Date: 10/8/2019

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 - STDA05-20

Milestone Due Date: September 30, 2019

Milestone Completion Date: September 26, 2019

Description of Milestone:
The STDA05-20 milestone comprises the following 8 tasks.

VTK-m Release 3 We will provide a release of VTK-m software and associated documentation. The source code repository will be tagged at a stable state, and, at a minimum, tarball captures of the source code will be made available from the web site. A version of the VTK-m User's Guide documenting this release will also be made available.

Lightweight Cell Library VTK and VTK-m each have their own code to perform operations on cells. This leads to code duplication as well as further code to resolve differences between the two. To better provide code sharing between VTK, VTK-m, and other visualization software, we will separate out the cell operations into a shared, lightweight library that can be included in both projects without heavy overhead.

Path Geometry Algorithms such as streamlines can generate 1D paths through space (typically represented as polylines). Representing these paths with 3D tube or ribbon geometry can provide important visual cues for the curve's shape and can add further properties to the curve like orientation, magnitude, or uncertainty.

Specialized Data Models The data model in VTK-m has been designed to efficiently represent a large class of data layouts used by simulation codes. VTK-m currently supports a basic set of structured and unstructured meshes, but a variety of specialized representations have not been implemented yet. These include extruded meshes, which are used to represent the toroidal mesh in fusion simulations, and molecular meshes used in chemistry and biological simulations. In the particular case of fusion simulations, support for extruded meshes will make it possible to directly represent the toroidal mesh with zero-copy.

Feature-Sensitive Surface Norm Previous milestones provided the basic surface normal generation required for rendering. This algorithm usually provides a reasonable representation, but it often behaves poorly around certain features of the data. For example, the "winding" of the

polygons in a surface can be inconsistent, which means normals might point in opposite sides of the surface, causing poor shading and coloring when rendered. The averaging of normals should make sure that the directionality is consistent. Also, smooth normals do not shade sharp bends in the mesh correctly. Edges on these sharp bends must be "split" to correct the shading. These considerations take additional design and implementation.

Cell Metrics Over the past decade, mesh-quality metrics, such as "skew," "aspect ratio," "diagonal ratio," and others have helped simulation code developers identify problematic cells in tools such as VisIt and ParaView. Further, volume and area calculations are needed for integration-based analysis tasks. This milestone will ensure that these filters are available in VTK-m to fit the needs of VisIt and ParaView.

Contouring Creating contours, also known as isosurfaces in 3D, is a particular class of cell generation algorithms that is of great importance to scientific visualization. The initial Marching Cubes based implementation for contouring exists in VTK-m, but there are a variety of specializations that can be applied both to support contouring on more generalized, unstructured meshes, and to optimize the performance for more specific types of meshes. Contouring algorithms can also be used as a building block for several other algorithms, such as slicing and extracting parts from volume fractions. LANL will lead the algorithm development, SNL will integrate new VTK-m framework features with this algorithm.

Advanced Flow Algorithms Particle advection enables many other advanced flow visualization algorithms such as FTLE and stream surfaces. This milestone will explore which of these algorithms are most worthwhile in VTK-m and implement select algorithms.

Completion Proof of the Milestone:
The predefined objective completion criteria of all tasks of this milestone is to merge the functionality in VTK-m through merge requests. Additionally, all tasks require documentation in the VTK-m User's Guide. The VTK-m release task requires a dedicated version of the User's Guide. The lightweight cell library requires software in its own repository and documentation of that software.

The following table provides evidence for each implemented feature with links to the completed merge requests (evidence that the implementation is merged into the master branch) and a link to the excerpt from the VTK-m User's Guide documenting the feature.

Deliverable	Merge Requests	Documentation
VTK-m Release 3	There are too many merge requests to list independently, but an overview of the major changes for VTK-m 1.4 (the release for this deliverable) is given at https://gitlab.kitware.com/vtk/vtk-m/-/tags/v1.4.0.	http://m.vtk.org/images/3/3c/VTKmUsersGuide-1-4.pdf

Lightweight Cell Library	• Fix gradient issue at apex of pyramid cells (!1785) • Fix cell derivatives for polygon cell shape (!1819) • Use Light-Weight Cell Library (!1852)	https://jira.exascaleproject.org/secure/attachment/19775/LightweightCellLibraryDocumentation.zip
Path Geometry	• Poly line path geom (!1710) • Add Tube Filter (!1781)	https://jira.exascaleproject.org/secure/attachment/19502/Tube%20Pages%20from%20VTKmUsersGuide.pdf
Specialized Data Models	• extruded type (!1711) • Add serialization code for CellSetExtrude. (!1788) • vtkm::cont::CastAndCall now supports CellSetExtrude (!1805)	https://jira.exascaleproject.org/secure/attachment/19811/Cell%20Set%20Extrude%20Pages%20from%20VTKmUsersGuide.pdf
Feature-Sensitive Surface Norm	• Add worklets to SurfaceNormals that orient normals and triangle windings (!1614) • Split sharp edge assertion issue 373 (!1693) • Add support for BitFields. (!1629) • FindFirstSetBit now correctly compiles with the Intel compiler (!1668) • Gcc 485 fixes (!1666) • Add Mask capabilities to worklets (!1528) • Add Split sharp edges worklet and filter (!1416)	https://jira.exascaleproject.org/secure/attachment/19236/Surface%20Normal%20Pages%20from%20VTKmUsersGuide.pdf
Cell Metrics	• Add Mesh Quality and Cell Metrics (!1714) • Mesh quality metrics: Jacobian, other improvements (!1855)	https://jira.exascaleproject.org/secure/attachment/19900/Mesh%20Quality%20Pages%20from%20VTKmUsersGuide.pdf
Contouring	• Prevent floating point exceptions in Marching Cubes. (!1766) • Contour Filter (!1780) • Fix bad normals in contour test (!1794) • Tangle Source (!1804)	https://jira.exascaleproject.org/secure/attachment/19317/Contour%20Pages%20from%20VTKmUsersGuide.pdf
Advanced Flow Algorithms	• Adding changes for temporal advection, and adding test (!1590) • Particle status merge (!1756) • Lagrangian coherent structures (!1764)	• https://jira.exascaleproject.org/secure/attachment/19403/LCS%20Pages%20from%20VTKmUsersGuide.pdf

- Adding necessary imports (!1779)
- Stream surface worklet (!1740)
- Add new particle status (!1744)
- Add filter for stream surface worklet. (!1782)
- Simplify particle advection example (!1816)
- More cases for Grid Evaluator (!1811)
- Fix LCS filter to use flowmaps (!1809)
- Fixing temporal advection example (!1814)
- Increase CUDA stack size for ParticleAdvection worklets. (!1832)
- Use vtkm::FloatDefault for particle advection code. (!1827)

- https://jira.exascaleproject.org/secure/attachment/19501/Stream%20Surface%20Pages%20from%20VTKmUsersGuide.pdf

Tasks to Complete the Milestone:
In the case for each of the deliverables, 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, David Pugmire, David Rogers, Li-ta Lo, Berk Geveci, Robert Maynard, Hank Childs

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. SAND2019-XXXX R