Completion Proof of the Milestone:
Each task represents development in the VTK-m library. The lifespan of a development meta-task (spanning multiple P6 Activities) involves, where applicable, a design, implementation, and documentation. The design is captured using the GitLab issue tracker associated with the VTK-m code repository. The development is captured in GitLab merge requests, which are an artifact of

the development workflow within the VTK-m software. Documentation usually includes contributions to the *VTK-m User's Guide*.

The following list provides evidence for each task in the form of GitLab issues, GitLab merge requests, and documentation.

- Visualization for WDM application
  - Design
    - Visualization for WDM application (#469)
- Advanced Flow Techniques (ORNL)
  - Design
    - Particle Advection Enhancements (#472)
  - Merge Requests
    - Add device options to particle advection example. (!1885)
    - Fix for smallStep in integrator. (!1879)
    - optimization for MaskSelect when mask is initialized with all 1s. (!1891)
    - Particle advection storage using array of structs. (!1854)
- Deployment in VisIt
  - Design
    - VTK-m support in VisIt (#468)
- Data files for tests
  - Design
    - Update Unit Test Infrastructure to Use Data Files Stored in Git LFS (#462)
- Volume Fractions
  - Design
    - Report for "Volume Fractions:1 Design" (#465)
    - Reconstruction of Material Interfaces (#466)
    - Sparse Representation of Volume Fractions (#467)
- Higher order mesh support
  - Design
    - Report for "Higher order mesh support:1 Design" (#470)
- ArrayHandle thread safety
  - Design
    - ArrayHandleBasicImpl should not toy with execution environment pointers (#384)
    - Implement thread safety in ArrayHandle (#435)
  - Merge Requests
    - Access ArrayHandle internals in a critical section (!1905)
- Perf Regression Testing Infrastructure
  - Design
    - Benchmarking and Performance Regression Testing (#448)
- Structured Grid Contouring optimization
  - Design
    - Optimizing Structured Grid Contouring (#449)
- Resampling/Statistics/Random Numbers

    - Design
        - Random Number Generator (#461)
- Test image comparison
    - Design
        - Implement Test Image Comparison (#437)
- VTK-m Maintenance (Kitware)
    - Merge Requests
        - Release 1.5.0 (!1893)
        - Refactor CellSetExplicit to remove NumIndices. (!1874)
        - CellSetConnectivity now inherits from FilterDataSet (!1870)
        - Improve the performance of the Image and Graph Connectivity algorithms (!1873)
        - Workaround issue with older C++. (!1881)
        - Fix unused argument warning. (!1883)
        - VTK-m filters now have a consistent set of header guards (!1886)
        - Update vtkc to lcl (!1884)
        - MSVC 2019 needs a different integer sequence impl, too. (!1887)
        - ExecutionSignatures are now optional for simple worklets (!1882)
        - Fix lcl dashboard issues (!1888)
        - Add an ImageMedian filter (!1890)
        - Initial documentation on how to cut a VTK-m release (!1894)
        - Fix dashboard issues (!1896)
        - Fix UnitTestError (!1897)
        - VTK-m now excludes large files from CMake Unity builds (!1907)
        - Improve Threshold runtime and compile performance (!1909)
        - improve moments algorithm performance (!1895)
        - Remove unused function interface components (!1910)
        - VTK-m use a jobs pool for compiling large compilation units. (!1911)
        - Make sure vtkm_setup_job_pool() doesn't output any status (!1912)
        - correct setting up job pool for tests (!1914)
        - Another round of simplifications to FunctionInterface (!1915)
        - Correct warnings on OSX caused by differing symbol visibility settings (!1913)
        - Prefixed CACHE_LINE_SIZE and PAGE_SIZE constexpr variables (!1916)
        - Slightly improve compile times by making less unique temp types (!1917)
        - Fix MSVC flags for CUDA builds. (!1919)
        - UnitTestFetchArrayTopologyMapIn now compiles with VS2015 (!1921)
- VTK-m Maintenance (SNL)
    - Merge Requests
        - Update logging for enabled devices (!1877)
        - Update MeshQuality examples (!1866)
        - Fix compile error for Xcode 9 (!1880)
        - Fix Compilation bug with Fill method (!1889)
        - Add Matrix.h to cellmetrics include (!1892)
        - Register Variant as trivially copyable if possible (!1898)

- Do not use std::is_trivially_copyable on GCC 4.X (!1900)
- Fix compiler warning in Tube worklet (!1903)
- Only workaround incorrect GCC behavior for OpenMP on GCC (!1904)
- Remove invalid arrays from multiplexer (!1901)
- Add VTKM_DEPRECATED macro (!1902)

Tasks to Complete the Milestone:
In the case for each of the tasks, implementation started in a private topic branch. That branch was later submitted as a merge request where the code was run through regression tests across multiple test platforms. The merge requests were also subjected to human reviewers for approval. After necessary modifications were made, the code was merged to VTK-m's master branch. Subsequently, documentation was written for the VTK-m User's Guide.

Person(s) Responsible for Completing the Milestone:

- Kenneth Moreland, Sandia National Laboratories
- Dave Pugmire, Oak Ridge National Laboratory
- David Honegger Rogers, Los Alamos National Laboratory
- Berk Geveci, Kitware, Inc.
- Hank Childs, University of Oregon
- Thomas Otahal, Sandia National Laboratories
- Nick Davis, Sandia National Laboratories
- Mark Kim, Oak Ridge National Laboratory
- James Kress, Oak Ridge National Laboratory
- Jieyang Chen, Oak Ridge National Laboratory
- Li-Ta (Ollie) Lo, Los Alamos National Laboratory
- Robert Maynard, Kitware, Inc.
- Sujin Philip, Kitware, Inc.
- Abhishek Yenpure, University of Oregon
- Steven Walton, University of Oregon

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. SAND2020-????? R