# PHASTA Science Gateway for High Performance Computational Fluid Dynamics

**Extended Abstract** 

Cameron W. Smith Rensselaer Polytechnic Institute Troy, NY smithc11@rpi.edu

> Suresh Marru\* Indiana University Bloomington, Indiana smarru@iu.edu

#### **ABSTRACT**

The Parallel Hierarchic Adaptive Stabilized Transient Analysis (PHASTA) software supports modeling compressible or incompressible, laminar or turbulent, steady or unsteady flows in 3D using unstructured grids. PHASTA has been applied to industrial and academic flows on complex, as-designed geometric models with over one billion mesh elements using upwards of one million compute cores. The PHASTA Science Gateway (phasta.scigap.org) brings these increasingly critical technologies to a larger user base by providing a central hub for simulation execution, simulation data management, and documentation. Researchers and engineers using the gateway can easily define and execute simulations on the TACC Stampede2 Skylake and Knights Landing nodes without being burdened by the details of remote access, the job scheduler, and filesystem configuration. In addition to simplifying the simulation execution process, the gateway creates a searchable archive of past jobs that can be shared with other users to support reproducibility and increase productivity. Our poster presents the construction of the gateway with Apache Airavata, the simulation definition process, applications it currently supports, and our ongoing efforts to expand functionality, the user base, and the community.

## **CCS CONCEPTS**

 Computing methodologies → Massively parallel and highperformance simulations; Simulation tools; • Software and its engineering → Reusability; • Applied computing → Computeraided design;

#### **KEYWORDS**

CFD, Paralllel Unstructured Mesh, Apache Airavata, Science Gateway

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

For an other uses, contact the owner/author(s).

PEARC '18, July 22−26, 2018, Pittsburgh, PA, USA

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-6446-1/18/07.

https://doi.org/10.1145/3219104.3229243

Eroma Abeysinghe\*
Indiana University
Bloomington, Indiana
eabeysin@iu.edu

Kenneth E. Jansen University of Colorado, Boulder Boulder, Colorado kenneth.jansen@colorado.edu

#### **ACM Reference Format:**

Cameron W. Smith, Eroma Abeysinghe, Suresh Marru, and Kenneth E. Jansen. 2018. PHASTA Science Gateway for High Performance Computational Fluid Dynamics: Extended Abstract. In *PEARC '18: Practice and Experience in Advanced Research Computing, July 22–26, 2018, Pittsburgh, PA, USA*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3219104.3229243

## 1 INTRODUCTION

PHASTA is a parallel [20] computational fluid dynamics (CFD) analysis package that supports simulation of complex physical flow systems on unstructured meshes [4, 21, 22, 29, 35] using a stabilized finite element method [34]. Users define PHASTA simulations using CAD modeling systems, mesh generators, and attribute definition interfaces. Problem definition information is then uploaded to the PHASTA science gateway for execution on high performance computing systems. When analysis completes the user downloads a subset of the outputs for post-processing.

Sections 2 and 3 review the technologies supporting PHASTA simulation definition and the gateway's implementation. Next, in Section 4, the goals of the NSF SI2-S2I2 conceptualization grant to generalize and improve workflows and tools for the greater fluid dynamics community are introduced. Section 5 closes the abstract with a summary of capabilities and ongoing developments.

#### 2 PHASTA SIMULATION WORKFLOW

A typical PHASTA workflow is composed of the following steps: (1) problem definition and mesh generation, (2) processing user inputs to produce data structures PHASTA needs for analysis execution, (3) partitioning and load balancing [30], (4) adaptive flow analysis, and (5) post-processing. The PHASTA science gateway automates the execution of steps (2) through (4) on a remote HPC system. Step (4) uses in-memory streams [28] to iteratively execute the PHASTA analysis with mesh adaptation, dynamic load balancing, and reordering functionality provided by the PUMI unstructured mesh services library [8, 11, 24]. Steps (1) and (5) are executed on a user's system using interactive tools.

In Step (1), user's define a PHASTA simulation based on a geometric model; a specific combination of topological and shape information for the problem domain's boundary [25, 33]. The source of this information can be full-featured solid modeling kernels (e.g.,

<sup>\*</sup>Science Gateways Research Center, Pervasive Technology Institute

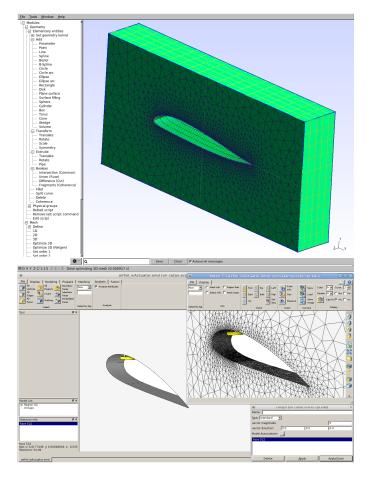


Figure 1: Graphical interfaces for Gmsh mesh generation (top) and Simmetrix SimModeler mesh generation and problem definition (bottom).

OpenCASCADE [23], Parasolid [26], ACIS [2], and Granite [19]), a simplified kernel [7], or voxel image data [12], Problem definition information is associated with geometric model entities via a graphical interface, such as Simmetrix SimModeler [27], or via a plain text file. Mesh generation is supported via script, API, and graphical interfaces. Figure 1 depicts the Gmsh [5, 6] and SimModeler mesh generation interfaces. Functionalities also exist to import meshes from ALFR UGRID [13] and other file formats. Importing meshes in this manner comes at the cost of reduced geometric model information which limits the ability of mesh adaptation to improve the geometric approximation of the mesh to the model.

After a job completes, PHASTA checkpoint files can be loaded into Kitware's ParaView [31] to visualize analysis fields on the partitioned mesh.

## 3 PHASTA SCIENCE GATEWAY PORTAL

## 3.1 Design and Implementation

Providing PHASTA analysis capabilities through a web-based interface allows users to execute complex simulations on HPC systems without being encumbered by remote access, filesystem, and job

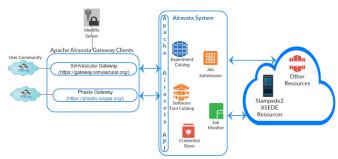


Figure 2: PHASTA Gateway interactions with Apache Airavata and computational resources.

scheduler interactions. Users are also given a built-in simulation data management functionality; the inputs, outputs, and execution environment metadata associated with a completed gateway simulation is archived for later reuse and sharing with other users. Reuse and sharing of a given problem can support fixing errors, performance tuning, studying the effect of model or input parameter changes, and publication. Embedding data management and remote execution into the gateway enables experts in those domains to provide simplified interfaces that let researchers and engineers focus on gaining knowledge from their simulations.

The PHASTA gateway is implemented over the open source Apache Airavata [14] software framework. To focus on the community, PHASTA gateway is leveraging SciGaP services, the Science Gateway Platform as a Service [17, 18], a hosted platform based on Apache Airavata. Leveraging a common platform builds upon identity management, accounts and authorization, and the ability to access multiple computational resources. Figure 2 displays the functional features provided by Apache Airavata middleware that connect the gateway web client interface to multiple remote HPC and other resources.

## 3.2 Access

PHASTA gateway users are given access by the gateway's administrators. User identity management is handled by the Keycloak [1] identity management system. Keycloak supports user access through either existing organizational accounts via a federated authentication system, or by creation of local accounts in the gateway. Sets of users with common access and analysis needs can be placed into groups [16]. Specifically, groups can be used to associate users with HPC system allocations, to limit resource usage on a system, and control which workflows and software deployments are accessible. For example, a group of workshop participants may have access to a limited number of compute nodes and wall time relative to a group created for researchers. Likewise, research groups may need access to a version of the software that may include experimental physics models that are not ready for a production or training environment.

## 3.3 Experiment Execution

The PHASTA gateway allows its users to create, execute, monitor, share, and manage computational jobs, which are referred to

as 'experiments' in the gateway interface. Launching an experiment in the gateway submits a computational job to the XSEDE Stampede2 cluster. Apache Airavata communicates with Stampede2 through secure SSH connections. Gateway administrators can generate SSH key+token pairs as needed using Credential Store [10] in the gateway. PHASTA problem information required for simulation execution is specified in the gateway interface through file uploads and fields for STRING, INTEGER and FLOAT values.

Once the job is submitted, the job status in the cluster will be periodically updated. Additionally, users can provide their email addresses to receive messages with these status updates. At job completion or failure the gateway will display all the output files moved from the Stampede2 to the gateway's data storage area. Output could be in the form of files as well as text derived from specified files by the gateway admin at the application configuration level.

PHASTA experiments that require more wall time than allowed by the system for a single batch job can be automatically split into a sequence of chained jobs. The number of jobs needed depends on the HPC system's job queue limits. For example, if the user requests 5760 minutes for a job in the Stampede2 'normal' queue whose max wall-time is 2880 minutes, the gateway will submit two jobs, breaking each at the maximum wall-time limit. The second PHASTA job in the sequence will restart using a PHASTA file-based checkpoint-restart mechanism [3].

#### 4 EXPANDING THE COMMUNITY

A next generation of tools, like the PHASTA Science Gateway and the SimVascular Supercomputing Gateway [15, 32], are needed by the broader fluid dynamics community to accelerate discovery, facilitate deeper interactions between experimental and computational practitioners, and provide graduate students in the field a high quality set of interoperable components to base their work on. To support the definition of extensible end-to-end workflows the tools need to leverage best practices from fluid dynamics research, data analytics/science, and software engineering. Towards this, the Computational Fluid Dynamics Software Infrastructure (CFDSI) [9] is being conceptualized over the next 18 months to determine the specific capabilities needed, and their operational requirements in an HPC environment. In this first year of CFDSI, an all-hands kick-off meeting and several sub-committee meetings will be held to formalize these software details. Topics include distinguishing modeling and discretization errors, defining comprehensive benchmarks and tests to verify and validate components, post-processing high-order field information, and reducing the barrier for verification against experiments. Although these capabilities are focused on supporting the fluid dynamics community, improvements in functionality, performance, and usability of the tools that provide them will impact other problem domains governed by partial differential equations.

#### 5 CLOSING REMARKS

The PHASTA Science Gateway simplifies the execution of parallel, adaptive PHASTA simulations by hiding the details on data management and job submission for complex workflows on high performance computing system. For new PHASTA users the gateway reduces the technical barrier to interactions with the graphical

and text-based tools for problem definition, mesh generation, and post-processing. Ongoing efforts with the Science Gateway team are addressing gateway usability improvements and conceptualization of a software institute for CFDSI: Model, Data, and Analysis Integration for End-to-End Support of Fluid Dynamics Discovery and Innovation.

#### **ACKNOWLEDGMENTS**

The work is supported by the National Science Foundation under Grant No.: NISC SI2-S2I2 Conceptualization of CFDSI: Model, Data, and Analysis Integration for End-to-End Support of Fluid Dynamics Discovery and Innovation and Collaborative Research: SI2-SSI: Open Gateway Computing Environments Science Gateways Platform as a Service (OGCE SciGaP).

## **REFERENCES**

- Marcus Christie, Anuj Bhandar, Supun Nakandala, Suresh Marru, Eroma Abeysinghe, Sudhakar Pamidighantam, and Marlon Pierce. 2017. Using Keycloak for Gateway Authentication and Authorization. 4. DOI: http://dx.doi.org/10.6084/ m9.figshare.5483557.v1
- [2] Jonathan Corney. 1997. 3D Modeling Using the Acis Kernel and Toolkit (1st ed.). John Wiley & Sons, Inc., New York, NY, USA.
- [3] Jing Fu, Ning Liu, Onkar Sahni, Kenneth E Jansen, Mark S Shephard, and Christopher D Carothers. 2010. Scalable parallel I/O alternatives for massively parallel partitioned solver systems. In Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on. IEEE, 1–8.
- [4] Azat Yu. Galimov, Onkar Sahni, Richard T. Lahey Jr., Mark S. Shephard, Donald A. Drew, and Kenneth E. Jansen. 2010. Parallel Adaptive Simulation of a Plunging Liquid Jet. Acta Mathematica Scientia 30, 2 (March 2010), 522–538. DOI: http://dx.doi.org/10.1016/S0252-9602(10)60060-4
- [5] Christophe Geuzaine and Jean-François Remacle. 2009. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. Int. J. Numerical Methods in Eng. 79, 11 (May 2009), 1309–1331. DOI: http://dx.doi.org/ 10.1002/nme.2579
- [6] Christophe Geuzaine and Jean-François Remacle. 2013. Gmsh: A threedimensional finite element mesh generator with built-in pre- and post-processing facilities. (2013). http://gmsh.info/
- [7] Daniel Ibanez. 2017. Gmsh model generation library. (2017). https://github.com/ ibaned/gmodel/tree/ca6af6876265a02ecf0bc97aad8d3c5a6c19a638
- [8] Daniel A. Ibanez, E. Seegyoung Seol, Cameron W. Smith, and Mark S. Shephard. 2016. PUMI: Parallel Unstructured Mesh Infrastructure. ACM Trans. Math. Softw. 42, 3, Article 17 (May 2016), 28 pages. DOI: http://dx.doi.org/10.1145/2814935
- [9] Kenneth E. Jansen. 2018. CFDSI GitHub repository. (2018). https://github.com/ CFDSI
- [10] Thejaka Amila Kanewala, Suresh Marru, Jim Basney, and Marlon Pierce. 2014. A credential store for multi-tenant science gateways. In Cluster, Cloud and Grid Computing (CCGrid), 2014 14th IEEE/ACM International Symposium on. IEEE, 445–454.
- [11] Anil Kumar Karanam, Kenneth E. Jansen, and Christian H. Whiting. 2008. Geometry based pre-processor for parallel fluid dynamic simulations using a hierarchical basis. Engineering with Computers 1 (2008), 17.
- [12] Ottmar Klaas, Mark Beall, and Mark S. Shephard. 2014. Generation of Geometric Models and Meshes from Segmented Image Data. In Proc. NAFEMS Americas Conf. NAFEMS, 1–4.
- [13] David L. Marcum. 2017. SimSys Software: UG\_IO SimSys Files. (2017). http://www.simcenter.msstate.edu/software/downloads/doc/ug\_io/
- [14] Suresh Marru, Lahiru Gunathilake, Chathura Herath, Patanachai Tangchaisin, Marlon Pierce, Chris Mattmann, Raminder Singh, Thillina Gunarathne, Eran Chinthaka, Ross Gardler, Aleksander Slominski, Ate Douma, Srinath Perera, and Sanjiva Weerawarana. 2011. Apache Airavata: A Framework for Distributed Applications and Computational Workflows. In Proc. ACM Workshop on Gateway Comput. Environments. 21–28. DOI: http://dx.doi.org/10.1145/2110486.2110490
- [15] Alison L. Marsden. 2017. SimVascular Supercomputing Gateway. (2017). https://gateway.simvascular.org/
- 16] Supun Nakandala, Suresh Marru, Marlon Piece, Sudhakar Pamidighantam, Kenneth Yoshimoto, Terri Schwartz, Subhashini Sivagnanam, Amit Majumdar, and Mark A Miller. 2017. Apache Airavata Sharing Service: A Tool for Enabling User Collaboration in Science Gateways. In Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact. ACM, 20.

- [17] Marlon Pierce, Suresh Marru, Borries Demeler, Amitava Majumdar, and Mark Miller. 2013. Science Gateway Operational Sustainability: Adopting a Platformas-a-Service Approach. (Sept. 2013). DOI: http://dx.doi.org/10.6084/m9.figshare. 790760.v1
- [18] Marlon Pierce, Suresh Marru, Borries Demeler, Raminderjeet Singh, and Gary Gorbet. 2014. The Apache Airavata Application Programming Interface: Overview and Evaluation with the UltraScan Science Gateway. In Proc. 9th Gateway Comput. Environments Workshop. 25–29. DOI: http://dx.doi.org/10.1109/GCE.2014.15
- [19] PTC. 2014. The PTC Creo Granite Interoperability Kernel. (2014). http://www.ptc.com/~/media/Files/PDFs/CAD/GRANITE\_Interoperability\_Kernel.pdf
- [20] Michel Rasquin, Cameron Smith, Kedar Chitale, E. Seegyoung Seol, Benjamin A. Matthews, Jeffrey L. Martin, Onkar Sahni, Raymond M. Loy, Mark S. Shephard, and Kenneth E. Jansen. 2014. Scalable Implicit Flow Solver for Realistic Wing Simulations with Flow Control. Comput. in Sci. & Eng. 16, 6 (Dec. 2014), 13–21. DOI: http://dx.doi.org/10.1109/MCSE.2014.75
- [21] Joseph M. Rodriguez, Onkar Sahni, Richard T. Lahey Jr., and Kenneth E. Jansen. 2013. A Parallel Adaptive Mesh Method for the Numerical Simulation of Multiphase Flows. *Comput. & Fluids* 87 (Oct. 2013), 115–131. DOI: http://dx.doi.org/10. 1016/j.compfluid.2013.04.004
- [22] Onkar Sahni, Min Zhou, Mark S Shephard, and Kenneth E Jansen. 2009. Scalable implicit finite element solver for massively parallel processing with demonstration to 160k cores. In Proc. Int. Conf. High Performance Comput., Networking, Storage and Anal. (SC). IEEE, 1–12.
- [23] Open CASCADE SAS. 1999. Open CASCADE Modeling Kernel. (1999). https://www.opencascade.com/content/latest-release
- [24] SCOREC. 2014. PUMI GitHub repository. (2014). https://github.com/SCOREC/ core
- [25] Mark S Shephard, M.W. Beall, R.M. O'Bara, and B.E. Webster. 2004. Toward simulation-based design. Finite Elements in Anal. and Design 40, 12 (July 2004), 1575–1598.
- [26] Siemens. 1986. Parasolid 3D Geometric Modeling Engine. (1986). http://www.plm.automation.siemens.com/en\_us/products/open/parasolid/
- [27] Simmetrix. 1997. Simmetrix: Enabling Simulation-Based Design. (1997). http://www.simmetrix.com/
- [28] Cameron W. Smith, Brian Granzow, Gerrett Diamond, Dan A. Ibanez, Onkar Sahni, Kenneth E. Jansen, and Mark S. Shephard. 2018. In-memory Integration of Existing Software Components for Parallel Adaptive Unstructured Mesh Workflows. Concurrency and Computation: Practice and Experience (March 2018), 1–19. DOI: http://dx.doi.org/10.1002/cpe.4510
- [29] Cameron W. Smith and Kenneth E. Jansen. 2015. PHASTA GitHub repository. (2015). https://github.com/PHASTA/phasta
- [30] Cameron W. Smith, Michel Rasquin, Dan Ibanez, Kenneth E. Jansen, and Mark S. Shephard. 2018. Improving Unstructured Mesh Partitions for Multiple Criteria Using Mesh Adjacencies. SIAM J. Scientific Comput. (Feb. 2018), C47–C75. DOI: http://dx.doi.org/10.1137/15M1027814
- [31] Amy Henderson Squillacote, James Ahrens, Charles Law, Berk Geveci, Kenneth Moreland, and Brad King. 2007. *The paraview guide*. Vol. 366. Kitware.
- [32] Adam Updegrove, Nathan M. Wilson, Jameson Merkow, Hongzhi Lan, Alison L. Marsden, and Shawn C. Shadden. 2017. SimVascular: An Open Source Pipeline for Cardiovascular Simulation. *Annals of Biomedical Engineering* 45, 3 (01 March 2017), 525–541. DOI: http://dx.doi.org/10.1007/s10439-016-1762-8
- [33] Kevin Weiler. 1988. The radial edge structure: a topological representation for non-manifold geometric boundary modeling. In Geometric Modeling for CAD Appl. First IFIP WG5.2 Work. Conf. 3–36.
- [34] Christian H. Whiting and Kenneth E. Jansen. 2001. A stabilized finite element method for the incompressible Navier-Stokes equations using a hierarchical basis. Int. J. Numerical Methods in Fluids 35, 1 (Jan. 2001), 93–116. DOI: http: //dx.doi.org/10.1002/1097-0363(20010115)35:1<93::AID-FLD85>3.0.CO;2-G
- [35] M. Zhou, O. Sahni, H. J. Kim, C. A. Figueroa, C. A. Taylor, M. S. Shephard, and K. E. Jansen. 2010. Cardiovascular Flow Simulation at Extreme Scale. *Computational Mech.* 46, 1 (Dec. 2010), 71–82.