NASA SBIR/STTR
Small Business Innovation Research / Small Business Technology Transfer

NASA SBIR 2014 Phase I Solicitation

S5 Information Technologies

NASA Missions and Programs create a wealth of science data and information that are essential to understanding our earth, our solar system and the universe. Advancements in information technology will allow many people within and beyond the Agency to more effectively analyze and apply these data and information to create knowledge. For example, modeling and simulation are being used more pervasively throughout NASA, for both engineering and science pursuits, than ever before. These are tools that allow high fidelity simulations of systems in environments that are difficult or impossible to create on Earth, allow removal of humans from experiments in dangerous situations, provide visualizations of datasets that are extremely large and complicated, and aid in the design of systems and missions. In many of these situations, assimilation of real data into a highly sophisticated physics model is needed. Information technology is also being used to allow better access to science data, more effective and robust tools for analyzing and manipulating data, and better methods for collaboration between scientists or other interested parties. The desired end result is to see that NASA data and science information are used to generate the maximum possible impact to the nation: to advance scientific knowledge and technological capabilities, to inspire and motivate the nation's students and teachers, and to engage and educate the public.

Subtopics

S5.01 Technologies for Large-Scale Numerical Simulation

Lead Center: ARC

Participating Center(s): GSFC

NASA scientists and engineers are increasingly turning to large-scale numerical simulation on supercomputers to advance understanding of complex Earth and astrophysical systems, and to conduct high-fidelity aerospace engineering analyses. The goal of this subtopic is to increase the mission impact of NASA's investments in supercomputing systems and associated operations and services. Specific objectives are to:

- Decrease the barriers to entry for prospective supercomputing users.
- Minimize the supercomputer user's total time-to-solution (e.g., time to discover, understand, predict, or design).
- Increase the achievable scale and complexity of computational analysis, data ingest, and data communications.
- Reduce the cost of providing a given level of supercomputing performance on NASA applications.
- Enhance the efficiency and effectiveness of NASA's supercomputing operations and services.

Expected outcomes are to improve the productivity of NASA's supercomputing users, broaden NASA's supercomputing user base, accelerate advancement of NASA science and engineering, and benefit the supercomputing community through dissemination of operational best practices.

The approach of this subtopic is to seek novel software and hardware technologies that provide notable benefits to NASA's supercomputing users and facilities, and to infuse these technologies into NASA supercomputing operations. Successful technology development efforts under this subtopic would be considered for follow-on funding by, and infusion into, NASA's high-end computing (HEC) projects - the High End Computing Capability project at Ames and the Scientific Computing project at Goddard. To assure maximum relevance to NASA, funded SBIR contracts under this subtopic should engage in direct interactions with one or both HEC projects, and with key HEC users where appropriate. Research should be conducted to demonstrate technical feasibility and NASA relevance during Phase I and show a path toward a Phase II prototype demonstration.

Offerors should demonstrate awareness of the state-of-the-art of their proposed technology, and should leverage existing commercial capabilities and research efforts where appropriate. Open source software and open standards are strongly preferred. Note that the NASA supercomputing environment is characterized by:

- HEC systems operating behind a firewall to meet strict IT security requirements.
- Communication-intensive applications.
- Massive computations requiring high concurrency.
- Complex computational workflows and immense datasets.
- The need to support hundreds of complex application codes - many of which are frequently updated by the user/developer.

As a result, solutions that involve the following must clearly explain how they would work in the NASA environment:

- Grid computing.
- Web services.
- Client-server models.
- Embarrassingly parallel computations.
- Technologies that require significant application re-engineering.

Projects need not benefit all NASA HEC users or application codes, but demonstrating applicability to an important NASA discipline, or even a key NASA application code, could provide significant value.

Specific technology areas of interest:

- Efficient Computing - In spite of the rapidly increasing capability and efficiency of supercomputers, NASA's HEC facilities cannot purchase, power, and cool sufficient HEC resources to satisfy all user demands. This subtopic element seeks dramatically more efficient and effective supercomputing approaches in terms of their ability to supply increased HEC capability or capacity per dollar and/or per Watt for real NASA applications. Examples include:
    - Novel computational accelerators and architectures.
    - Cloud supercomputing with high performance interconnects (e.g., InfiniBand).
    - Enhanced visualization technologies.
    - Improved algorithms for key codes.
    - Power-aware "Green" computing technologies and techniques.
    - Approaches to effectively manage and utilize many-core processors including algorithmic changes, compiler techniques and runtime systems.
- User Productivity Environments - The user interface to a supercomputer is typically a command line in a text window. This subtopic element seeks more intuitive, intelligent, user-customizable, and integrated interfaces to supercomputing resources, enabling users to more completely leverage the power of HEC to increase their productivity. Such an interface could enhance many essential supercomputing tasks: accessing and managing resources, training, getting services, developing and porting codes (e.g., debugging and performance analysis), running computations, managing files and data, analyzing and visualizing results, transmitting data, collaborating, etc.
- Ultra-Scale Computing - Over the next decade, the HEC community faces great challenges in enabling its users to effectively exploit next-generation supercomputers featuring massive concurrency to the tune of

millions of cores. To overcome these challenges, this subtopic element seeks ultra-scale computing technologies that enable resiliency/fault-tolerance in extreme-scale (unreliable) systems both at job startup and during execution. Also of interest are system and software co-design methodologies, to achieve performance and efficiency synergies. Finally, tools are sought that facilitate verification and validation of ultra-scale applications and systems.

S5.02 Earth Science Applied Research and Decision Support

Lead Center: SSC

Participating Center(s): AFRC, ARC, GSFC, JPL

The NASA Applied Sciences Program (http://nasascience.nasa.gov/earth-science/applied-sciences) seeks innovative and unique approaches to increase the utilization and extend the benefit of Earth Science research data to better meet societal needs. One area of interest is new decision support tools and systems for a variety of ecological applications such as managing coastal environments, natural resources or responding to natural disasters.

Currently, creating decision support tools (DST) that effectively utilize remote sensing data requires significant efforts by experts in multiple domains. This creates a barrier to the widespread use of Earth observations by state and local governments, businesses, and the public. This subtopic aims to democratize the creation of Earth science driven decision support tools and to unleash a creative explosion of DST development that significantly increases the return on investment for Earth science missions.

Specifically, this subtopic develops core capabilities that can be integrated to build multiple remote sensing driven DSTs customized to the requirements of different users in varied fields. Proven development and commercialization strategies will be used to meet these objectives. Similar to Eclipse, this subtopic will create an open-source DST development framework that enables components from multiple providers to be seamlessly integrated. This subtopic will also create software components that plug into the framework and open source tools that help users create new components. The components will provide functionality ranging from basic operations, such as retrieval of data meeting user-specified criteria from online repositories and visualization, to sophisticated data processing and analysis algorithms, such as atmospheric correction, data fusion, computational model interfaces, and machine learning based quality control.

To expedite DST development and deployment by knowledgeable users, this subtopic seeks an open source graphical workflow tool, similar to Labview or Simulink, which enables well informed users to quickly create a functional DST from a catalog of software components. Ultimately, a more sophisticated graphical workflow development tool, similar to MIT's Scratch would enforce functionally, but not necessarily logically, “correct by construction” rules that would enable a broad population of people to successfully create DSTs. Open source and commercial components, as well as services, will be available through an online “store” similar to iTunes or Google Play.

The framework, components and resulting DSTs should be able to run in a commercial cloud such as Amazon EC2 or Google Compute Engine. Cloud enabled components and DSTs, those that can intelligently take advantage of flexible computing resources for processing, analysis, visualization, optimization, etc. are highly desired.

Ideally, users should be able to create, configure deploy DSTs, and view outputs such as status, reports, alerts, plots, maps, etc. via desktop computers (Windows 7 and OS X) as well as tablet and smart phones running recent versions of Android (4.0 and later) and iOS (5.0 and later). An HTML5 web application in a standards compliant browser, such as Chrome, can provide the required level of interoperability and capability. Due to serious security issues, Java and Flash based approaches will not be considered.

S5.03 Algorithms and Tools for Science Data Processing, Discovery and Analysis, in State-of-the-Art Data Environments

Lead Center: GSFC

The size of NASA's observational data sets is growing dramatically as new missions come on line. In addition, NASA scientists continue to generate new models that regularly produce data sets of hundreds of terabytes or more. It is growing ever increasingly difficult to manage all of the data through its full lifecycle, as well as provide effective data analytical methods to analyze the large amount of data. For example, the HyspIRI mission is expected to produce an average science data rate of 800 million bits per second (Mbps), JPSS-1 will be 300 Mbps and NPP is already producing 300 Mbps, compared to 150 Mbps for the EOS-Terra, Aqua and Aura missions. Other examples are SDO with a rate of 150 Mbps and 16.4 Gigabits for a single image from the HiRise camera on the Mars Reconnaissance Orbiter (MRO).

This subtopic area seeks innovation and unique approaches to solve issues associated around the use of "Big Data" within NASA. The emphasis of this subtopic is on tools that leverage existing systems, interfaces, and infrastructure, where it exists and where appropriate. Reuse of existing NASA assets is strongly encouraged.

Specifically, innovations are being sought in the following areas:

- *Parallel Processing for Data Analytics* - Open source tools like the Hadoop Distributed File Systems (HDFS) have shown promise for use in simple MapReduce operations to analyze model and observation data. In addition to HDFS, there is a rapid emergence and adoption of cloud software packages integrated with object stores, such as OpenStack and Swift. The goal is to accelerate these types of open source tools for use with binary structured data from observations and model output using MapReduce or a similar paradigm.
- *High Performance File System Abstractions* - NASA scientists currently use a large number of existing applications for data analysis, such as GrADS, python scripts, and more, that are not compatible with an object storage environment. If data were stored within an object storage environment, these applications would not be able to access the data. Many of these applications would require a substantial amount of investment to enable them to use object storage file systems. Therefore, a file system abstraction, such as FUSE (file system in user space) is needed to facilitate the use of existing data analysis applications with an object storage environment. The goal is to make a FUSE-like file system abstraction robust, reliable, and highly performing for use with large NASA data sets.
- *Data Management of Large-Scale Scientific Repositories* - With increasing size of scientific repositories comes an increasing demand for using the data in ways that may never have been imagined when the repository was conceived. The goal is to provide capabilities for the flexible repurposing of scientific data, including large-scale data integration, aggregation, representation, and distribution to emerging user communities and applications.
- *Server Side Data Processing* - Large data repositories make it necessary for analytical codes to migrate to where the data are stored. Hadoop does that at the level of a single HDFS. In a densely networked world of geographically distributed repositories, tiered intermediation is needed. The goal is to provide support for migratable codes and analytical outputs as first class objects within a provenance-oriented data management cyberinfrastructure.
- *Techniques for Data Analysis and Visualization* - New methods for data analytics that scale to extremely large data sets are necessary for data mining, searching, fusion, subsetting, discovery, visualization, and more. In addition, new algorithms and methods are needed to look for unknown correlations across large, distributed scientific data sets. The goal is to increase the scientific value of model and observation data by making analysis easier and higher performing. Among others, some of the topics of interest are:
    - Techniques for automated derivation of analysis products such as machine learning for extraction of features in large image datasets (e.g., volcanic thermal measurement, plume measurement, automated flood mapping, disturbance mapping, change detection, etc.).
    - Workflows for automated data processing, interpretation, and distribution.
- *Accelerated Large Scale Data Movement* - There are a multitude of large distributed data stores across NASA that includes both observation and model data. The movement of data across the network must be optimized to take full advantage of large-scale data analytics, especially when comparing model to observation data. The goal is to optimize data movement in the following ways:
    - Accelerate and make it easier to move data over the wide area to facilitate large-scale data management and analysis.
    - Optimize the movement of data within more local environments, such as the usage of Remote Direct Memory Access (RDMA) within HDFS.
    - Virtualization of high-speed network interfaces for use within cloud environments.

Research proposed to this subtopic should demonstrate technical feasibility during Phase I, and in partnership with scientists, show a path toward a Phase II prototype demonstration, with significant communication with missions and programs to ensure a successful Phase III infusion. It is highly desirable that the proposed projects lead to software that is infused into NASA programs and projects.

Tools and products developed under this subtopic may be used for broad public dissemination or within a narrow scientific community. These tools can be plug-ins or enhancements to existing software, on-line data/computing services, or new stand-alone applications or web services, provided that they promote interoperability and use standard protocols, file formats and Application Programming Interfaces (APIs) or prevalent applications.

S5.04 Integrated Science Mission Modeling

Lead Center: GSFC

Participating Center(s): ARC, JPL

NASA seeks innovative systems modeling methods and tools to:

- Define, develop and execute future science missions, many of which are likely to feature designs and operational concepts that will pose significant challenges to existing approaches and applications.
- Enable disciplined system analysis for the ongoing management and decision support of the space science technology portfolio, particularly with regard to understanding technology alternatives, relationships, priorities, timing, availability, down-selection, maturation, investment needs, system engineering considerations, and cost-to-benefit ratios; to examine "what-if" scenarios; and to facilitate multidisciplinary assessment, coordination, and integration of the technology roadmaps as a whole.

Use of System Modeling Language (SysML) is encouraged but not required. SysML is a general purpose graphical modeling language for analyzing, designing and verifying complex systems that may include hardware, software, information, personnel, procedures and facilities. As a language, SysML represents requirements, structure, behavior, and equations in nine different diagram types, and can represent both hardware and software models. The language can be extended to provide metamodels for different disciplines, and is supported by multiple commercial tools. SysML is finding increased use throughout the agency to support systems engineering and analysis.

Specific areas of interest include the following:

- Integration of system and mission modeling tools with high-fidelity multidisciplinary design and modeling tools, supporting efficient analysis methods that accommodate uncertainty, multiple objectives, and large-scale systems - This requires the development of robust interfaces between SysML and other tools, including CAD/CAE/PDM/PLM applications, used to support NASA science mission development, implementation and operations. The objective is to produce a unified environment supporting mixed systems-level and detailed analysis during any lifecycle phase, and rapid analysis of widely varying concepts/configurations using mixed-fidelity models, including geometry/mesh-based models when required. The human interface for such a system could be a "dashboard" (web-based is highly desirable) which initially allows for monitoring of the dataflow across a heterogeneous set of tools and finally allows for control of the data flow between the variety of applications.
- Modeling and rapid integration of programmatic, operational, and risk elements - Fully integrated system model representations must include non-physics based constructs such as cost, schedule, risk, operations, and organizational model elements. Novel methods and tools to model these system attributes are critical. In addition, approaches to integrate these in a meaningful way with other system model elements are needed. Methods that consider the development of these models as by-products of a collaborative and/or concurrent design process are particularly valuable.
- Library of SysML models of NASA related systems - Using a library of SysML models, engineers will be able to design their systems by reusing a set of existing models. Too often, these engineers have to begin from scratch the design of the systems. A library of verified and validated models would provide a way for the engineers to design a new spacecraft by assembling existing models that are domain specific, and

therefore easy to adapt to the target system. In order to provide for seamless integration between SysML models each model must identify its level of abstraction both in terms of the modeling of time (progression: no ordering of events, qualitative ordering of events, metric time ordering of events) and the modeling of space (progression: lumped parameters models, distributed parameter models). Such levels of abstraction "certificates" for SysML will help determine integration interface requirements between any two models.
- Profiles for spacecraft, space robotics, and scientific instruments - Profiles provide a means of tailoring SysML for particular purposes. Extensions of the language can be inserted. This allows an organization to create domain specific constructs which extend existing SysML modeling elements. By developing profiles for NASA domains such as Spacecraft, Space Robotics and Scientific Instruments, powerful mechanisms will be available to NASA systems engineers for designing future space systems.
- Requirements Modeling - SysML offers requirements modeling capabilities, thus providing ways to visualize important requirements relationships. There is a need to combine traditional requirements management, supported by tools including but not limited to DOORS and CRADLE, and SysML requirements modeling in a standardized and sustainable way.
- Functional Modeling - The intermediate data products between requirements and specification are detailed functional models that identify all of the functions required to achieve the mission profile(s). There is a critical need to model this layer as it is a key data product to provide traceability between requirements and implementation.
- Model and Modeling Process Synthesis - As model-based design broadens and integrates larger and more complex models, methods for how to sequence and operate the design synthesis, evaluation (e.g., V&V) and elaboration process will become more important, as will considerations of how model-based processes are made compatible with existing review and development cycles.

S5.05 Fault Management Technologies

Lead Center: MSFC

Participating Center(s): ARC, JPL

As science missions are given increasingly complex goals and have more pressure to reduce operations costs, system autonomy increases. Fault Management (FM) is one of the key components of system autonomy. FM consists of the operational mitigations of spacecraft failures. It is implemented with spacecraft hardware, on-board autonomous software that controls hardware, software, information redundancy, and ground-based software and operations procedures.

Many recent Science Mission Directorate (SMD) missions have encountered major cost overruns and schedule slips during test and verification of FM functions. These overruns are due to a lack of understanding of FM functions early in the mission definition cycles and to FM architectures that do not provide attributes of transparency, verifiability, fault isolation capability, or fault coverage. The NASA FM Handbook is under development to improve the FM design, development, verification and validation and operations processes. FM approaches, architectures, and tools are needed to improve early understanding of needed FM capabilities by project managers and FM engineers and to improve the efficiency of implementing and testing FM.

Specific objectives are to:

- Improve the ability to predict FM system complexity and estimate development and operations costs.
- Enable cost-effective FM design architectures and operations.
- Determine completeness and appropriateness of FM designs and implementations.
- Decrease the labor and time required to develop and test FM models and algorithms.
- Improve visualization of the full FM design across hardware, software, and operations procedures.
- Determine extent of testing required, completeness of verification planned, and residual risk resulting from incomplete coverage.
- Increase data integrity between multi-discipline tools.
- Standardize metrics and calculations across FM, SE, S&MA and operations disciplines.
- Increase reliability of FM systems.

Expected outcomes are better estimation and control of FM complexity and development costs, improved FM designs, and accelerated advancement of FM tools and techniques.

The approach of this subtopic is to seek the right balance between sufficient reliability and cost appropriate to the mission type and risk posture. Successful technology development efforts under this subtopic would be considered for follow-on funding by, and infusion into, SMD missions. Research should be conducted to demonstrate technical feasibility and NASA relevance during Phase I and show a path toward a Phase II prototype demonstration.

Offerors should demonstrate awareness of the state-of-the-art of their proposed technology, and should leverage existing commercial capabilities and research efforts where appropriate.

Specific technology in the forms listed below is needed to increase delivery of high quality FM systems. These approaches, architectures and tools must be consistent with and enable the NASA FM Handbook concepts and processes:

- *FM Design Tools* - System modeling and analyses significantly contributes to the quality of FM design; however, the time it takes to translate system design information into system models often decreases the value of the modeling and analysis results. Examples of enabling techniques and tools are modeling automation, spacecraft modeling libraries, expedited algorithm development, sensor placement analyses, and system model tool integration.
- *FM Visualization Tools* - FM systems incorporate hardware, software, and operations mechanisms. The ability to visualize the full FM system and the contribution of each mechanism to protecting mission functions and assets is critical to assessing the completeness and appropriateness of the FM design to the mission attributes (mission type, risk posture, operations concept, etc.). Fault trees and state transition diagrams are examples of visualization tools that could contribute to visualization of the full FM design.
- *FM Verification and Validation Tools* - As complexity of spacecraft and systems increases, the extensiveness of testing required to verify and validate FM implementations can be resource intensive. Automated test case development, false positive/false negative test tools, model verification and validation tools, and test coverage risk assessments are examples of contributing technologies.
- *FM Design Architectures* - FM capabilities may be implemented through numerous system, hardware, and software architecture solutions. The FM architecture trade space includes options such as embedded in the flight control software or independent onboard software; on board versus ground-based capabilities; centralized or distributed FM functions; sensor suite implications; integration of multiple FM techniques; innovative software FM architectures implemented on flight processors or on Field Programmable Gate Arrays (FPGAs); and execution in real-time or off-line analysis post-operations. Alternative architecture choices could help control FM system complexity and cost and could offer solutions to transparency, verifiability, and completeness challenges.
- *Multi-discipline FM Interoperation* - FM designers, Systems Engineering, Safety and Mission Assurance, and Operations perform analyses and assessments of reliabilities, failure modes and effects, sensor coverage, failure probabilities, anomaly detection and response, contingency operations, etc. The relationships between multi-discipline data and analyses are inconsistent and misinterpreted. Resources are expended either in effort to resolve disconnects in data and analyses or worse, reduced mission success due to failure modes that were overlooked. Solutions that address data integrity, identification of metrics, and standardization of data products, techniques and analyses will reduce cost and failures.