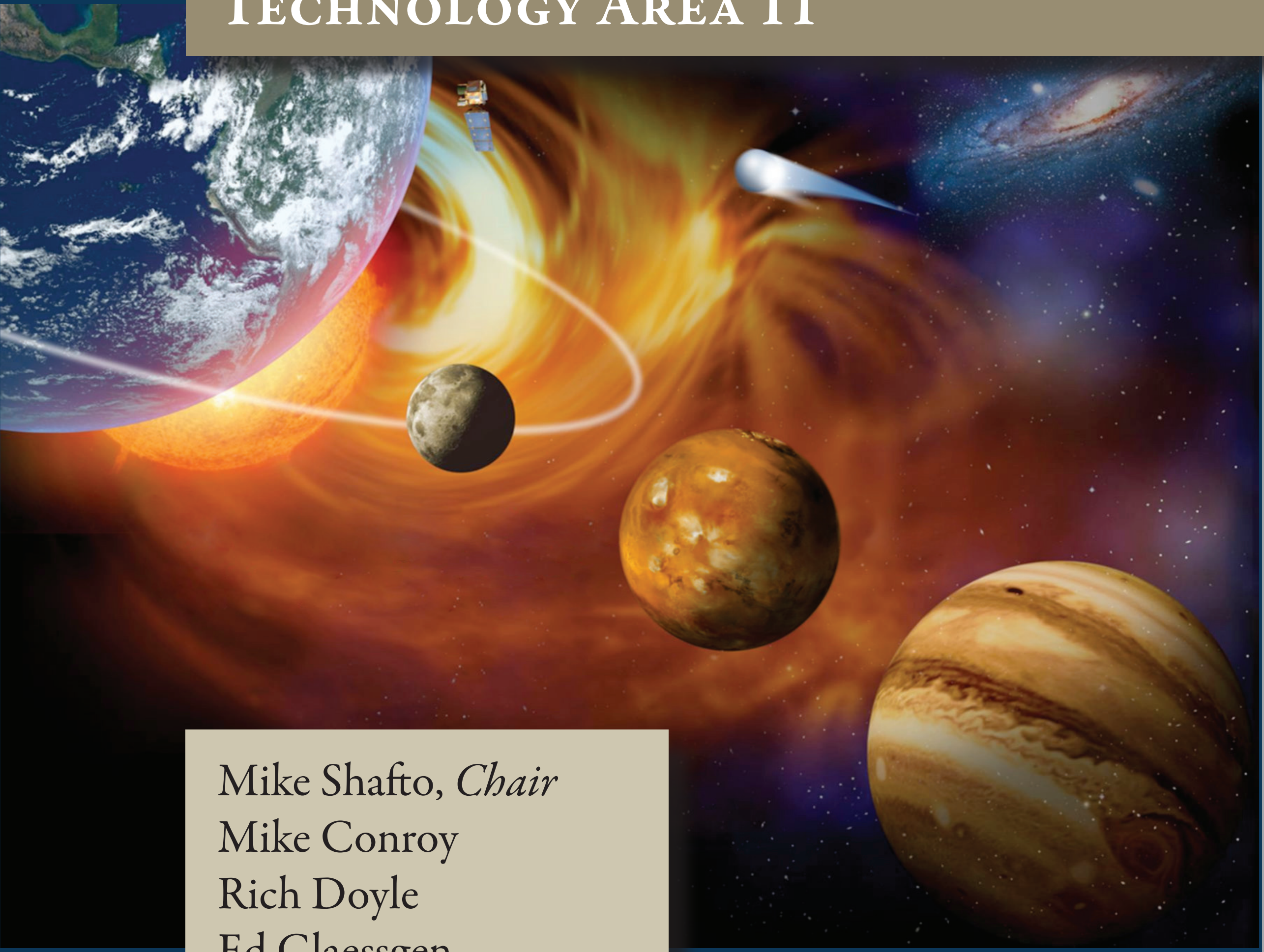
National Aeronautics and Space Administration

DRAFT MODELING, SIMULATION, INFORMATION TECHNOLOGY & PROCESSING ROADMAP

TECHNOLOGY AREA 11



Mike Shafto, *Chair*
Mike Conroy
Rich Doyle
Ed Glaessgen
Chris Kemp
Jacqueline LeMoigne
Lui Wang

November • 2010

This page is intentionally left blank

Table of Contents

FOREWORD
EXECUTIVE SUMMARY TA11-1
1. GENERAL OVERVIEW TA11-2
1.1. Technical Approach TA11-2
1.2. Benefits TA11-2
1.3. Applicability/Traceability to NASA Strategic Goals TA11-2
1.4. Top Technical Challenges TA11-5
2. DETAILED PORTFOLIO DISCUSSION TA11-7
2.1. Summary description and Technical Area Breakdown Structure (TABS) TA11-7
2.2. Description of each column in the TABS TA11-8
2.2.1. Computing TA11-8
2.2.1.1. Flight Computing TA11-8
2.2.1.2. Ground Computing TA11-11
2.2.2. Modeling TA11-11
2.2.2.1. Software Modeling and Model-Checking TA11-11
2.2.2.2. Integrated Hardware and Software Modeling TA11-12
2.2.2.3. Human-System Performance Modeling TA11-12
2.2.2.4. Science and Aerospace Engineering Modeling TA11-13
2.2.2.5. Frameworks, Languages, Tools, and Standards TA11-16
2.2.3. Simulation TA11-17
2.2.3.1. Distributed Simulation TA11-17
2.2.3.2. Integrated System Lifecycle Simulation TA11-17
2.2.3.3. Simulation-Based Systems Engineering TA11-18
2.2.3.4. Simulation-Based Training and Decision Support TA11-19
2.2.4. Information Processing TA11-19
2.2.4.1. Science, Engineering, and Mission Data Lifecycle TA11-19
2.2.4.2. Intelligent Data Understanding TA11-19
2.2.4.3. Semantic Technologies TA11-21
2.2.4.4. Collaborative Science and Engineering TA11-21
2.2.4.5. Advanced Mission Systems TA11-22
3. INTERDEPENDENCY WITH OTHER TECHNOLOGY AREAS TA11-25
4. POSSIBLE BENEFITS TO OTHER NATIONAL NEEDS TA11-25
ACRONYMS TA11-26
ACKNOWLEDGEMENTS TA11-27
REFERENCES TA11-27

FOREWORD

NASA's integrated technology roadmap, including both technology pull and technology push strategies, considers a wide range of pathways to advance the nation's current capabilities. The present state of this effort is documented in NASA's DRAFT Space Technology Roadmap, an integrated set of fourteen technology area roadmaps, recommending the overall technology investment strategy and prioritization of NASA's space technology activities. This document presents the DRAFT Technology Area 11 input: Modeling, Simulation, Information Technology & Processing. NASA developed this DRAFT Space Technology Roadmap for use by the National Research Council (NRC) as an initial point of departure. Through an open process of community engagement, the NRC will gather input, integrate it within the Space Technology Roadmap and provide NASA with recommendations on potential future technology investments. Because it is difficult to predict the wide range of future advances possible in these areas, NASA plans updates to its integrated technology roadmap on a regular basis.

EXECUTIVE SUMMARY

Figure 1 is an initial draft strategic roadmap for the Modeling, Simulation, Information Technology and Processing Technology Area. Although highly notional, it gives some idea of how the developments in Computing, Modeling, Simulation, and Information Processing might evolve over the next 10-20 years. In spite of the breadth and diversity of the topic, a small number of recurring themes unifies the requirements of aerospace engineering, remote science, current space operations, and future human exploration missions. There is a surprising consensus regarding the research needed to realize the benefits of simulation-based science and engineering, and of emerging paradigms in information technology.

For our purposes, **Computing** covers innovative approaches to flight and ground computing that have the potential to increase the robustness of future aerospace systems and the science return on long-duration exploration missions. **Flight Computing** requires ultra-reliable, radiation-hardened platforms which, until recently, have been costly and limited in performance. **Ground Computing** requires supercomputing or other high-performance platforms to support petabyte-scale data analysis and multi-scale physical simulations of systems and vehicles in unique space environments. In both multi-core flight computing and high-end ground computing, performance is constrained by complex dependencies among hardware, algorithms, and problem structure. We assume that fundamental advances in the Computing area will benefit modeling, simulation, and information processing.

Modeling, simulation and decision making are closely coupled and have become core technologies in science and engineering. In the simplest sense, a model represents the characteristics of something, while a simulation represents its behavior. Through the combination of the two, we can make better decisions and communicate those decisions early enough in the design and development process that changes are easy and quick, as opposed to during production when they are extremely costly and practically impossible.

Modeling covers several approaches to the integration of heterogeneous models to meet challenges ranging from Earth-systems and Heliophysics modeling to autonomous aerospace vehicles. The main topics of this section are **Software Modeling**, **Integrated Hardware and Software Modeling**, and large-scale, high-fidelity **Science and Aerospace Engineering Modeling**, for example, Earth-system (climate and weather) modeling or modeling the performance of an autonomous Europa lander.

Simulation focuses on the architectural challenges of NASA's distributed, heterogeneous, and long-lived mission systems. Simulation represents behavior. It allows us to better understand how things work on their own and together. Simulation allows us to see and understand how systems might behave very early in their lifecycle, without the risk or expense of actually having to build the system and transport it to the relevant environment. It also allows us to examine existing large, complex, expensive or dangerous systems without the need to have the physical system in our possession. This is especially attractive when the system in question is on orbit, or on another planet and impossible to see or share with others; or when the system does not yet exist, as in the case of advanced autonomous vehicles.

Information Processing includes the system-level technologies needed to enable large-scale remote science missions, automated and reconfigurable exploration systems, and long-duration human exploration missions. Given its scope, potential impact, and broad set of implementation issues, simulation provides many of the challenges faced in advanced information processing systems. There are software engineering challenges related to simulation reuse, composition, testing, and human interaction. There are systems questions related to innovative computing platforms. There are graphics and HCI questions related to the interface with users of varying disciplines and levels of expertise. In NASA's science and engineering domains, data-analysis, modeling, and simulation requirements quickly saturate the best available COTS or special-purpose platforms. Multi-resolution, multi-physics, adaptive and hybrid systems modeling raise challenges even for theoretical computer science. At the most practical level, separation of concerns becomes critical when systems and services are no longer locally provisioned and coupled to local implementations. For example, systems that decouple storage and data processing, as well as data storage and data management, are more likely to take early advantage of elastic technology paradigms like cloud: There is more opportunity to leverage the lowest common denominator services that cloud provides, such as compute and store, without navigating other large complex subsystems independent of pay-for-play cloud services.

1. GENERAL OVERVIEW

A mission team's ability to plan, act, react and generally accomplish science and exploration objectives resides partly in the minds and skills of engineers, crew, and operators, and partly in the flight and ground software and computing platforms that realize the vision and intent of those engineers, crew and operators.

Challenges to NASA missions increasingly concern operating in remote and imperfectly understood environments, and ensuring crew safety in the context of unprecedented human spaceflight mission concepts. Success turns on being able to predict the fine details of the remote environment, possibly including interactions among humans and robots. A project team needs to have thought through carefully what may go wrong, and have a contingency plan ready to go, or a generalized response that will secure the crew, spacecraft, and mission.

Balanced against these engineering realities are always-advancing science and exploration objectives, as each mission returns exciting new results and discoveries, leading naturally to new science and exploration questions that in turn demand greater capabilities from future crew, spacecraft and missions.

This healthy tension between engineering and system designs on the one hand, and science investigations and exploration objectives on the other results in increasing demands on the functionality of mission software and the performance of the computers that host the on-board software and analyze the rapidly growing volume of science data. Mission software and computing must become more sophisticated to meet the needs of the missions, while extreme reliability and safety must be preserved. Mission software and computing also are inherently cross-cutting, in that capabilities developed for one mission are often relevant to other missions as well, particularly those within the same class.

1.1. Technical Approach

Computing covers innovative approaches to flight and ground computing with the potential to increase robustness and science-return on long-duration missions. Innovative computing architectures are required for integrated multi-scale data-analysis and modeling in support of both science and engineering. **Modeling** covers several approaches to the integration of heterogeneous models to meet challenges ranging from Earth-systems modeling to NextGen air traffic management. **Simulation** focuses on the design, planning, and operational challenges of NASA's distributed, long-lived mission systems. And finally, **Information Processing** includes the technologies needed to enable large-scale remote science missions, automated and reconfigurable air-traffic management systems, and long-duration human exploration missions.

In spite of the breadth and diversity of the topic, a small number of recurring themes unifies the requirements of aerospace engineering, remote science, current space operations, and future human exploration missions. Modeling, simulation and decision making are closely coupled and have become core technologies in science and engineering (McCafferty, 2010; Merali, 2010; NRC, 2010). In the simplest sense, a model represents the characteristics of something, while a simulation represents its behavior. Through the combination of the two, we can make better decisions and communicate those decisions early enough in the design and development process that changes are easy and quick, as opposed to during production when they are extremely costly and practically impossible. We can also discover complex causal relations among natural processes occurring across vast spatiotemporal scales, involving previously unknown causal relations.

1.2. Benefits

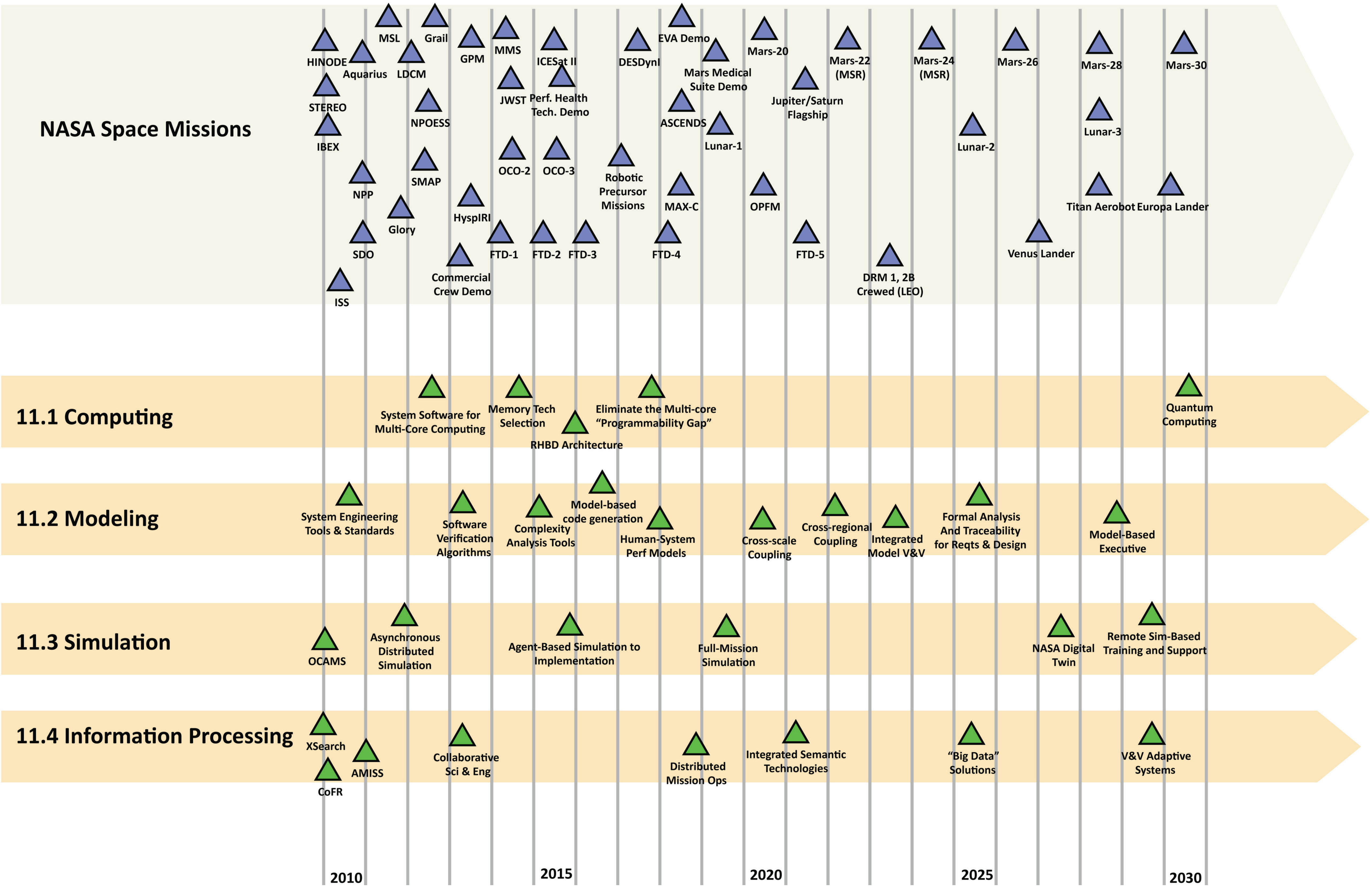
Some of the major benefits of high-fidelity modeling and simulation, supported by high-performance computing and information processing, include creativity (creative problem solving): experimentation; what-if analysis; insight into constraints and causality; large-scale data analysis and integration, enabling new scientific discoveries; training and decision-support systems that amortize costs by using modeling and simulation directly in mission systems; simplification and ease of operation; testing and evaluation of complex systems early in the lifecycle; high-reliability real-time control; mission design optimization, trade studies, prediction; analysis of complex systems; risk reduction; reduced mission design-cycle time; and lower lifecycle cost.

1.3. Applicability/Traceability to NASA Strategic Goals

Aerospace Engineering

NASA's expansive and long-term vision for aerospace engineering is heavily dependent on numerous developments and breakthroughs in MSITP including high-fidelity multi-scale physical modeling, data access and analysis, and improved HEC

Figure 1: *Modeling, Simulation, Information Technology and Processing Technology Area Strategic Roadmap (TASR).*



resources. These developments support goals such as developing conflict alert capability for terminal operations (2013); integration of high-fidelity hypersonic design tools (2015); improving overall vehicle safety (2016); developing technologies to enable fuel burn reductions of 33% (2016), 50% (2020) and 70% (2026) for subsonic fixed wing vehicle; and demonstrating a 71dB cumulative noise level below Stage 4 for subsonic vehicle (2028).

Space and Earth Science

Future large-scale deep space missions following the Mars Science Laboratory (MSL; 2011) will require model-based systems engineering to control cost and schedule variations (MAX-C Rover, 2018; Outer Planets Flagship Mission – OPFM, 2020; Mars Sample Return – MSR, 2020+). These same model-based techniques may find additional uses throughout the lifecycle to achieve system reliability in challenging operational environments. Other concepts for future planetary missions, e.g., a Venus Lander, a Titan Aerobot, a Europa Lander, can be expected to push further on needs for greater onboard autonomy and flight computing.

Future missions like the James Webb Space Telescope (JWST; 2014) will require high-fidelity integrated multi-scale physical modeling to anticipate and mitigate design issues. NASA's increasingly integrative and productive Heliophysics and Earth Science missions [Aquarius (2010), NPP (2010), SDO (2010), LDCM (2011), SMAP (2012), NPOESS (2012), GPM (2013), HyspIRI (2013), MMS (2014), ICESat II (2015), DESDynI (2017), ASCENDS (2018)] will require major innovations in data management and high-performance computing, including onboard data analysis.

Space Operations

ISS operations are already incorporating advanced near-real-time data mining and semantically oriented technologies for operational data integration and intelligent search. Model-based approaches to operations automation have also been demonstrated and are operational today in the ISS MCC. These advanced technologies can be extended to support Commercial Cargo and Commercial Crew Demo Flight (2014+). They will reduce staffing requirements and costs, as well as enhancing software maintainability, extension, and reuse. Technical challenges remain, however, in scaling up agent-based architectures and ensuring their flexibility and reliability.

Human Exploration

Agent-based simulation-to-implementation concepts have been successfully demonstrated in field studies and in ISS operations, showing how such technologies can be systematically implemented to support Human Research such as: Biomed Tech Demo (2012+); Performance Health Tech Demo (2014); and Mars Medical Suite Demo (2018). Integrated software-hardware modeling will benefit Closed-Loop ECLSS (2014); EVA Demo (2018); and Nuclear Thermal Propulsion (2020). Simulation-based systems engineering is needed in planning and developing Flagship Technology Demonstrations, such as Advanced In-Space Propulsion Storage & Transfer (2015) and Advanced ECLSS on ISS (2017). Modeling and simulation, as well as high-performance computing, have been identified as key elements in planning and developing Exploration Precursor Robotic Missions (2014-2019).

1.4. Top Technical Challenges

Technical Challenges in Priority Order	
1.	Advanced Mission Systems (TABS 4.5): Adaptive Systems
2.	Integrated System Lifecycle Simulation (TABS 3.2): Full Mission Simulation
3.	Simulation-Based Systems Engineering (TABS 3.3): NASA Digital Twin
4.	Software Modeling (TABS 2.1): Formal analysis and traceability of requirements and design
5.	Integrated Hardware and Software Modeling (TABS 2.2): Advanced Integrated Model V&V
6.	Modeling (TABS 2.4): Cross-scale and inter-regional coupling
7.	Flight Computing (TABS 1.1): System Software for Multi-Core Computing
8.	Integrated Hardware and Software Modeling (TABS 2.2): Complexity Analysis Tools
9.	Flight and Ground Computing (TABS 1.1 and 1.2): Eliminate the Multi-core "Programmability Gap"
10.	Software Modeling (TABS 2.1): Software Verification Algorithms

Flight and Ground Computing (TABS 1.1 and 1.2): Eliminate the Multi-core "Programmability Gap": For all current programming models, problem analysis and programming choices have a direct impact on the performance of the resulting code. This leads to an incremental, costly approach to developing parallel distributed-memory programs. What has emerged today is a "Programmability Gap", the mismatch between traditional sequential software development and today's multi-core and accelerated computing environments. New parallel languages break with the conventional programming paradigm by offering high-level abstractions for control and data distribution, thus providing direct support for the

Technical Challenges in Chronological Order	
2010-2016	Flight Computing (TABS 1.1): System Software for Multi-Core Computing
	Software Modeling (TABS 2.1): Software Verification Algorithms
	Integrated Hardware and Software Modeling (TABS 2.2): Complexity Analysis Tools
	Flight and Ground Computing (TABS 1.1 and 1.2): Eliminate the Multi-core “Programmability Gap”
	Integrated System Lifecycle Simulation (TABS 3.2): Full Mission Simulation
2017-2022	Science Modeling (TABS 2.4): Cross-scale and inter-regional coupling
	Integrated Hardware and Software Modeling (TABS 2.2): Advanced Integrated Model V&V
	Software Modeling (TABS 2.1): Formal analysis and traceability of requirements and design
2023-2028	Simulation-Based Systems Engineering (TABS 3.3): NASA Digital Twin
	Advanced Mission Systems (TABS 4.5): Adaptive Systems

specification of efficient parallel algorithms for multi-level system hierarchies based on multicore architectures. There is a need to experiment with new languages such as Chapel, X-10 and Haskell and to determine their utility for a broad range of NASA applications. Virtual machine (VM) approaches and architecture-aware compiler technology should also be developed.

Flight Computing (TABS 1.1): System Software for Multi-Core Computing: The first multi-core computing chips for space applications are emerging and the time is right to develop architecturally sound concepts for multi-core system software that can use the unique attributes of multi-core computing: software-based approaches to fault tolerance that complement radiation hardening strategies, energy management approaches that account for the energy costs of data movement, and programmability and compiler support for exploiting parallelism.

Software Modeling (TABS 2.1): Formal analysis and traceability of requirements and design: Formal analysis and related tools have been effective in reducing flight software defects for missions like MSL. Current tools, however, focus on the coding and testing phases. No comparable tools have been shown to be effective in the requirements and design phases, where the most costly defects are introduced. Manual methods in these phases, based on unstructured natural language, are almost always vague, incomplete, untraceable, and even self-contradictory. These need to be replaced with methods based on formal semantics, so that checking and traceability can be partially automated.

Software Modeling (TABS 2.1): Software Verification Algorithms: Software verification and validation are critical for flight and ground system reliability and encompass 70% of overall software costs. For human space missions, software costs are roughly equal to hardware design and development costs – and software costs and risks are expected to grow as increasing number of system functions are implemented in software. To support future human space missions, we need to develop automated methods that increase the assurance of these software systems while greatly decreasing costs. Advanced model-based methods may be able to address these issues, but this will require significant scaling up of current methods to address verification of automated operations software, verification of system health management software, verification for adaptive avionics and automated test-generation for system software validation.

Integrated Hardware and Software Modeling (TABS 2.2): Complexity analysis tools: Complexity analysis tools should be developed for use in concept development and requirements definition. This would allow system engineering and analyses to be done early enough and to be incorporated into Pre-Phase-A design centers, into mission-costing models, and into various trade-space evaluation processes. Ideally, these tools would link with design, implementation, and test tools. Finally, process templates should be developed that build on this new class of tools.

Integrated Hardware and Software Modeling (TABS 2.2): Advanced Integrated Model V&V: Massively parallel models of key spacecraft modules are needed for integration into Sierra or similar high-fidelity frameworks. These include models of optics, space environments, and control systems. New methods and tools are needed for integrated model V&V, including rigorous methods for selecting the most cost-effective test sequences. Component and subsystem test methods for spacecraft-unique issues are needed, along with advanced workflow and rapid model-prototyping technologies. Orders-of-magnitude improvements in accuracy will be required in order to gain project acceptance and meet the goal of application to at least one Flagship NASA mission in parallel with conventional practices.

Science Modeling (TABS 2.4): Cross-scale and inter-regional coupling: Cradle-to-grave modeling of a solar eruption, from its initiation to its impact on terrestrial, lunar, or spacecraft environments, requires the resolution of each sub-domain and the coupling of sub-domains across

their boundaries. Calculations of this kind entail not only the requirements for modeling of each domain, but the added complications of information transfer and execution synchronization. The often vastly different time and spatial scales render the combined problem intractable with current technology. A minimum computational requirement for such forefront calculation involves the sum of all coupled domain calculations (optimistically assuming the calculations can be pipelined). Similar challenges exist in Earth-systems modeling, for example, climate-weather modeling, and for multi-scale, multi-physics engineering models.

Integrated System Lifecycle Simulation (TABS 3.2): Full-Mission Simulation: Perform complete mission lifecycle analysis, planning and operations utilizing best available system, subsystem and supporting models and simulations, both hardware and software. The mission lifecycle includes: design and development of mission elements, any supporting infrastructure, any supporting integration work, any additional opportunities, launch operations, fight operations, mission operations, anomaly resolution, asset management, return operations, recovery operations and system/data disposition.

Simulation-Based Systems Engineering (TABS 3.3): NASA Digital Twin: A digital twin is an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its flying twin. The digital twin is ultra-realistic and may consider one or more important and interdependent vehicle systems, including propulsion/energy storage, avionics, life support, vehicle structure, thermal management/TPS, etc. In addition to the backbone of high-fidelity physical models, the digital twin integrates sensor data from the vehicle's on-board integrated vehicle health management (IVHM) system, maintenance history, and all available historical/fleet data obtained using data mining and text mining. The systems on board the digital twin are also capable of mitigating damage or degradation by recommending changes in mission profile to increase both the life span and the probability of mission success.

Advanced Mission Systems (TABS 4.5): Adaptive Systems: NASA missions are inherently ventures into the unknown. The agency appears poised on the cusp of a trend where science and exploration missions must first undergo a characterization phase in the remote environment before safe and effective operations can begin. As an example, any NEO mission concept likely requires an initial gravitational characterization phase from a parking orbit to determine the gradients of the body's gravitational field before landing or touch-and-go operations are attempted. Will it be possible to send spacecraft and human-robotic teams into these environments to accomplish characterization prior to actual operations? What would be the role of onboard capabilities for intelligent systems, data analysis, modeling, and possibly machine learning? An adaptive-systems approach could revolutionize how missions are conceived and carried out, and would extend NASA's reach for scientific investigation and exploration much farther out into the solar system.

2. DETAILED PORTFOLIO DISCUSSION

2.1. Summary description and Technical Area Breakdown Structure (TABS)

The topic area of **Modeling, Simulation, Information Technology and Processing** is inherently broad (see Figure 2). Software and computing in one form or another touches all of the other roadmap topic areas. While those roadmap efforts are addressing needs from specific domain perspectives, this TA focuses on needed advances in underlying foundational capabilities for software and computing, and how they are to support increasing demands for modeling and simulation, as well as needs for science and engineering information processing.

In the area of Computing, there are needs for flight computing architectures that scale, separate data processing from control functions, and include intelligent fault-tolerance technology. Advances in ground computing are needed to meet emerging requirements for modeling, simulation, and analysis. There is also a need to close the performance and architectural gaps between ground and flight computing, in order to increase capability on the flight side and enable the migration of functions from ground to flight systems.

Research in Modeling will address the need for coherent frameworks that have the potential to reduce ambiguity, uncertainty, and error. Efficiency of model development is also an important theme. Coherence and efficiency are mutually reinforcing, focusing attention on composability, transparency, understandability, modularity, and interoperability of models and simulations.

Effective and accurate Simulation in turn depends on integrable families of models that cap-

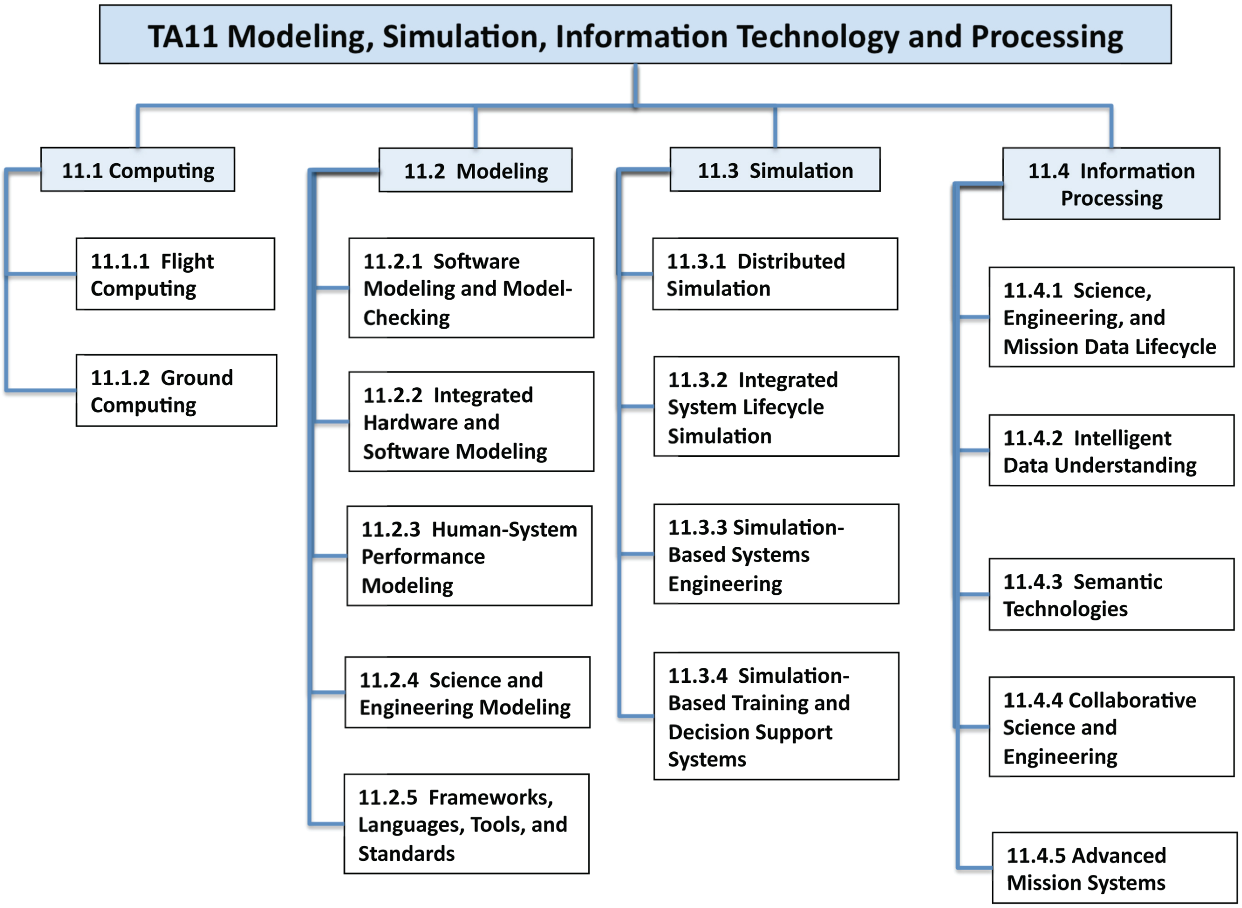


Figure 2. *Modeling, Simulation, Information Technology and Processing Technology Area Breakdown Structure (TABS).*

ture different aspects of phenomena and different scales of resolution. The variability of parameters in simulation exercises is currently represented by probability distributions, but uncertainty due to lack of knowledge is difficult to quantify. Uncertainty and risk must be addressed if simulation is to be used intelligently in decision-support, and these issues are equally important in science applications.

The requirements for high-fidelity simulation open the door to broader challenges in Information Processing. There are software engineering challenges related to simulation reuse, composition, testing, and human interaction. There are systems questions related to innovative computing platforms. There are graphics and HCI questions related to the interface with users of varying disciplines and levels of expertise. In NASA's science and engineering domains, data-analysis, modeling, and simulation requirements quickly saturate the best available supercomputing platforms. Multi-resolution, multi-physics, adaptive and hybrid systems modeling raise challenges even for theoretical computer science.

2.2. Description of each column in the TABS

2.2.1. Computing

The computing roadmap is summarized in Table 1.

2.2.1.1. Flight Computing

Pinpoint landing, hazard avoidance, rendezvous-and-capture, and surface mobility are directly tied to the availability of high-performance space-based computing. In addition, multi-core architectures have significant potential to implement scalable computing, thereby lowering spacecraft vehicle mass and power by reducing the number of dedicated systems needed to implement onboard functions. These requirements are equally important to space science and human exploration missions.

Intelligent adaptive capabilities will often be tightly integrated with spacecraft command and control functions. Onboard computers must

Table 1. *Computing Roadmap*

Milestone	Relevance	Timeframe
Low power, high performance flight computing capability (e.g., multi-core) available	Flight computing capability of 10 GOPS @ 10 W for computationally intensive needs (high data rate instruments)	**2013**
Develop and demonstrate multi-core fault mitigation methods and integrate with avionics FDIR methods	Combination of radiation hardening, RHBD, and software-based approaches to fault tolerance to achieve availability similar to existing flight processors	**2014**
Demonstrate capability to move cores on and off line and continue computation without loss of data	Programmers freed from managing details of multi-core management and resource allocation	**2015**
Demonstrate a policy based approach for energy management at runtime, complemented by energy-aware compilers	Low energy flight computation handled by compilers and runtime environment with options for programmer optimization	**2016**
Demonstration of flight use of multi-core processing	Onboard data product generation for high date rate instruments and/or real-time vision-based processing for e.g., terrain-relative navigation	**2020**
Multi-core flight processing enables new types of missions	Onboard model-based reasoning (e.g., mission planning, fault management) for surface, small body or similar missions requiring a high degree of adaptability in an uncertain operational environment	**2025**
Highly capable flight computing enables reconfigurable spacecraft and versatile missions	Onboard computation directly supports missions requiring a characterization phase at the target (e.g., small body proximity ops, atmospheric mobility) to refine environmental models, operations concepts, and spacecraft configuration and mission design as needed	**2030**

therefore demonstrate high reliability and availability. In systems deployed beyond low-earth orbit, robust operation must be maintained despite radiation and thermal environments harsher than those encountered terrestrially or in low earth orbit.

COTS platforms typically out-perform radiation-hardened space platforms by an order of magnitude or more. Systems based on commercial processors that mitigate environment-induced faults with temporal and spatial redundancy (e.g., triplication and voting) could be viable for space-based use. The commercial sector is basing current and future high performance systems on multi-core platforms because of their power and system-level efficiencies. The same benefits are applicable to spaceflight systems. In addition, multi-core enables numerous system architecture options not feasible with single-core processors.

Status: There are two promising processor development activities that are currently undergoing evaluation for their suitability for use in NASA's programs: the HyperX and the Maestro.

The **HyperX** (Hx) development involves collaboration between NASA's Exploration Technology Development Program (ETDP) and NASA's Innovative Partnership Program. Test results formed the basis for single-event radiation mitigation strategies that were developed in 2009. To validate the radiation mitigation techniques, four HyperX processor boards are being flown as a part of the Materials International Space Station Experiment-7 (MISSE-7). As of May 14, 2010, the boards had executed over 140,000 iterations with no faults.

The second processor development activity of interest to NASA is the assessment of the **Maestro** Processor. This effort began as an initiative to develop a processor that performs with a high level of efficiency across all categories of application processing, ranging from bit-level stream processing to symbolic processing, and encompassing processor capabilities ranging from special purpose digital signal processors to general-purpose processors. A radiation hardened version of the Tile-64, a commercial processor, Maestro was developed in a collaboration among several government agencies. The throughput for this device is consistent with NASA needs and metrics. Its power dissipation, however, compromises its desirability for use in numerous spaceflight systems. Accordingly, one of NASA's objectives for future space-based computing should be to leverage investments in Maestro or other processor platforms to deliver a system with throughput comparable to Maestro, but at lower power dissipation.

Future Directions: Whereas radiation tolerant COTS-based boards that offer increased performance are available (~1000 MIPS), the performance is offered at the expense of reduced power efficiency. NASA should seek to concurrently advance the state of the art of two metrics (sustained throughput and processing efficiency) of high-performance radiation-hardened processors. Goals are throughput greater than 2000 MIPS with efficiency better than 500 MIPS/W.

The need for power-efficient high-performance radiation-tolerant processors and the peripheral electronics required to implement functional systems is not unique to NASA; this capability could also benefit commercial aerospace entities and other governmental agencies that require high-capability spaceflight systems. NASA should therefore leverage to the extent practical relevant ex-

ternal technology- and processor-development projects sponsored by other organizations and agencies. Although the primary discriminating factors — sustained processor performance, power efficiency, and radiation tolerance — are among the key metrics, an equally important factor is the availability of tools to support the software development flow. The NASA and larger aerospace community will not accept a very capable flight computing capability if the associated tool set and other support for software development and reliability is considered inadequate.

Fault Tolerance and Fault Management: There is a need to develop a hierarchical suite of fault mitigation methods that takes advantage of the differences and strengths of multi-core processor architectures and lays the groundwork to be integrated with mainstream spacecraft Fault Detection Isolation and Response (FDIR) methods. With large numbers of cores, one imagines it should be possible to reduce power to the system when the number of cores needed is lower. Conversely, when the need arises, it should be possible to increase the number of powered cores to absorb increased load. The actual power savings realizable is a design consideration; for example, degree of emphasis on optimizing power dissipation.

The ability to manage cores for energy considerations is related to the ability to re-assign computations to support fault tolerance. More specifically, fault tolerance considerations may identify the need to take certain faulted cores off-line. Energy management considerations may identify the need to reduce the number of active (powered) cores. When the goal is to manage overall power load, there would be no particular constraint on the specific cores to be powered down. On the other hand, when the goal is to manage energy expenditure efficiently for a given computation, choices about data movement may be the driver, indicating a need to coordinate among specific cores based on their locality. Thus both fault tolerance and energy management considerations may introduce objectives for allocating / de-allocating certain cores; sometimes these objectives will be aligned, and sometimes they may be competing. What is ultimately needed is an overall core management capability and set of strategies that are responsive to both fault tolerance and energy management drivers. Computing systems do not currently have all of these controls, but multi-core computing will need them. Computational throughput is a capability that can be modulated for the contingencies of fault tolerance, and for a given energy cost.

Programmability: Another important objective for multi-core space computing is to reduce software development time while managing complexity and enhancing testability of the overall flight code base. Programs on spacecraft are enormously varied. Spacecraft run fixed- and variable-rate estimation and control loops with both soft and hard real-time requirements, perform power switching, collect large amounts of data in files, send telemetry to the ground, process commands from the ground, conduct science data processing, monitor board and peripheral hardware, and perform a host of other functions. There is every reason to expect that the demand will only increase.

The cost of programming multi-core processors can be high; the prevailing method is to hand-code algorithms tailored to the problem and the processor. (See also Section 2.2.1.2.) NASA cannot afford this approach, and it is also inconsistent if fault tolerance is a goal, since the core allocations for handcrafted algorithms carefully laid out by the programmer would likely be compromised when a core is taken off-line. Energy-management considerations will also disrupt pre-planned allocations of cores. For both fault tolerant behavior and energy management efficiency, the system should be able to re-organize and adapt to reduced core availability. Some choices can be made at compile time, transparently to the programmer, but a robust flight computing system must support run-time re-allocation of cores. The issues for fault tolerance and energy management are similar, and we should expect the mechanisms developed for the one to be applicable to the other.

Energy Management: The systems and software engineer should be provided with a set of models and tools to minimize energy usage in a multi-core flight processor environment throughout the lifecycle. Multi-core energy management spans software code architectural definition, development, and execution, i.e., the run-time environment. Early work in multi-core architecture analysis has shown that as chip density increases and the number of cores scales up, the energy penalty price paid for moving data to and from the core to memory and I/O begins to dominate. This analysis implies that the importance of data locality becomes much more critical. This analysis further suggests that operating on data in a core's local register set or local L1/L2 cache will achieve a higher level of energy/performance efficiency than from data residing in main memory. From a software-coding viewpoint, a low energy design approach would entail minimizing aggregate data movement across high energy cost centers in the chip. High fidelity chip energy models can provide a guide to the programmer on the appropriate trade space that can be explored in optimizing the energy use. Additional support for the programmer would also come from energy-aware compilers.

2.2.1.2. Ground Computing

Supercomputing: Parallelization of numerical solvers on multi-processor architectures has been an active research area for decades. The over-arching issues driving current research concern the complex interdependencies among hardware, software, and problem structure. For all current programming models, the problem analysis and programming choices have a direct impact on the performance of the resulting code. Current approaches attempt to balance distributed and shared memory-models, parallel performance benefits and ease of use. Languages such as UPC, Fortran, and Titanium require programmers specify the distribution of data, while allowing them to use a global name space to specify data accesses. These languages provide an incremental approach to developing parallel distributed-memory programs.

Future Directions: PGAS models, such as UPC, bridge the gap between the shared-memory programming approach and the distributed-memory approach. The UPC model offers performance improvement through an iterative refinement process. However, in order to gain performance, the programmer must pay attention to where data is located and must minimize remote data access. This is nontrivial for most codes. (See also Section 2.2.1.1.) While the UPC language provides constructs to achieve this goal, it would be preferable for the compiler to apply aggressive communications-reducing optimizations automatically. However, this seems to be beyond the capabilities of current compilers (Jin et al., 2009)

Quantum computing: There are now around a dozen alternative approaches to implementing quantum computers. Major corporations have research groups working on quantum computing. In addition to progress on quantum algorithms there have been parallel improvements in quantum computing hardware. Most importantly, it is now known that quantum error correction is possible in principle and so it is not necessary to achieve perfection in quantum hardware. Secondly, there are many different models of quantum computation which are all equivalent to one another but may be easier or harder to implement in different types of physical hardware. Advances have been made on specialized quantum computing devices, such as the quantum annealing machine developed with the assistance of JPL. This has control of around 72 functional qubits, and has been used to solve combinatorial optimization problems arising in computer vision and machine learning.

At this stage quantum computing is still a research project with unclear eventual payoff for NASA. Nevertheless, it represents the only potentially radical advance in computer science in over 100 years. Quantum algorithms are predicted to solve certain computational problems exponentially faster than is possible classically, and can solve many more problems polynomially faster. Rudimentary quantum computers now exist and have been used to solve actual problems. More sophisticated special-purpose quantum computing hardware is on the verge of surpassing the capabilities of mid-range classical computers on certain optimization problems.

2.2.2. Modeling

2.2.2.1. Software Modeling and Model-Checking

Status: Flight software defect prevention, detection, and removal are currently accomplished by a combination of methods applied at each phase of software development: requirements, design, coding, and testing. Most test methods focus on the coding and testing phases, but the most serious defects are inserted in the requirements and design phases. (This is true of systems development in general, not just software.) Automated and semi-automated (tool-supported) processes in all phases are at various TRLs. These include requirements capture and analysis, design verification techniques, logic model checking, static source code analysis, coding standards, code review processes, model-driven verification, runtime monitoring, fault containment strategies, and runtime verification methods.

A good example of the state-of-the-art in tool-based methods is the scrub system developed and used by Gerard Holzmann's group at JPL. scrub uses a broad spectrum of techniques, including automatic tracking of all peer- and tool-reported issues, as well as integration of peer-review comments and tool-generated automated analyses.

Future Directions: Although current methods have demonstrated substantial and quantifiable benefits on real flight software, there is widespread agreement among experts that many serious challenges remain. The following list is based in part on Tallant et al. (2004), Welch et al. (2007), Parnas (2010), and Holzmann (personal communication, 2010).

Near-term: Auto-coding methods; embedded run-time auto-testing; rapid prototyping tools and frameworks; relational algebra and game theoretic semantics; mathematical documentation of code; automated verification management; simulation-based design

Mid-term: Formal requirements specifications, using predicates on observable behavior; formal methods for traceability analysis; linear logic & other theorem proving technologies; formalization of arrays, pointers, hidden-state, side-effects, abstract data types, non-terminating programs, non-determinism; in general, treating features of real-world software as "normal" rather than anomalous; modeling time as an ordinary continuous variable, consistent with standard science and engineering usage

Far-term: V&V of run-time design, including real-time performance; analytical test-planning and test-reduction; probabilistic testing; model-based mission-system development

2.2.2.2. Integrated Hardware and Software Modeling

Status: Recent experiences with deep-space systems have highlighted cost-growth issues during integration and test (I&T) and operations. These include changes to the design late in the life-cycle, often resulting in a ripple-effect of additional changes in other areas, unexpected results during testing due to unplanned interaction of fault responses, and operational limitations placed on the spacecraft based on how the system was tested (in order to "fly-as-you-test"). These issues cause cost and schedule growth during system development.

Future Directions

Model-based Fault-Management Architectures: In model-based architectures, engineers specify declarative models of operational constraints and system components (including component connectivity, command-ability, and behavior). The component models are collectively reasoned over as a system model by diagnosis-and-recovery engines. The diagnosis engine detects and diagnoses failures based on the system model and observations. The detection and diagnosis process is sound and can be complete with respect to the system model—there is no need to develop one-off detectors, i.e., "monitors." The recovery engine operates in a similar fashion; it determines an appropriate recovery action (or sequence of actions) based on operational constraints, system model, estimated system state, and intended system state (which is inferred from observed commands). The recovery engine is sound and can be complete with respect to the system model and operating constraints—there is no need to handcraft individual "responses."

Planning & Execution Architectures: Model-based planning & execution architectures provide failure diagnosis and recovery capabilities exceeding those of model-based fault-management architectures. In particular, their execution models unify command and control authority such that (i) the distinction between "nominal" and "off-nominal" control is eliminated, (ii) conflicts (e.g., resource, goal) are detected prior to execution, and (iii) short- (i.e., reactive) and long-term (i.e., deliberative) planning of spacecraft operations, resources, and behavior is enabled.

2.2.2.3. Human-System Performance Modeling

Challenges: In NASA's planned human exploration missions, for the first time in the history of spaceflight, speed-of-light limitations will impose operationally significant communication delays between flight crews and mission control. These delays will require revolutionary changes in today's ground-centered approach to off-nominal situation management. Crewmembers will have to identify, diagnose, and recover from the most time-critical anomalies during dynamic flight phases without any real-time ground assistance. The list of needed technologies includes onboard decision support tools, innovative modes of human-automation interaction, and distributed, asynchronous command and control technologies. Since the quality of these decision-support technologies is critically dependent on systems engineering decisions such as sensor placement, sensor coverage, and the overall architecture of vehicle and habitat command and data handling systems, these technologies cannot be add-ons; they must be part of a model-based systems engineering design and development process from the outset.

Future Directions: It is necessary to develop and integrate high-fidelity models of candidate vehicle and habitat systems and crew-vehicle interfaces into a virtual real-time mission operations simulation environment, complete with instrumentation to measure crew-system interactions and crew performance (cf. Sections 2.2.2.2, 2.2.3.3, 2.2.3.4). The human-systems modeling approach, that is, integrating systems engineering and human factors methods into process and tool development, will ensure that new and relevant capabilities are infused into all vehicle and habitat designs and associated operations concepts.

This will enable cost-effective analyses of a wide range of both nominal and off-nominal operations scenarios, to develop an understanding of both human decision processes and the enhancement to decision-making provided by advanced decision support tools, from both crew and mission-control perspectives. The cost of developing this type of model-based technology is amortized by a number of additional uses of the results: a platform for industry and academic partners to incorporate and test decision-support technologies and conduct additional behavioral studies; an incubator for human-systems technology spinoffs (such as natural language command and control interfaces with smart device and smart home applications) ; a medium for hands-on involvement with deep-space mission operations concepts by the public through web-based operational simulations and associated gaming capabilities; a means to engage and involve the next generation of scientists and engineers in deep-space mission development activities and STEM education; an integration laboratory to develop advanced solutions to off-nominal situation management in related operational domains such as robotic missions, distributed military operations, and next-generation airspace operations; in-situ distributed training system for just-in-time training for crew and mission support personnel.

2.2.2.4. Science and Aerospace Engineering Modeling

Science Modeling: As stated in NASA Science Mission Directorate 2010 Science Plan (2010 SMD SP), the NASA Science program seeks answers to questions such as, "How and why are Earth's climate and the environment changing? How and why does the Sun vary and affect Earth and the rest of the solar system?" To answer these questions, models and assimilation systems as well as numerical simulations are the main tools to synthesize the large array of information from many current and future scientific measurements, to relate them to physical processes and to plan for future missions. Two specific examples of such modeling are Earth System Modeling and Assimilation, and Heliophysics modeling.

Earth science models "help to quantify the interactions and balances between the various components acting on a wide variety of scales in both space and time" (2008 ESMA).

Assimilation here means the use of models to synthesize diverse in-situ and satellite data streams into a single product (analysis) that combines the strengths of each data set and also of the model. For example, ocean and land data assimilation systems are used for climate analyses and short-term climate prediction.

Earth science modeling and assimilation are core elements of the science program to improve the prediction of weather and extreme weather events, global land cover change, global water cycle, and the climate system. These technologies support international programs such as the World Climate Research Program (WCRP), the World Weather Research Programme (WWRP) and the Global Earth Observation System of Systems (GEOSS). They range from comprehensive, global whole-earth systems models to local, more process-oriented models (ocean, atmosphere, land surface). Climate and weather modeling spans timescales from local weather to short-term climate prediction to long-term climate change. Data assimilation uses available observations together with a model forecast to provide the best estimate of the state of a physical system. Some of the most important technologies are:

- High-resolution models: Increasing the model spatial resolution provides a better representation of physical processes, as well as a better use of and a better comparison with satellite data. It also enables to assess regional impacts of climate variability and change, gives better input to future mission design (through Observing System Simulation Experiments, OSSEs) and provides better forecasts. For example, increases in model resolution will bring better predictions hurricane intensity, by taking full advantage of high-resolution data such as CloudSat, CALIPSO and future GPM data.
- Integrated models: Increasing computing power also enables to increase model complexity, for example by integrating information from several process-oriented models, understanding the interactions between physical, chemical and biological processes in the atmosphere, ocean and land surface. Such capabilities will contribute to the development of an Integrated Earth System Analysis (IESA) that was identified as one of the top priorities by

the Interagency working Group for Climate Variability and Change.

- Adaptability data management and analytics: For example, as stated in (2008 ESMA), the temporary storage requirement will grow from about 0.5 TB per run to 2 TB per run in 2013. Online access to products for field campaigns will grow from 10 TB in 2009 to 40 TB in 2013. Petabytes of data are shared between climate centers (more than 1 Terabyte a day). Innovative visualization and analysis of model output will enable new discoveries.

In Heliophysics, modeling and numerical simulations have recently become very important to support the understanding of the overall dynamics of the Sun-to-Earth or Sun-to-Planet chain and to forecast and describe space weather. As in Earth Science, Heliophysics models are now widely used to assist in the scientific analysis of spacecraft-provided data sets, as well as in mission planning and conception. Examples of Heliophysics models are those dealing with processes such as the magnetic reconnection, the physics of shocks, particle acceleration and of rotational discontinuities, the turbulent dissipation of magnetic and velocity fields in space plasmas, the initiation of Solar Flares and of Coronal Mass Ejections (CMEs), the structure and evolution of Interplanetary Coronal Mass Ejections (ICMEs), the Photosphere-corona connection, cross-scale and inter-regional coupling of space plasmas (similar to multi-fluid like models) and Space Weather. Space Weather is of particular importance, as it covers all the same topics as general heliophysics science, requires cross-scale coupling, and has many implications on other modeling activities, particularly Earth Science.

Like Earth Science, Heliophysics moels deal with very large volumes of data, both from observations and from simulations; for example, the new Solar Dynamics Observatory (SDO) mission generates data at a rate of about 1 TB per day (2008 HS). Significant improvements in storage, processing, assimilation and visualization technologies are required, including data mining, automated metadata acquisition, and high-end computing.

Some specific Heliophysics modeling challenges include multi-scale problems for processes on scales of ~1km that determine evolution of system of > 107 km; time scales with the solar cycle of about 11 years, and the proton cyclotron time of about 1s; systems of about 106 km that generate km-scale features, e.g., auroral arcs; coupling to lower atmosphere and other planetary environments; particle models coupling to fluid models; analysis of complex data sets.

One possible cross cutting revolutionary technological concept for all Science and Exploration systems, is the "Sensor Web", which represents a new paradigm for data assimilation and could lead to significant improvements in Science Modeling. It is defined to be an intelligent data collection system comprised of widely deployed, heterogeneous sensors. A sophisticated communications fabric would enable rapid, seamless interaction between instruments and Science numerical models, enabling the data assimilation system to identify an "optimal" sequence of targeted observations and autonomously collect data at specific locations in space and time. Future missions would be made more cost-effective, as the sensor web capabilities would maximize the return on investment of next-generation observing systems.

As a summary, in order to take full advantage of the nation's investment in satellite- and spacecraft-provided datasets and also to assist in future mission planning and design, some of the technological advancements necessary to Science Modeling are:

- Programming Environment (still expecting Fortran to be the main programming language):
  - » Near-Term:
    - ◊ Development of Fortran-compatible, parallel libraries
    - ◊ Development of software standards and interoperability standards
  - » Mid-Term:
    - ◊ Establishment of modeling testbeds and transition support to new High Performance Processors to take full advantage of new HPC technology, such as improved parallel I/O and optimal trade-offs between memory, I/O and processor utilization
- Improvement of software modeling frameworks/community models access (such as ESMF and CCMC)
  - » Long-Term:
- Development of software engineering tools to facilitate a transparent adaptation of modeling code to architecture evolution
- Data Access and Analysis
  - » Near-Term:
    - ◊ Development of tools to deal with exponentially increasing amounts of data, soon to reach Exabytes (1018 bytes) per day
    - ◊ Development of fast and transparent access

    - between distributed and remote data storage (bandwidth, firewalls) and simulations
  - Mid-Term:
    - Development of "Sensor Web" capabilities with on-demand, real- or near-real-time satellite and in situ targeted observations that can be used in the modeling process
    - Development of standards for data sharing and distribution (format, metadata, naming conventions, ontologies)
  - Long-Term:
    - Development of data mining and computer-aided discovery tools
    - Development of tools to perform diagnostics and new data acquisition while running the models
    - Full deployment of advanced Sensor Web-like frameworks for transparent, rapid and seamless interaction between observing systems and Science numerical models
- Optimal utilization of HEC resources available to modeling, assimilation, simulation and visualization,
  - Short-Term (by 2015):
    - ~10-100 Teraflop sustained (4X – 10X resolution, ensembles) (currently about 2 Teraflop)
    - 300 TB RAM (currently, 70 TB)
    - 20 PB+ online disk cache (currently, 1.5 PB)
    - 100 Gbit/sec sustained network (currently, 1 Gbit/sec)
  - Mid-Term: Use of coprocessors and accelerators, e.g.,FPGAs
  - Long-Term: Use of quantum computing

Aerospace Engineering. There are five common themes among the high-priority challenges for NASA's aerospace engineering programs:

- Physics-based analysis tools to enable analytical capabilities that go far beyond existing modeling and simulation capabilities and reduce the use of empirical approaches in vehicle design.
- Multidisciplinary design tools to integrate high-fidelity analyses with efficient design methods and to accommodate uncertainty, multiple objectives, and large-scale systems.
- Advanced configurations to go beyond the ability of conventional technologies to achieve ARMD Strategic Objectives (i.e., capacity, safety and reliability, efficiency and performance, energy and the environment, synergies with national and homeland security, support to space).
- Intelligent and adaptive systems to significantly improve the performance and robustness of vehicles and command-and-control systems.
- Complex interactive systems to better understand the nature of and options for improving the performance of advanced aerospace systems encompassing a broad range of crewed and autonomous platforms.

Research in aerospace engineering can be divided among four major categories [consistent with the five themes above]: Aerosciences; Propulsion and Power; Dynamics, Control, Navigation, Guidance and Avionics; and Intelligent and Human Integrated Systems, Operations, Decision Making and Networking. Additionally, a fifth category (Materials, Structures, Mechanical Systems and Manufacturing) was addressed by its own roadmap (TA 12). Each of these themes and categories is highly dependent on a combination of advanced modeling, simulation, information technology and processing (MSITP). Specific activities in aerospace engineering that are relevant to MSITP include:

- Aerosciences includes the development of rapid, computationally efficient and accurate computational methods to enable consideration of novel vehicle configurations at off-design conditions; improved transition and turbulence models for prediction of the location and extent of separated regions and shocks, the effects of streamline curvature and juncture flows; and multidisciplinary, multiscale and multiphysics analyses to consider propulsion-vehicle integration, aeroacoustics, aeroelasticity, and other cross-cutting design considerations from the beginning of the design process; and supporting elements such as development of higher-order algorithms, adaptive grid generation techniques, and quantification of uncertainty.
- Propulsion and Power includes validated physics-based numerical simulation codes for component-level analysis and the improvement of multidisciplinary, system-level design tools for vehicle analysis; modeling and experimental validation of combustion process physics such as injection, mixing, ignition, finite-rate kinetics, turbulence–chemistry interactions, and combustion instability to improve efficiency and life; computationally efficient modeling of highly complex, nonlinear, coupled aeroelastic/

flight dynamics phenomena; and physics-based modeling tools such as computational fluid dynamics (CFD), life prediction tools, and steady-state and dynamic performance evaluation.

- Dynamics, Control, Navigation, Guidance and Avionics Intelligent and Human Integrated Systems includes closed-loop flow control algorithms to enable morphing of the shape of a control surface or to control the flow locally to increase lift and reduce drag; models to control interactions between flight controls, propulsion, structures, noise, emission, health monitoring and possibly closed loop aerodynamic control; and direct adaptation schemes and learning adaptive schemes to develop intelligent and adaptive controls.
- Intelligent and Human Integrated Systems, Operations, Decision Making and Networking includes analysis of complex interactive systems including development of safe separation algorithms that simultaneously meet precision sequencing, merging, spacing requirements in addition to environmental constraints; methods for modeling human-machine interactions; and risk models.
- Materials, Structures, Mechanical Systems and Manufacturing includes physics-based stiffness, strength and damage models spanning length scales from nano- to continuum (including computational materials modeling); high-fidelity analyses for structural design; modeling of pyrotechnic dynamics and pyrotechnic shock generation; structural, landing, impact and deployment simulation; instability and buckling simulation; multidisciplinary design and analysis; material processing and manufacturing processes; computational NDE; and the Virtual Digital Fleet Leader (Digital Twin — see Section 2.2.3.3).

2.2.2.5. Frameworks, Languages, Tools, and Standards

Status: In order to support the increasing complexity of NASA missions and exploration of game-changing technology, a modeling and simulation framework is required. Using this framework, investments and alternatives can be evaluated in a repeatable manner, and mission system changes can be planned and monitored. Towards this objective, architecture frameworks have been established as a guide for the development of architectures. Such frameworks define a common approach for architecture description, presentation, and integration. The framework is intended to ensure that architecture descriptions can be compared and related across boundaries.

The System Modeling Language (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. SysML models include many different diagram types, capturing structural, behavioral and architectural information. The SysML approach integrates H/W and S/W disciplines, and it is scalable, adaptable to different domains, and is supported by multiple commercial tools.

Future Directions: The benefits of using a uniform architecture framework across the agency cannot be realized without a set of case studies available for system engineers to review. There is a need to develop some key exemplars of architectural views for specific organizations and projects in order to help clarify the use of this technology. By modeling a full project such as ground system architecture within an architecture framework, the method and the models will serve as a template for future projects.

Library of SysML models of NASA related systems: Using a library of SysML and UML 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. The envisioned library of 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.

Profiles for spacecraft, space robotics, space habitats: Profiles provide a means of tailoring UML/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 Space Habitats, we will provide powerful mechanisms to NASA systems engineers for designing future space systems. Existing SysML Profiles such as MARTE for Real Time Embedded Systems should be assessed and applied to current NASA Projects.

Requirements Modeling: SysML offers requirements modeling capabilities, thus providing ways to visualize important requirements relationships. There is a need to combine traditional requirements management and SysML requirements modeling in a standardized and sustainable way.

Executable models: Executable models seek to define, precisely, the execution semantics of the relevant elements of the models. The benefits of this are that the models can be unambiguously interpreted; they can be executed, which, given a suitable environment, means that they can be validated early in the development lifecycle; they can be translated to target code achieving 100% code generation. This can provide a strong analysis completion criterion, which is: "The model is complete when it successfully executes the tests designed for it".

2.2.3. Simulation

2.2.3.1. Distributed Simulation

Status: NASA has several flavors of distributed simulation. In one case, systems and applications run at a single location and the larger team views the simulation from distributed immersive clients. In this case, provisions are made to allow the team members to view the simulations live or, in many cases, after the fact from stored and shared information. The other major distributed simulation case is computationally distributed simulation. This case borrows heavily from the distributed simulation community of another government agency for tools and standards. In this case, applications, models and simulations are hosted by each of the elements involved in the simulation and these elements share information as needed to simulate the entire system. It is important to note that, while NASA and the other government agency are using similar tools and protocols, non-NASA simulations typically involve a large number of basic simulations while NASA simulations typically involve a small number of complex simulations.

Future Directions: There is a need for large scale, shared, secure immersive environments to support team development and analysis of information. These environments must allow for rapid development and inclusion of new information / knowledge, provide an interface where intelligent agents can work alongside the humans and manage the authority elements necessary to meet national and international intellectual property and national security requirements. In addition, these environments must allow for the significantly varied communications capabilities like those associated with teams comprised of both terrestrial and extraterrestrial members.

There is a need for very large (multi-Petabyte), distributed, managed, secure data. While this is not rocket science, the problem needs to be solved in such a way that the data is accessible to those who need it, as well as the extended community, including participatory exploration. There is also a need for high speed computer networks to move, share and allow secure interaction with these large data sets.

2.2.3.2. Integrated System Lifecycle Simulation

Status: NASA tools and analysis capabilities address most, if not all, of the system lifecycle. Tools exist to support: rapid concept development, system integration, process development, supply chain management, industrial base management, simulation based test, operational support and capability re-utilization. However, NASA is just now taking steps to allow these simulation and analysis tools to share information with each other across Programs and Projects.

Integration is impeded by ad hoc data formats, system incompatibilities, fragmented programs and the lack of a central data hosting capability. Added to these challenges are the huge data elements and the need to store information across a system's multi-decade lifecycle. The issues can be translated into the format, the fidelity, or the security/proprietary nature of the information. Other concerns identified the need to "Monte Carlo" the analysis tools. This translates into the ability to take a piece of Fortran code from the 1980's, feed it data from a C++ vehicle planning tool, and send the information to an excel spread sheet for graphing. One area, data formats, has received significant attention in recent years. The desire to share information, and the steady adoption of XML for a data interface, has led to many tools being able to share information, either as files or by communicating directly to one other. A very visible example of this is the earth science communities' utilization of Google Earth to share science information via KML.

Future Directions: The primary need is not for point technologies, but the interfaces, algorithms, and networked platforms necessary to apply them to large, complex, multi-decadal, systems of systems. Formats and integration still have a considerable way to go, but progress is being made. The areas with less visible progress include application integration, provision for huge, multi-decadal data and the legal implications associated with international partnerships and advanced technology. However, based on stakeholder discussions, progress has been made in some areas that can be leveraged.

Provisions to allow for storage and management of huge multi-decadal data are not far along. Distributed Simulation technologies will enable team members to provide the data needed, without providing the secret pieces, to ensure successful subsystems and system level design integration and test. Multi-fidelity models, and the ability to share only the "safe" fidelities, will enable detailed systems integration activities to be performed early enough in the lifecycle to make a significant positive impact on system lifecycle cost.

2.2.3.3. Simulation-Based Systems Engineering

Status: The technical elements necessary to perform Simulation-Based System Engineering are in place, and in use, all across NASA. Stakeholder interviews have pointed out instances and project reviews have demonstrated the results. However, application has been limited to projects large enough to absorb the early costs associated with the effort, no matter the lifecycle benefits.

Future generations of aerospace vehicles will require lighter mass while being subjected to higher loads and more extreme service conditions over longer time periods than the present generation of vehicles. The requirements placed on systems and subsystems ranging from propulsion and energy storage to avionics to thermal protection will be greater than previously experienced while demands on long-term reliability will increase. Thus, the extensive legacy of historical flight information that has been relied upon since the Apollo era will likely be insufficient to either certify these new vehicles or to guarantee mission success. Additionally, the extensive physical testing that provided the confidence needed to fly previous missions has become increasingly expensive to perform.

Future Directions: The modeling and simulation approaches that are being advocated throughout this roadmapping exercise represent improvements in the state of the art of their respective disciplines. What is not considered by these individual efforts is a comprehensive integration of best-physics models across disciplines. If those best-physics (i.e., the most accurate, physically realistic and robust) models can be integrated, they will form a basis for certification of vehicles by simulation and real-time, continuous, health management of those vehicles during their missions. They will form the foundation of a NASA digital twin.

A **digital twin** is an integrated multiphysics, multiscale simulation of a vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its corresponding flying twin. The digital twin is ultra-realistic and may consider one or more important and interdependent vehicle systems, including propulsion/energy storage, avionics, life support, vehicle structure, thermal management/TPS, etc. Manufacturing anomalies that may affect the vehicle may also be explicitly considered. In addition to the backbone of high-fidelity physical models, the digital twin integrates sensor data from the vehicle's on-board integrated vehicle health management (IVHM) system, maintenance history and all available historical/fleet data obtained using data mining and text mining. By combining all of this information, the digital twin continuously forecasts the health of the vehicle/system, the remaining useful life and the probability of mission success. The systems on board the digital twin are also capable of mitigating damage or degradation by recommending changes in mission profile to increase both the life span and the probability of mission success.

One application of the digital twin is to fly the actual vehicle's future mission(s) before its launch. Even without the benefit of continuous sensor updates, the digital twin will enable the effects of various mission parameters to be studied; effect of various anomalies to be determined; and fault, degradation and damage mitigation strategies to be validated. Additionally, parametric studies can be conducted to determine the flight plan and mission parameters that yield the greatest probability of mission success. This application becomes the foundation for certification of the flying twin.

A second application is to mirror the actual flight of its flying twin. Once the vehicle is in flight, the continuous updates of actual load, temperature and other environmental factors will be input to the models enabling continuous predictions for the flying twin. Additionally, updates of the flying twin's health parameters, such as the presence and extent of damage or the temperature of the engine, can be incorporated to reflect flight conditions. Since the algorithms comprising the digital twin are modular, the best-physics models of individual systems or subsystems can be upgraded throughout the life of the vehicle.

A third application is to perform in-situ forensics in the event of a potentially catastrophic fault or damage. Because the digital twin closely mirrors the state of health of the flying twin, it is well suited to analyzing potentially catastrophic events. Once the sensor suite on-board the flying twin has relayed the degraded state of health to the digital twin, the digital twin can begin to diagnose the causes of the anomaly.

A fourth application is to serve as a platform where the effects of modifications to mission parameters, not considered during the design phase of the flying twin, can be studied. If, for example, mission control wants to determine the effects of a failed actuator and the best mitigation, the digital twin can be used to determine new load distributions throughout the structure, the fatigue life of the structure under the new loads and the corresponding remaining life. As a result, mission managers can make the most informed decision possible about whether or not to make the change.

2.2.3.4. Simulation-Based Training and Decision Support

Note: Decision Support technologies are discussed in Sections 2.2.2.3 and 2.2.4.5.

Status: For astronauts to remain proficient in their training, on-board Simulation-Based Trainers (SBTs) are required to support long term missions; especially in the robotics and EVA domains. This on-board training objective requires that SBT software systems be scalable with respect to supported platforms, which usually means laptops for on-board SBTs.

SBT architecture is typically made up of the following components: 1) real-world environment simulation (physics models like gravity and dynamics), 2) real or flight software, 3) real or flight sensor/effector models, and 4) human interfaces (hand controllers, displays, real-world views or cameras). Depending on the required fidelity and cost factors, these components can range in fidelity, but for accurate training, real-time performance is one requirement that cannot be sacrificed.

Future Directions: SBT computer resources are getting smaller, faster and cheaper, so the future looks bright with respect to training fidelity and computer and graphics technologies. Maintainable software architectures and simulation frameworks always have room for improvement. Providing a high fidelity configurable simulation of the real system, while controlling cost and maintenance, will always be a challenge. One way to accomplish this is by reducing the number of flight computers and avionics equipment required in the simulator for the desired fidelity. Maintainable, cost effective trainers will then be feasible using COTS flight processor emulators that execute binary flight software builds on COTS PCs. This also gives the additional benefit of being more easily able to insert malfunctions and execute restart procedures from checkpoints.

2.2.4. Information Processing

2.2.4.1. Science, Engineering, and Mission Data Lifecycle

NASA information systems are highly data intensive, requiring an understanding of their architectures and specialized software and hardware technologies. (cf. Section 2.2.2.4) In the scientific data systems domain there is an increasing demand for improving the throughput of data product generation, for providing access to the data, and for moving systems towards greater distribution. A well-architected data system infrastructure plays a key role in achieving this if it can support many of these critical features discussed: distributed operations, adaptability to capture and manage different types of data as well as the scalability and elasticity required to support evolving exploration goals over time. The Table 2 summarizes the key data-intensive system areas of interest thus far and categorizes our current state-of-the-art versus where NASA should be in the next 5-10 years in the particular category area.

2.2.4.2. Intelligent Data Understanding

Status: Modern spacecraft can acquire much more data than can be downloaded to Earth. The capability to collect data is increasing at a faster rate than the capability to downlink. Onboard data analysis offers a means of mitigating the issue by summarizing the data and enabling the ability to download a subset containing the most valuable portion of the collected data. Onboard intelligent data understanding (IDU) includes the capability to analyze data, and is closely coupled to the capability to detect and respond to interesting events. The overall concept goes beyond merely collecting and transmitting data, to analyzing the collected data and taking onboard intelligent action based on the content. To date, there have been some notable successes with science event detection and response on NASA spacecraft, enabled by onboard intelligent data understanding. These include planning and executing follow-up observations of volcanic eruptions, floods, forest fires, sea-ice break-up events and the like from the Earth Observing One spacecraft, and more recently, tracking dust devils on the surface of Mars from the MER rovers.

Future Directions: IDU includes a variety of capabilities such as situational awareness, data min-

Table 2. *Summary of the key data-intensive system areas of interest*

Data Intensive Systems Thrust Area	Current Capability	Future Capability
Reference Architectures	Limited reference information	Explicit models for information and technical architecture
Distributed Architectures	Limited distributed infrastructure and data sharing	Highly distributed architectures
Information Architecture	Limited semantic models	Models that capture the semantics in science and mission data
Core Infrastructure	Data management services tightly coupled	Distributed data management services (messaging and metadata/data storage)
Data Processing and Production	Locally hosted clusters and other computational hardware	Wider use of map reduce and other open source capabilities
Data Analysis	Centralized data analysis for computation, tools and data	Separation of computation, tools and data
Data Access	Limited data sharing and software services	Standards-based approaches for accessing and sharing data
Search	Product and dataset-specific searches with form fields.	
	Rich queries, including facet-based, free-text searches, web-service based indexing	
Data Movement	Limited use of parallelized and high throughput data movement technologies	Movement towards higher performing data movement technologies
Data Dissemination	Distribution tightly bound to existing data movement technologies in place.	Distribution of massive data across highly distributed environments.

ing for target identification, and triggering rapid response. IDU serves as an approach to maximize information return on a limited downlink channel. This permits collecting data at the capacity of an instrument rather than preselecting the time and location for observations. IDU permits extended monitoring of an environment for rare events without overburdening downloads. The capability enables the capture and subsequent immediate follow-up of short-lived events, which is not possible with traditional spacecraft paradigms. Also, the ability to analyze data and immediately use this information onboard can enable reaction in a dynamic environment, a capability that could improve not only information collection but also spacecraft safety such as reaction to unanticipated hazards. This capability enables operations in uncertain and rapidly changing environments where an adequately rapid feedback loop with ground operators is not possible. There are a number of capabilities that require advancement in order to realize the full potential of onboard intelligent data understanding.

Event detection. There must be effective computational mechanisms to identify high information content data. This may involve recognizing features or events that have been pre-specified as interesting, or identifying novel events. For some purposes, detectors specialized to a specific feature and instrument are most appropriate. Another approach is to develop general-purpose detectors. This enables a single method to find a variety of features and to identify events that were not predefined. A third category of event detection focuses exclusively on novel or anomalous events.

Data summarization addresses limited downlink capacity while realizing the full observation potential of an instrument. This can include statistical approaches ensuring that the full diversity of observations is delivered. Decision products can be generated onboard and transmitted with reduced bandwidth, providing rapid response to events, with the full resolution data following later. Scene summarization that consists of identifying and characterizing observed features over time is another form of data summarization. This capability can be used to provide extensive survey information when it is not possible to return the full resolution data collected for the entire survey.

Data prioritization. Upon detection or summary of data, decisions must be made onboard as to what is the highest information content data to be transmitted. There are a number of approaches to prioritizing data from pre-specified priorities to the use of machine learning methods that dynamically update priorities based on understanding of the data and information gathered to date. There is considerable work needed to mature these methods; however, even more advanced prioritization approaches that consider the interdependence of both observations and objectives are possible.

Ultimately, models will be used to compare data collected with predicted observations to prioritize the data and identify unexpected trends as well as individual events. Another area of continued development is automated data prioritization based on conflicting science objectives. In every mission,

there are conflicting goals such as surveying a wide area vs. conducting an in-depth study of a focus region. In order to facilitate priority assessments, increasingly sophisticated scientific interest metrics will need to be developed and deployed.

Long-term goals should include a proposed spacecraft with onboard data understanding baselined as part of the mission, i.e., designed and planned throughout the mission lifecycle. Eventually, multi-spacecraft collaborative event detection, analysis, and response, enabled by distributed onboard IDU, can be realized.

2.2.4.3. Semantic Technologies

Status: Complex space systems accrue a significant number of maintenance data and problem reports. These are currently stored in unstructured text forms. The lack of common structure, semantics, and indexing creates major problems for engineers and others trying to discover recurring anomalies, causal or similarity relations among reports, and trends indicating systemic problems.

Text-mining and related approaches are particularly useful for identifying unknown recurring anomalies and unknown relations among them. The methods used to discover anomalies are based on document clustering, the process of finding, by quantitative methods, subsets of reports that have similar features.

Future Directions: Over the past 10-20 years, numerous technologies have been developed and tested for anomaly detection by text mining (Srivastava & Zave-Ulman, 2005). More effort is needed in two areas: (1) development and operational testing of text mining systems for deployment in aerospace environments, including many technical adaptations to specifically tailor them to aerospace operations; and (2) integration of methods that were formerly thought to be different and perhaps competing, namely, text mining, natural language processing, and ontologies. These lie along a continuum from relatively unstructured, data-driven, and inductive approaches, to highly structured, logic-based approaches (Malin et al., 2010).

As the use of semantic technologies becomes routine, human-intensive coding needs to be eliminated. Also fixed logical structures will be replaced by self-adapting approaches based on machine learning. Emerging technologies will include representation and visualization of complex data structures such as biological structures and processes; new technologies for incremental analysis of ultra large-scale; evolving text bases; automated meta-data acquisition; automated construction of ontologies from unstructured text; machine learning techniques for adaptive natural language understanding; automated management of multiple hypotheses and automated generation of hypothesis-testing plans.

2.2.4.4. Collaborative Science and Engineering

Status: Science and engineering processes are evolving as information technology enables design and development organizations, some geographically dispersed, to collaborate in both synchronous and asynchronous manners throughout the lifecycle of a project. Asynchronous collaboration has been recently dominated by electronic mail or document interchange via the Web. Synchronous collaboration usually consists of periodic teleconferences and videoconferences that are used to exchange updates of progress that was conducted “off-line” by members of the team.

Future Directions: As more NASA projects are executed by distributed teams, and as NASA provides more oversight of outsourced work, the need for real-time, continuous virtual collaboration and the supporting collaborative technology environments increases. Collaborative technology environments will allow distributed teams with disparate expertise and resources, including those of partner agencies and contractors, to work in a unified manner to exchange ideas, optimize their use of resources and complete projects much more efficiently than is currently possible.

These collaborative technology environments should be a human and technology resource that provides a collaborative engineering facility for projects in all phases of science and engineering from the initial proposal to the final report. Skilled scientists and engineers can use the facility’s collaborative process and sophisticated tools to develop multidisciplinary solutions, develop new understanding of scientific phenomena and discover the inter-relationships between phenomena to an extent that would otherwise not be possible.

Within this environment, geographically dispersed members of science and engineering teams can come together for real-time collaboration in an integrated environment where the capabilities for connecting their individual solutions and outcomes are well formulated and well understood. Moreover, the highly collaborative visualization nature of the work encourages broader thinking among discipline experts. Each discipline can see directly the impact of his/her work on other disciplines and benefit from the resulting interactions. The customer can also directly observe the impact of requirements and their changes on each discipline in terms of cost, schedule, and various scientific and engineering outcomes.

Telepresence is a technology that is progressing fairly rapidly and may soon be a viable means of collaboration. Within the next 20 years, immersive (sight, sound, and touch) virtual reality will likely be commonly available and enable real-time, continuous virtual collaboration. Here again, the encouragement and adoption of open standards are critical to rapid adoption and growth.

There are many candidate mathematically based representations, editors, and graphical descriptions that aim to further the goals of integrating continuous dynamics into discrete-event simulations, providing insight into causal relations, avoiding information overload and clutter, and leveraging multiple M&S development efforts through efficient composition. Triple Graph Grammars (TGGs) are being evaluated for their ability to formalize and automate complex bidirectional model transformations. Another challenge is finding appropriate ways to visualize very different knowledge representations at different levels of abstraction, including natural language or logic or equations (requirements), diagrams (design), state or flow diagrams (implementation), and dynamic behavior (test and operations).

2.2.4.5. Advanced Mission Systems

Status: Ground-based automation and flexible crew and space system concepts are required for science and exploration mission reliability and risk management. This topic area spans flight and ground automated planning, event detection and response, and fault protection, and targets both robotic and human-robotic operations concepts. Precision landing along with rendezvous and docking are examples of some anticipated future engineering functions implying a high degree of independence from ground-based control. Science event detection and response ultimately will generalize to multiple platforms, supported by space networking.

Automated planning and scheduling has already shown considerable improvements in space operation, primarily in ground usage. Surprisingly this technology is still not common in ground usage and has to date only very limited use onboard. Some notable examples of impact to space missions are indicated below:

- Over 30% increase in observation utilization of Hubble Space Telescope
- EO-1 – over $1M/year reduction in operations costs (~30% reduction overall) due to automation in uplink operations; reduction in downtime from ground station anomalies from 5 days to 6-8 hours; 30% increase in weekly observations
- Mars Exploration Rovers – MAPGEN mixed initiative planning system used to plan operations for the Spirit and Opportunity rovers at Mars.
- Reduction of two operations staff, increase of 26% in mission productivity, and 35% reduction in mission planning errors due to use of ASPEN automated planning system on a ground- and flight-based technology experiment for another government agency.
- 50% reduction in downlink data management planning for Mars Express, increased robustness due to ability to optimize and produce multi-day/week look-ahead plans
- AUTOGEN automation of Mars relay operations has enabled cost avoidance of over $7M with projected mission lifetime total savings of over $18M

In many of the above examples, onboard mission planning and execution has been integrated with IDU to enable an overall onboard event detection and response capability. Another strongly related technology investment area is flight computing. Onboard mission planning and execution represents a prominent example of the use of model-based reasoning techniques in flight software. The search- and memory-intensive requirements of model-based reasoning technologies stress flight computing concepts and solutions. Emerging multi-core capabilities for space applications hold promise for providing the needed general purpose and high performance flight computing capacity.

Future Directions:

Small Body and Near Earth Object (NEO) Missions: A number of future NASA missions include small body targets. These mission concepts are expected to become more complex and as such, new capabilities in mission design tools will be needed to provide rapid and highly accurate orbital thrust profiles. In addition, there is a need to decrease, from several months to a few hours or less, the time required to compute small body landing trajectories for complex gravity fields and topographies. These capabilities would improve operational efficiency, lower risk to spacecraft, and enhance scientific return in shorter time period.

Long Lived In Situ Robotic Explorers: In situ missions are a major focus of NASA's solar system exploration program. Proposed future rovers and balloons are expected to have the capability to explore with more onboard decisions based on local information and less dependence on frequent human intervention. This capability involves migrating much of the mission planning and sequencing to the remote exploration element and ensuring that the associated software is sophisticated enough to handle the many unexpected situations that are likely to occur.

Space Networking: Over the next twenty-five years, an increasing number of missions will be relying on the coordinated operation of multiple assets. At Mars, the viability of dedicated relay orbiters for communications with Earth has already been demonstrated by the Mars Exploration Rovers, which have returned the vast majority of their data via the **Mars Global Surveyor and Mars Odyssey** spacecraft. Instead of planning and implementing operational sequences and communications for a single spacecraft, mission operators will need techniques for carrying out similar functions, in a timely fashion, for multiple, and sometimes very different, spacecraft. Automated data-transfer scheduling, routing, and validation will be required. Evolution in communications based on space-based networking will require evolution of the operations tools and services that depend upon that infrastructure.

Distributed Mission Operations: Distributed mission operations are necessary because the capital resources that NASA develops, such as orbital and ground-based laboratories, are scarce, and the number of scientists who want access to them is growing. Currently, distributed mission operations are conducted by either co-locating the collaborating scientists or by the use of teleconferences and data sharing.

Future needs include collaborative virtual environments in which disparate teams of scientists can explore distant objects in near real-time by efficiently and intuitively utilizing robotic vehicles to conduct their experiments. Virtual reality-based distributed mission operations, including virtual telepresence, has the ability to revolutionize the way that science experiments are performed by enabling scientists to interact with distant objects as missions are executed. Thus, robotic missions will enable virtual crews composed of scientists in distributed locations to interrogate distant objects, including those having environments too extreme for visitation by humans, as though they were walking on their surface or flying through their atmosphere. NASA has extensive experience relevant to this concept with ground stations distributed around the world for spacecraft/satellite communication and orbital relays that maintain constant communication with in-flight resources. However, unresolved challenges in distributed mission operations include 1) the need to mitigate the effects of time delay in communication between the vehicle and the ground-based researcher, and 2) the ever-increasing need for more computational resources and data storage. The capability of performing science experiments over great distances and supplying that information in formats suitable for rapid assimilation must be focal points of development.

Mission Planning and Execution is a technology area addressing the need to achieve high-level goals and requirements of a mission by creating lower level activity plans or sequences. In operational missions, automation of elements or all of this process has been operationally demonstrated to increase reliability, improve mission return, increase mission responsiveness, and potentially reduce mission operations costs.

A number of areas of technology investment present major opportunities to increase the leverage of mission planning and execution systems within space exploration missions.

Representing and Reasoning about Complex States and Resources; Spatial reasoning: One of the challenges of space exploration is the prevalence of complex, one-of-a-kind systems. Because of the expense (including opportunity cost) of such systems, they must be modeled with extremely high fidelity to ensure mission safety. Additional research is needed to enable representation of more complex constraints such as pointing, power (especially power generation), and thermal constraints. In particular, representing and reasoning about spatial constraints represents a major area in need of future work for mission planning and execution systems. Spatial constraints frequently arise in terms of observational instrument viewing geometries (e.g., under what situations can this instrument view this target with the following illumination constraints). Better techniques, algorithms, and experience in representing and reasoning about these constraints will increase the effectiveness of future mission planning and execution systems.

Optimization, Eliciting User Preferences, Mixed Initiative Planning and Execution: Most

space operations problems are optimization problems in which some objective function is to be optimized which characterizes the desirability of achieving different objectives (e.g. prioritized science objectives) as well as avoiding undesired states (e.g. expending consumable resources such as propellant, or attempting risky behaviors). Particularly challenging are mixed problems, which can have both hard and soft constraints, and multi-objective optimization problems in which multiple competing objectives occur without any explicit model of trading off between the competing objectives. Further research into representations, search algorithms, and approaches for optimization in mission planning and execution are all areas of needed future work.

Adaptive Planning: In the long term, space systems will need to deal with changing environments, degrading hardware, and evolution of mission goals. Within this long-term horizon, space systems will need to be adaptive, that is, to adjust their control strategies and search strategies over changing context. To date, most operational automated mission planning systems use domain models that have been painstakingly engineered and validated by operations staff and technologists. In order to scale to more challenging and unknown operational environments over the long term, without incurring the costs of manual model and search algorithm updating, adaptive (e.g., machine learning) techniques must be developed and validated. This area currently remains as a large area of relatively untapped potential.

Multi Agent Planning, Distributed, Self-Organizing Systems: In order to further push the exploration frontier, larger numbers of assets ultimately will be needed. Current means of managing the operations of such platforms do not scale cost effectively to tens or hundreds of platforms. Mission planning and execution systems that coordinate with other such systems, without requiring human (ground-based) oversight, ultimately will be needed. Such distributed systems would coordinate multiple assets to achieve joint goals, form and disband teams appropriately, monitor team execution, and reformulate and re-plan in the face of unit failures or evolved goals. Early work in Earth-based systems (operational since 2004 to track volcanoes, floods, and the like) has illustrated the utility of such paradigms. However this early work uses only static, human-defined collaborations among multiple assets; automating asset discovery and coordination is an unaddressed challenge.

Ground Systems Automation: A Mission Operations System (MOS) or ground system constitutes that portion of an overall space mission system that resides on Earth. Over the past two decades, technological innovations in computing and software technologies have allowed an MOS to support ever more complex missions while consuming a decreasing fraction of project development costs. Demand continues for ground systems which will plan more spacecraft activities with fewer commanding errors, provide scientists and engineers with more functionality, process and manage larger and more complex data more quickly, all while requiring fewer people to develop, deploy, operate and maintain them. Technology needs are most prevalent in the following areas:

Monitoring Systems – capturing and distributing Flight System data, maintaining knowledge of Flight System performance and ensuring its continued health and safety.

Navigation & Mission Design – maintaining knowledge of Flight System location/velocity and planning its trajectory for future Mission activities

Instrument Operations – capabilities supporting mission instrument operations planning, decision-making, and performance assessment for a wide variety of instrument types.

Table 3. *Information Processing Roadmap*

Technology Capability	Relevance	Timeframe
Ground-based automated mission planning and scheduling for hard constraints and goals	Building block for automated mission planning	**2010-2015**
Onboard event driven goal re-selection	Building block for more resource-aware onboard decision making	**2015**
Automated systems with more complex states and resources, complex optimization constraints, mixed initiative planning	Increased ability to deploy ground systems to make impacts in ongoing missions	**2010-2020**
Embedded Planning and Scheduling	Enables widespread use of onboard planning, scheduling, and execution systems	**2010-2020**
Adaptive systems	Enables robust performance in the presence of changing goals, environments, and objectives	**2030+**
Multi-agent systems	Enables robust coordination of large numbers of remote assets, critical to enable such missions without prohibitive operations costs	**2030+**

3. INTERDEPENDENCY WITH OTHER TECHNOLOGY AREAS

- TA 4 ROBOTICS, TELE-ROBOTICS, AND AUTONOMOUS SYSTEMS: Avionics & processors; Autonomous control; Automated rendezvous & docking; Autonomous precision landing & hazard avoidance; Autonomous payload offloading & spacecraft servicing; Autonomy for operations; Telerobotic operations; Human-machine interaction; Robotic health monitoring and damage assessment; Intelligent software design; Integrated systems health management; Fault management techniques; Extreme environment electronics; Software reliability
- TA 5 COMMUNICATION AND NAVIGATION: Computing-communications trade-offs.
- TA6 HUMAN HEALTH, LIFE SUPPORT AND HABITATION SYSTEMS and TA7 HUMAN EXPLORATION DESTINATION SYSTEMS: Environmental monitoring & control; Life support & habitation systems; Long duration in-space human health maintenance and monitoring; Closed-loop life support; Fire detection & suppression; Displays and Controls for Crew, Crew systems, Wearable computing; EVA Technology; Medical Prognosis, Diagnosis, detection, and treatment; Bioinformatics; Surface systems health monitoring and damage detection; Human-tended docking systems; In-space and surface operations; System commonality, modularity, and interfaces
- TA 9 ENTRY, DESCENT, AND LANDING: Multi-scale, multi-physics modeling.
- TA 10 NANOTECHNOLOGY: Quantum mechanics, atomistic, molecular dynamics and multi-scale simulations; Simulations of deformation, stiffness and strength; Durability and damage tolerance; Oxidation and environmental effects; Thermal and electrical performance/conductivity; Interface design and performance; Material processing; Radiation effects and transport
- TA 12 MATERIALS, STRUCTURAL AND MECHANICAL SYSTEMS, AND MANUFACTURING: Quantum mechanics, atomistic, molecular dynamics, discrete and continuum plasticity simulations; Stiffness and strength; Fatigue, fracture and damage mechanics; Environmental effects; Structural mechanics and kinematics; Structural, landing and impact dynamics; Deployment simulations; Instability and buckling simulations; Probabilistic/nondeterministic design; Material processing and manufacturing simulations; Computational NDE; Digital Twin (Virtual Digital Fleet Leader).
- TA 13 GROUND AND LAUNCH SYSTEMS PROCESSING : Launch complex & ground ops; Advanced process support; Advanced mission operations; Spaceport interoperability; Safe, reliable & efficient launch ops; Range tracking, surveillance, & safety; Inspection & system verification; Telemetry; Weather prediction & decision making
- TA 14 THERMAL MANAGEMENT SYSTEMS: Conduction, convection and radiation heat transfer; Ablation, gas-surface interaction simulation; Aerothermodynamics

4. POSSIBLE BENEFITS TO OTHER NATIONAL NEEDS

Safety benefits in transportation, medicine; SensorWeb communication systems for disaster response; education: to inspire and motivate current, and future generations of students; outreach: virtual exploration; interdisciplinary collaboration; business applications: virtual interactive process design, manufacturing and logistics, digital factory; Semantic Web: standards, mark-up languages, open-source component libraries, interoperable simulation; grid- and service-based architectures; high-performance simulation on COTS hardware; data-based decision and policy support.

ACRONYMS

ARMD	(NASA) Aeronautics Research Mission Directorate
ASCENDS	Active Sensing of CO_2 Emissions over Nights, Days, and Seasons
CCMC	Community Coordinated Modeling Center
CFD	Computational Fluid Dynamics
CME	Coronal Mass Ejection
COTS	Commercial Off The Shelf (hardware or software)
DARPA	Defense Advanced Research Projects Administration
DESDynI	Deformation, Ecosystem Structure and Dynamics of Ice
ESMA	Earth Science Modeling and Assimilation
ESMD	(NASA) Exploration Systems Mission Directorate
ETDP	Exploration Technology Development Program
FDIR	Failure (Fault) Detection, Isolation and Recovery
FPGA	Field Programmable Gate Array
FTD	Flagship Technology Demonstrator
GbE	Gigabit Ethernet
GEOSS	Global Earth Observation System of Systems
GPM	Global Precipitation Measurement
Grail	Gravity Recovery and Interior Laboratory
GSFC	Goddard Space Flight Center
HCI	Human-Computer Interaction
HEC	High End Computing
Hyspiri	Hyperspectral-Infrared Imager
IBEX	Interstellar Boundary Explorer
ICESat	Ice, Cloud, and land Elevation Satellite
ICME	Interplanetary Coronal Mass Ejection
IDU	Intelligent Data Understanding
IESA	Integrated Earth System Analysis
ISHM	Integrated (Intelligent) System Health Management (Monitoring)
ISS	International Space Station
I&T	Integration and Test
IVHM	Integrated (Intelligent) Vehicle Health Management (Monitoring)
JPL	Jet Propulsion Laboratory
JSC	Johnson Space Center
JWST	James Webb Space Telescope
LDCM	Landsat Data Continuity Mission
LEO	Low Earth Orbit
LSA	Latent Semantic Analysis
MARTE	Modeling and Analysis of Real-Time and Embedded (Systems)
MAX-C	Mars Astrobiology Explorer-Cacher
MBSE	Model-Based Systems Engineering
u	micro (imitating Greek letter “mu”)
MIPS	Million Instructions per Second
MISSE	Materials International Space Station Experiment
MMS	Magnetospheric Multiscale
MOTS	Modified (or Modifiable) Off The Shelf (hardware or software)
MSITP	Modeling, Simulation, Information Technology and Processing
MSL	Mars Science Laboratory
NAS	NASA Advanced Supercomputer
NAS	National Airspace System
NextGen	Next Generation (Air Traffic Management or National Airspace System)
NEO	Near-Earth Object
NPOESS	National Polar-orbiting Operational Environmental Satellite System
NPP	NPOESS Preparatory Project
NRC	National Research Council
OCO	Orbiting Carbon Observatory
OOD	Object-Oriented Design
OPFM	Outer Planet Flagship Mission
PGAS	Partitioned Global Address Space
PCA	Principal Components Analysis
RHBD	Radiation-hardened by design
RHBP	Radiation-hardened by process
SBT	Simulation-Based Training
SDO	Solar Dynamics Observatory
SMAP	Soil Moisture Active & Passive
SMD	(NASA) Science Mission Directorate
SOMD	(NASA) Space Operations Mission Directorate
STEM	Science, Technology, Engineering, and Mathematics (education)
STEREO	Solar TErrestrial RElations Observatory
SysML	Systems Modeling Language
TA	Technical Area
TABS	Technical Area Breakdown Structure
TASR	Technical Area Strategic Roadmap
TGG	Triple Graph Grammar
TRL	Technology Readiness Level
UML	Unified Modeling Language
UPC	Unified Parallel C
VM	Virtual Machine
WCRP	World Climate Research Programme
WWRP	World Weather Research Programme
XML	Extensible Markup Language

ACKNOWLEDGEMENTS

The NASA technology area draft roadmaps were developed with the support and guidance from the Office of the Chief Technologist. In addition to the primary authors, major contributors for the TA11 roadmap included the OCT TA11 Roadmapping POC, Maria Bualat; the reviewers provided by the NASA Center Chief Technologists and NASA Mission Directorate representatives, and the following individuals: Rebecca Castaño, Micah Clark, Dan Crichton, Lorraine Fesq, Michael Hesse, Gerard Holzmann, Lee Peterson, Michelle Rienecker, David Smyth, and Colin Williams.

REFERENCES

ACM Taxonomy / IEEE Computer Society Keywords. http://www.computer.org/portal/web/publications/acmtaxonomy

Douglass, S., Sprinkle, J., Bogart, C., Cassimatis, N., Howes, A., Jones, R.M., & Lewis, R. (2009). Large-scale Cognitive Modeling Using Model-integrated Computing. Mesa, AZ: AFRL.

ESMA 2008 Panel Report

Extremely Large Databases Workshop, http://www-conf.slac.stanford.edu/xldb10/

Heliophysics 2008 Panel Report

Jin, H., Hood, R., & Mehrotra, P. (2009). A Practical Study of UPC Using the NAS Parallel Benchmarks. PGAS '09 Ashburn, Virginia.

Johnson, T., Paredis, C.J.J., & Burkhart, R. (2008). Integrating Models and Simulations of Continuous Dynamics into SysML. Presented at Modelica 2008.

Knill, E. (2010). Quantum Computing. Nature, 463(28), 441-443.

Malin, J.T., Millward, C., Gomez, F., & Throop, D. (2010). Semantic Annotation of Aerospace Problem Reports to Support Text Mining. IEEE Intelligent Systems.

McCafferty, D. (2010). Should Code Be Released? Communications of the ACM, 53(10), 16-17.

Merali, Z. (2010). Why Scientific Programming Does not Compute. Nature, 467, 775-777.

NRC Steering Committee for the Decadal Survey of Civil Aeronautics. (2006). Decadal Survey of Civil Aeronautics: Foundation for the Future. http://www.nap.edu/catalog/11664.html

NRC Computer Science and Telecommunications Board. (2010). Report of a Workshop on the Scope and Nature of Computational Thinking. Washington, DC: The National Academies Press.

Nature Magazine, "Big Data" Issue, http://www.nature.com/nature/journal/v455/n7209/full/455001a.html

Oblinger, D. (2006). Bootstrapped Learning: Creating the Electronic Student that learns from Natural Instruction. DARPA IPTO.

Parnas, D.L. (2010). Really Rethinking 'Formal Methods'. IEEE Computer, 43(1), 28-34.

Science Mission Directorate, 2010 Science Plan, http://science.nasa.gov/about-us/science-strategy/

Segaran, Hammerbacher. Beautiful Data, http://oreilly.com/catalog/9780596157128.

Srivastava, A., & Zane-Ulman, B. (2005). Discovering Recurring Anomalies in Text Reports Regarding Complex Space Systems. IEEE Aerospace Conf. Proc., CD-ROM, IEEE Aerospace, March 2005.

Tallant, G.S., et al. (2004). Validation & Verification of Intelligent and Adaptive Control Systems. In Proceedings of the 2004 IEEE Aerospace Conference. DOI: 10.1109/AERO.2004.1367591

Taylor, S.J.E., Lendermann, P., Paul, R.J., Reichenthal,S.W., Straßburger, S., & Turner, S.J. (2004). Panel on Future Challenges in Modeling Methodology. In R.G. Ingalls, M.D. Rossetti, J.S. Smith, and B.A. Peters (Eds.), Proceedings of the 2004 Winter Simulation Conference.

Welch, P., Brown, N., Moores, J., Chalmers, K., Sputh, B.(2007). Integrating and extending JCSP. In A.A. McEwan, S. Schneider, W. Ifill, & P. Welch (Eds.) Communicating Process Architectures. Amsterdam: IOS Press.

Yilmaz, L., Davis, P., Fishwick, P.A., Hu, X., Miller, J.A., Hybinette, M., Ören, T.I., Reynolds, P., Sarjoughian, H., & Tolk, A. (2008). Panel Discussion: What Makes Good Research in Modeling and Simulation? Sustaining the Growth and Vitality of the M&S Discipline. In S.J. Mason, R.R. Hill, L. Mönch, O. Rose, T. Jefferson, & J.W. Fowler (Eds.), Proceedings of the 2008 Winter Simulation Conference.

Zeigler, B., Praehofer, H., & Kim, T.G. (2000). Theory of Modeling and Simulation (second edition). New York: Academic Press.

November 2010

National Aeronautics and
Space Administration

NASA Headquarters
Washington, DC 20546

www.nasa.gov