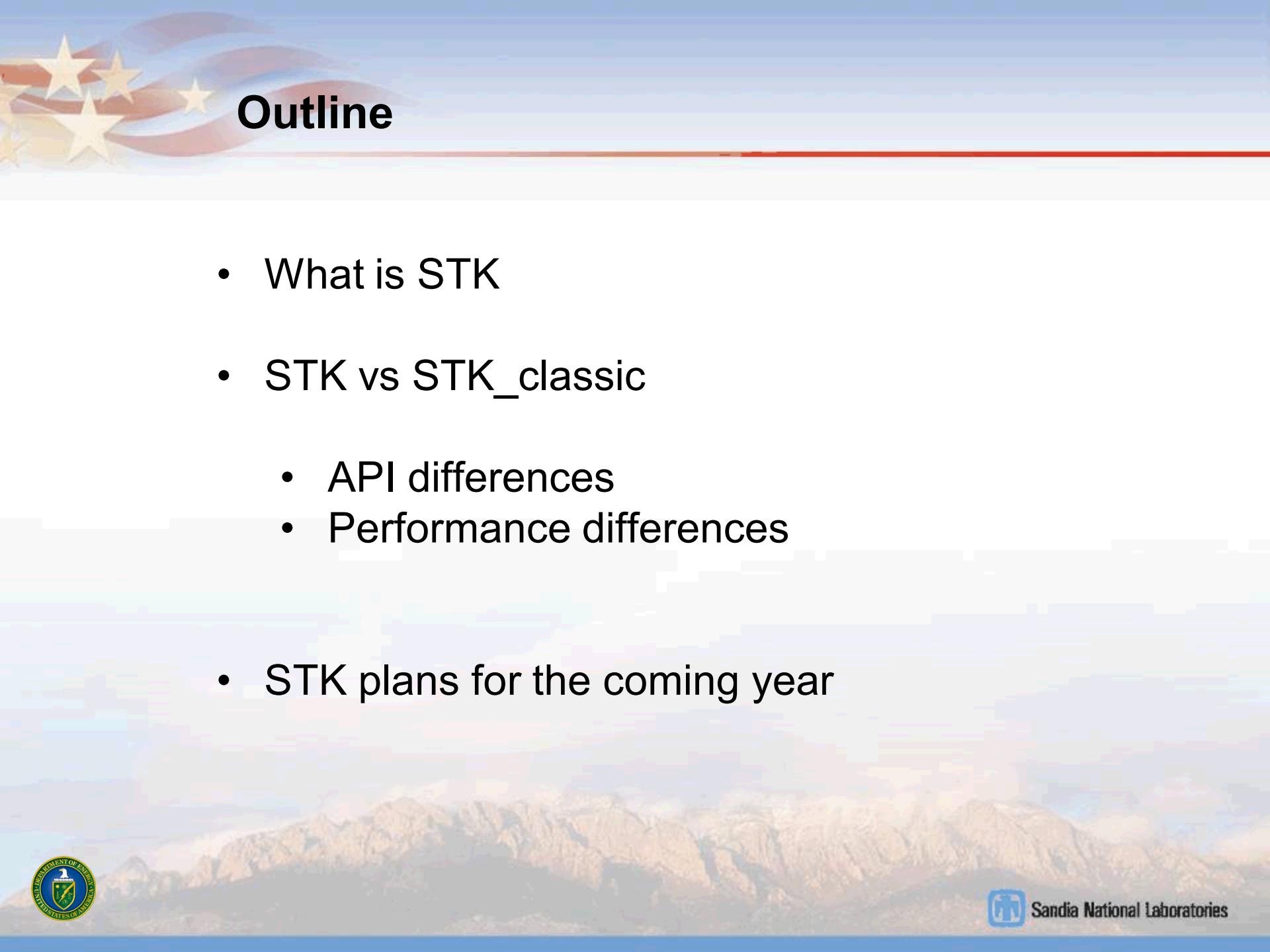


STK (Sierra Toolkit) Update

Trilinos User Group meetings, 2014

Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.





Outline

- What is STK
- STK vs STK_classic
 - API differences
 - Performance differences
- STK plans for the coming year



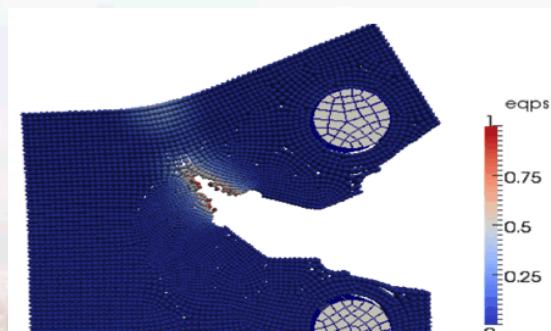
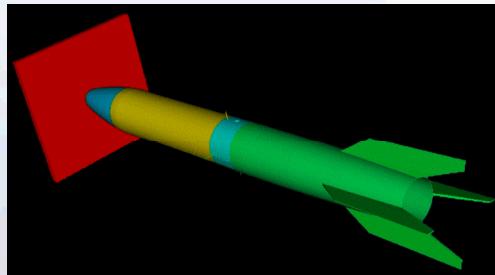
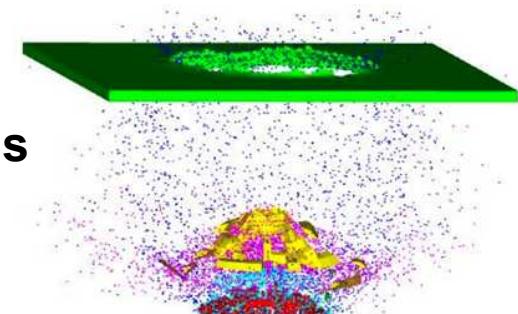
Sandia National Laboratories

What is STK

- STK is a collection of infrastructure modules supporting unstructured-mesh analysis applications.
- Developed in the SIERRA project

Finite Element/Finite Volume Analysis Applications

- Solid Mechanics
- Structural Dynamics
- Thermal/Fluid Mechanics



Sandia National Laboratories

SIERRA / STK development environment

- The SIERRA project uses a build system based on “bjam” (of boost lineage).

- We monitor nightly and continuous builds/tests and valgrind reports on cdash dashboards.

SierraNightly							
Dashboard		Calendar	Previous	Current	Project	Trac Site	
Rally Site External SQA Team Views Other GUIs							
No file changed as of Tuesday, October 14 2014 - 18:00 MDT							
Filters							
Match the following rule(s):							
Label : contains : TK							
Limit results to 0 rows (0 for unlimited)							
<input type="button" value="Apply"/> <input type="button" value="Clear"/> <input type="button" value="Create Hyperlink"/>							
Production - TLCC2							
Site	Build Name	Update		Build		Test	Build Time
		Files	Error	Warn	NotRun/Skipped		
sierra102	master-intel-12.1-debug-openmpi-1.6.4-mkl	0	0	0	0	257	Oct 14, 2014 - 18:00 MDT
uno-login3	master-intel-12.1-release-openmpi-1.6-mkl	0	0	0	0	243	Oct 14, 2014 - 18:00 MDT
chama-login8	master-intel-12.1-release-openmpi-1.6-mkl	0	0	0	0	248	Oct 14, 2014 - 18:00 MDT
chama-login8	master-intel-12.1-release-openmpi-1.6-mkl-installation	0	0	0	0	257 ¹⁵¹	Oct 14, 2014 - 18:00 MDT
sierra101	master-intel-12.1-release-openmpi-1.6.4-mkl	0	0	0	0	257 ¹⁵¹	Oct 14, 2014 - 18:00 MDT
Production - TLCC							
Site	Build Name	Update		Build		Test	Build Time
		Files	Error	Warn	NotRun/Skipped		
glory-login2	master-intel-12.1-release-openmpi-1.4.3-mkl	0	0	0	0	252	Oct 14, 2014 - 18:00 MDT
glory-login2	master-intel-12.1-release-openmpi-1.4.3-mkl-installation	0	0	0	0	252	Oct 14, 2014 - 18:00 MDT
Production - RedSky							
Site	Build Name	Update		Build		Test	Build Time
		Files	Error	Warn	NotRun/Skipped		
redsky-login4	master-intel-12.1-release-openmpi-1.6-mkl	0	0	0	0	252	Oct 14, 2014 - 18:00 MDT
redsky-login4	master-intel-12.1-release-openmpi-1.6-mkl-installation	0	0	0	0	252	Oct 14, 2014 - 18:00 MDT
Production - Cleo							
Site	Build Name	Update		Build		Test	Build Time
		Files	Error	Warn	NotRun/Skipped		
moLogin01e	master-intel-12.1.xe6-release-vendor	0	0	0	1 ¹	172 ₁	Oct 14, 2014 - 18:00 MDT
moLogin01e	master-intel-12.1.xe6-release-vendor-installation	0	0	0	1 ¹	172 ₁	Oct 14, 2014 - 18:00 MDT
Production - Sequoia							
Site	Build Name	Update		Build		Test	Build Time
		Files	Error	Warn	NotRun/Skipped		

- STK modules are updated into Trilinos periodically (thanks Brent!) and at that point the sources are built/tested using cmake/ctest support.



Sandia National Laboratories

Sierra ToolKit (STK) modules overview

Parallel-consistent Mesh database

- Heterogeneous element types
- Unstructured

Search

- Proximity, mesh independent

IO

- Bridge from mesh-data to external capability
- Built optionally

Topology

- Entity-local definitions for node orderings, etc

Util

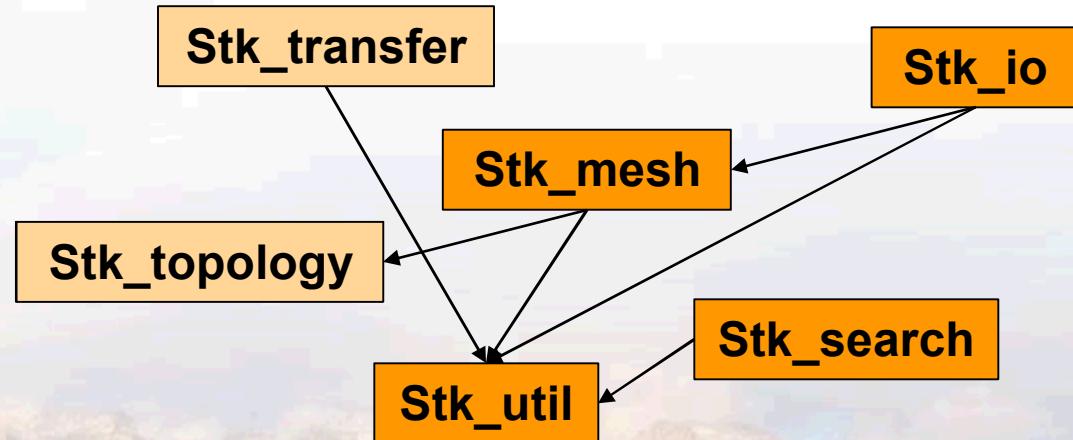
- Nearly everything depends on util directly or indirectly

Transfer

- New

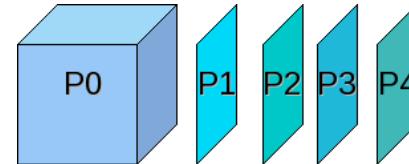
Dependency diagram:

- Arrows point towards a module that is used (depended on) by another module.

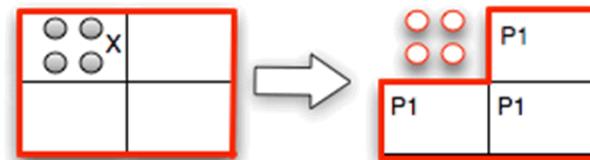


STK Mesh

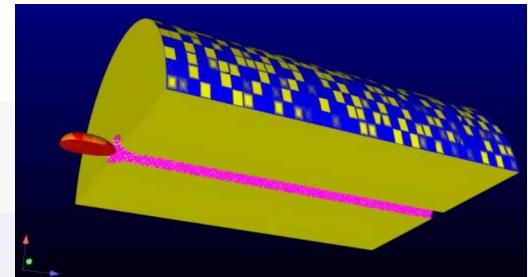
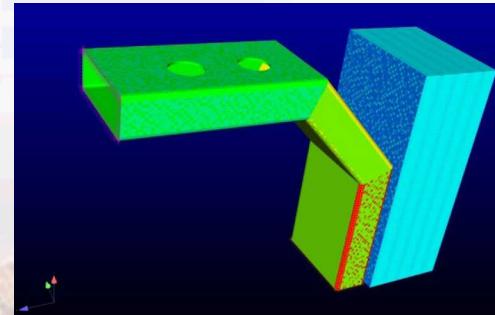
- Parallel distributed, parallel consistent



- Heterogeneous element types and field types



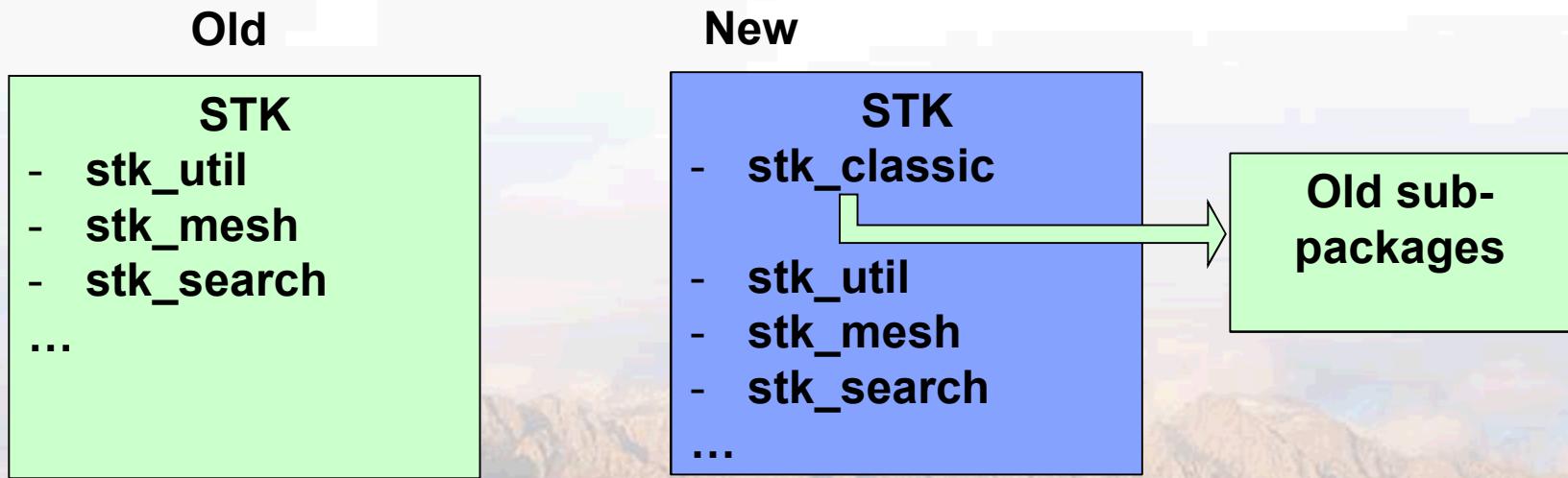
- Unstructured grid



Sandia National Laboratories

STK vs STK_classic

- Starting a couple years ago, we made major changes to the STK package, changing APIs and data-structures, etc.
 - Almost all changes are in `stk_mesh`...
- We froze the STK package in Trilinos, performed development in the SIERRA repository, and have now brought the new code back into Trilinos.



Sandia National Laboratories

STK vs STK_classic: API changes

Entity

- Entity changed from an object with methods, to an index.
- To do anything with an Entity now, you need a BulkData.

Example

Old **EntityId id = node.identifier();**

New **EntityId id = bulkdata.identifier(node);**



Sandia National Laboratories

STK vs STK_classic: API changes (continued)

Relations / Connectivity: Entities are connected...

- Previously, Entity held arrays of Relations.
- Relation class held connected entity, ordinal (local-id) and other data
- These items are now held in separate ‘raw’ arrays.

Example: connected nodes of an element

Old

```
PairIterRelation node_relations = element.relations(NODE_RANK);
Loop { Entity node = node_relations[i].entity();
```

New

```
unsigned num_nodes = bulkdata.num_nodes(element);
const Entity* nodes = bulkdata.begin_nodes(element);
Loop { Entity node = nodes[i];
```



Sandia National Laboratories



STK vs STK_classic: API changes (continued)

- Fields can only be associated with 1 entity rank
 - improved performance, reduced complexity
- IO: **MeshReadWriteUtils** replaced by **StkMeshIoBroker**
 - several API differences, see example materials
- **Shards** (topology definitions) is being replaced by **stk_topology**
 - we will maintain mappings back and forth for users who continue to use Shards.



Sandia National Laboratories

STK vs STK_classic: API changes (continued)

- More detail is covered in a conversion guide written by Jim Foucar (thanks Jim!)
- Also see documentation tests which illustrate usage of a lot of the new STK code-base.
See `stk/stk_doc_tests`



Sandia National Laboratories

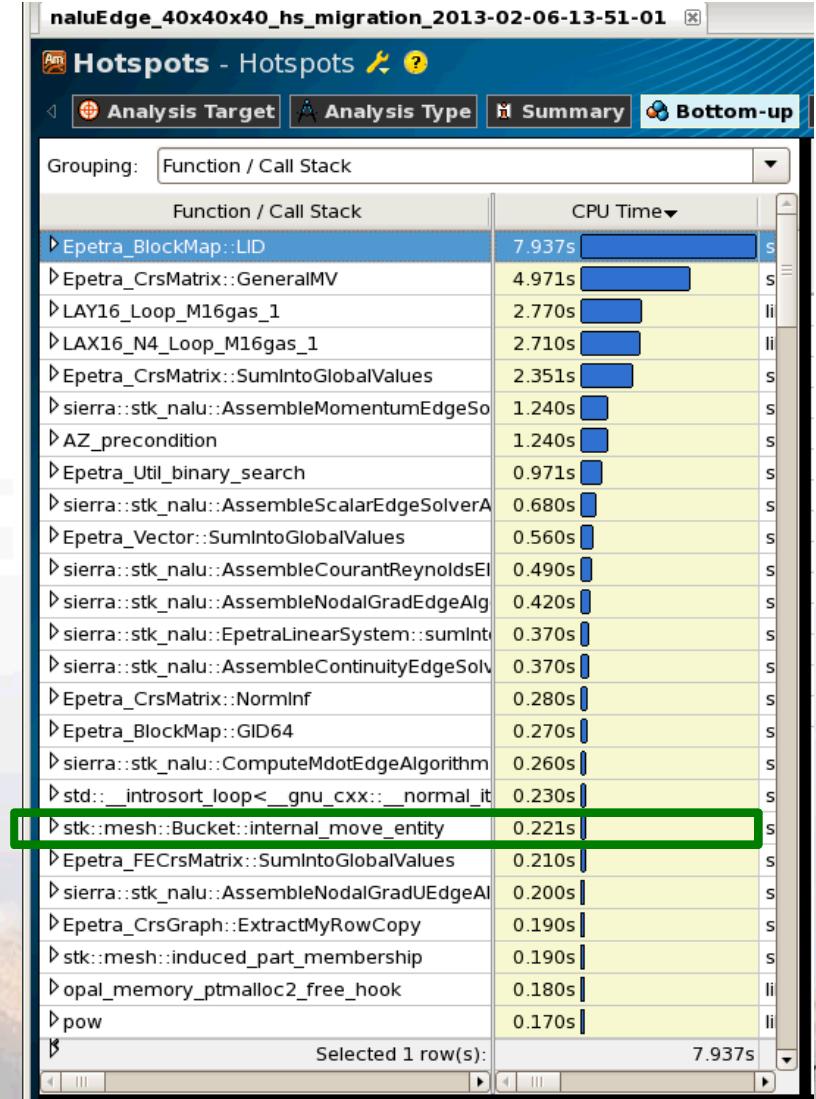
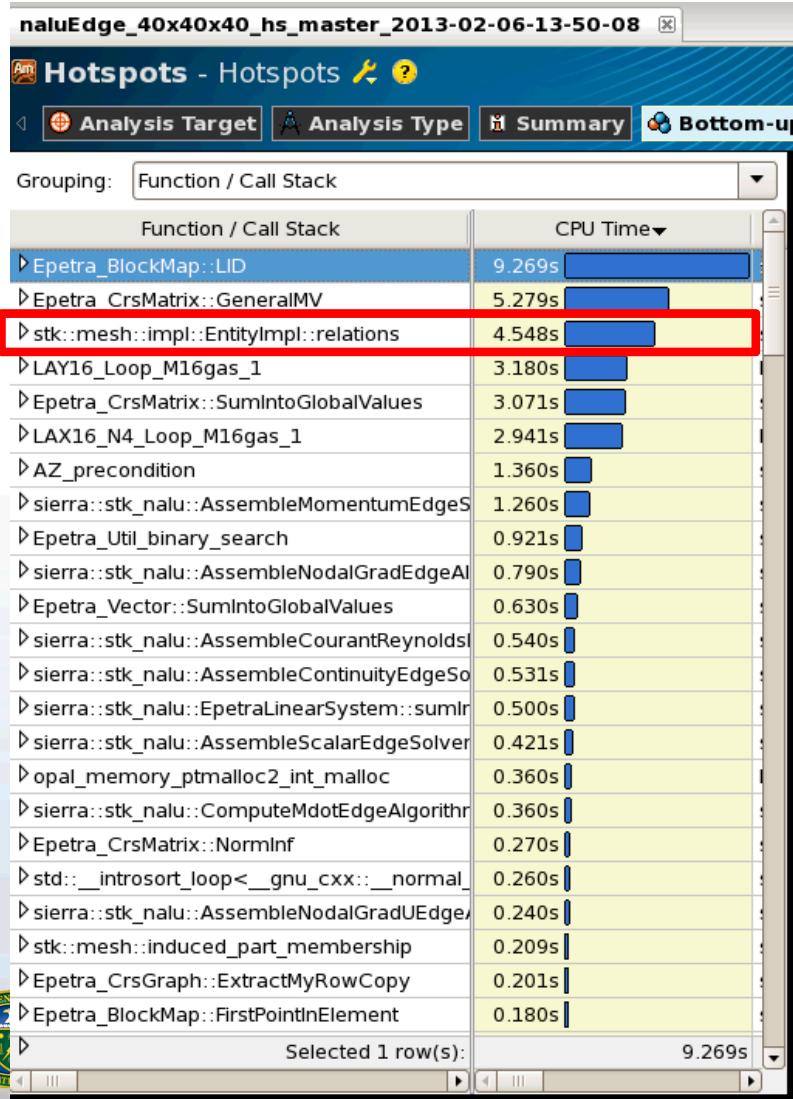
STK vs STK_classic: Performance improvements

- Entity changes and connectivity storage changes improved cache efficiency of connectivity traversal
- Restricting fields to a single entity-rank improved speed of field-data access (removed pointer hop(s))
- Selector caching improved speed of bucket access
- Scalable entity creation
 - `stk::mesh::create_edges(mesh)...`
 - ...was reported as a scalability bottleneck for runs using large numbers of MPI ranks



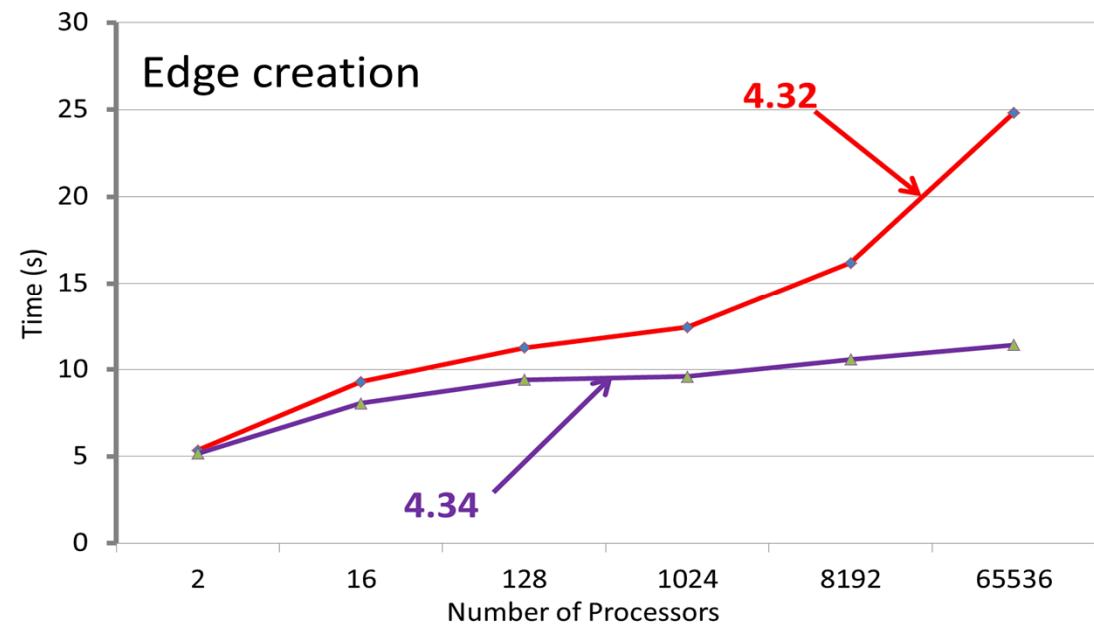
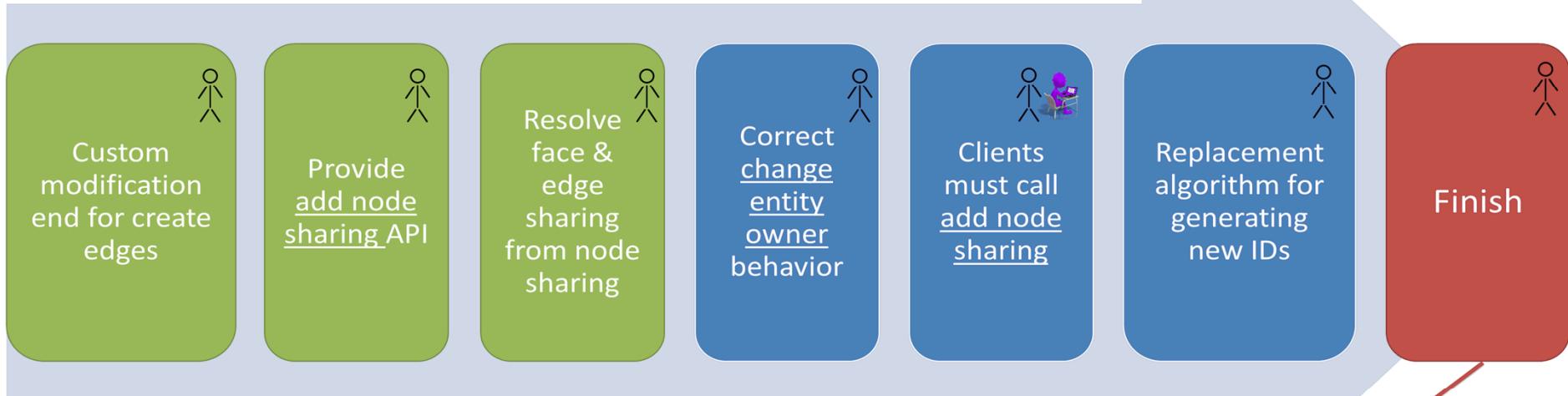
Sandia National Laboratories

STK vs STK_classic: Performance Improvements (continued)



Scalable Entity Creation

STK person
App person



- Make sure all internal code is using `generate_new_ids`
- Obtain new scalability data on Cielo

STK plans for the coming year

Feature Development, enhancements

Scalable Entity Creation

IO

Ghosting

Arbitrary Relations

Balance/Rebalance

IO Auto Decomp

Batch Mesh Modification

Research

- **Next generation platforms**
- **Threading, GPUs, Kokkos?**
- **Data Structures**
- **Hierarchical Decomposition**



Sandia National Laboratories