American Astronautical Society
Rocky Mountain Section

AAS 06-034

Achieving Responsive Space: The Viability of Plug-and-Play in Spacecraft Development

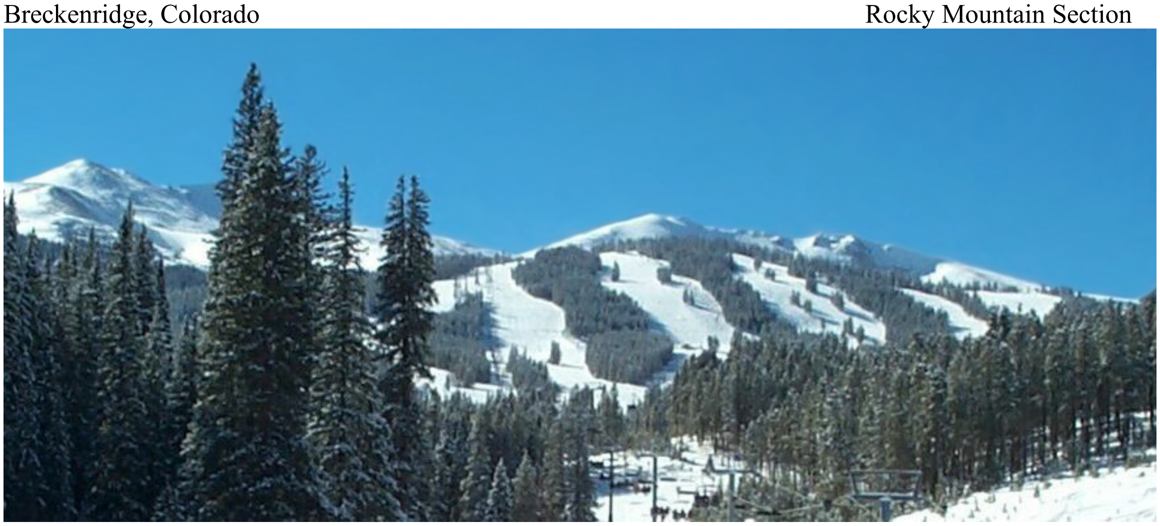
Paul Graven and Dr. Richard Van Allen
Microcosm, Inc.

L. Jane Hansen and Jon M. Pollack
HRP Systems, Inc.

29th ANNUAL AAS GUIDANCE AND CONTROL CONFERENCE

February 4-8, 2006
Breckenridge, Colorado

Sponsored by
Rocky Mountain Section



AAS Publications Office, P.O. Box 28130 - San Diego, California 92198

ACHIEVING RESPONSIVE SPACE: THE VIABILITY OF PLUG-AND-PLAY IN SPACECRAFT DEVELOPMENT

Paul Graven†, Dr. Richard Van Allen†, L. Jane Hansen and Jon M. Pollack

The emerging acceptance of responsive space as a means of allowing space-based assets to be fully functioning elements within the warfighter's arsenal has brought to light aspects of traditional spacecraft development that will require extensive modification if they are to meet the needs of the next generation spacecraft. Quick integration and launch of spacecraft has been identified as a means of achieving responsive space; and a key element to rapid integration may be the application of enhanced plug-and-play (PnP) standards to spacecraft components. Whether those components are partitioned as spacecraft and payload, spacecraft functional subsystems in conjunction with the payload, or individual spacecraft hardware components, the application of PnP will result in a paradigm shift with respect to traditional spacecraft development methodologies.

Microcosm, in conjunction with HRP Systems, under several SBIRs from AFRL/VSSE, has developed a rapid prototype implementation of PnP components focused on a guidance, navigation, and control (GN&C) application. One objective was to create a self-configuring network, including resource discovery, configuration, and management, in addition to providing system level configuration and graceful degradation based on the plugged-in, available components. This objective was to be achieved using GN&C applications which are typically highly deterministic and time dependent, as a challenging proof-of-concept. A two-state GN&C application was implemented in software to provide control laws that reflect a low fidelity solution and a higher fidelity solution based on component availability and measurement quality.

This paper will discuss PnP as a disruptive technology, especially when applied to a GN&C system implementation. In addition, an overview of prototypical implementations, developed by Microcosm and HRP Systems as part of several AFRL funded SBIR efforts, will be presented to provide a foundation for our vision of the future. This prototype work will be discussed in generic terms, specifically for the purpose of establishing a basis of knowledge for the concept extension that will be advocated later in the paper. Finally, the paper will provide a recommendation of PnP concepts that can be overlaid on the spacecraft development process, specifically in terms of GN&C, to make responsive space attainable in the near term.

† American Astronautical Society (AAS) members.

PLUG-AND-PLAY: A DISRUPTIVE TECHNOLOGY

Plug-and-Play (PnP) implies the ability to insert a component and have it become an operational element within the system, automatically, whether the system is powered on or not. The concept of PnP is most often associated with personal computers (PCs) where a mouse (or other peripheral device) can be quickly and easily integrated into the system configuration using the universal serial bus (USB) standard for both data and power. Most PCs are also capable of identifying the correct device driver for the new component, or if necessary, they can access the internet and acquire the driver that is needed. This rapid integration of an operational workstation by an untrained individual seems attractive to the responsive space requirement for rapid integration. However, the circumstances are not quite equivalent. For example, it is highly unlikely that spacecraft components will be hot swapped on-orbit or even on the integration floor. Additionally, the upfront non-recurring engineering (NRE) that was incurred by the personal computing industry, supported by the large volume of sales, in conjunction with the huge overhead required in terms of computer memory, throughput, and initialization processing, will never be practical for the spacecraft embedded software market. The utility of PnP, with respect to operationally responsive spacecraft, is to identify the key components that enhance rapid integration, fault tolerance, and reuse, and map them onto the spacecraft development process.

To make PnP viable in an operational guidance, navigation, and control (GN&C) system, standards must exist. It is through the use of standards that a system can identify a component and communicate with it, without having component specific software being written for each application. In the case of the PC, standards for power distribution, data formatting, data transmission, and user interfaces have been established through the USB protocol. In the realm of spacecraft development, software development and avionics standards and protocols also exist. By extending these standards and protocols to include mechanisms for discovery and configuration (as well as reconfiguration), the concept of PnP can be used to decrease integration and test time, increase on-orbit reliability, and enhance software reuse, making operationally responsive space attainable.

The task of creating a standard, even if extending existing standards, as well as adhering to new standards in the process of developing embedded flight software, takes a lot of engineering support. In fact, it is probably easier to continue to do business as usual, rather than attempt to change the thinking and the approach. However, with the up-front investment of time and energy and a shift in the development processes and methodologies, the long-term rewards seem compelling. The initial attempts at implementing PnP may not achieve the rigorous standards of traditional spacecraft in terms of timing, latency, and throughput, but will be “good enough” to move the industry forward and evolve the technology such that it will eventually equal or exceed the performance of existing systems. Thus, PnP, as applied to the spacecraft development environment, and GN&C components in particular, represents a disruptive technology.

The term disruptive technology was coined by Clayton M. Christensen (Harvard Business School Professor) and described in his 1997 book “The Innovator’s Dilemma”.

Essentially, disruptive technologies are "innovations that result in worse product performance, at least in the near-term." The difference between disruptive technologies and sustaining technologies is illustrated in Figure 1. Disruptive technologies are generally "cheaper, simpler, smaller, and, frequently, more convenient to use." ("The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail" June 10 1997 HBS Press Book). Christensen's research explains why "established companies—even those competently managed by smart people—have such trouble countering or embracing disruptive innovations that are on the horizon. His theory is that organizations customarily develop mind-sets and processes that revolve around doing what they already know how to do. Once that pattern becomes established, managers have great difficulty justifying to others or even themselves the need to turn their processes upside down to respond to a barely emergent market change. By the time the threat is apparent, however, it is usually too late; upstart companies have seized a substantial lead.[1]

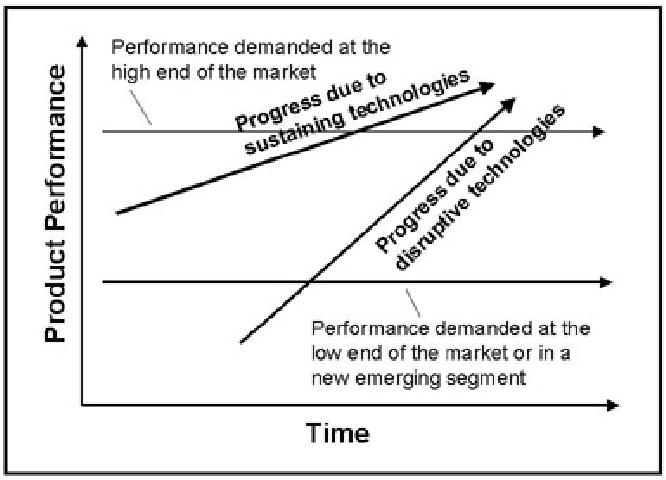


Figure 1 Disruptive vs. Sustaining Technologies[2]

Overlaying this business concept, developed by Christensen, incremental improvements in traditional spacecraft development can be viewed as sustaining technology while operationally responsive spacecraft would be the down-market application. In that context, PnP avionics and specifically PnP GN&C components and subsystems are the disruptive technology which in the near-term will enhance product performance along dimensions necessary to meet the challenges of responsive space, and which have the potential to eventually displace mainstream technology and practice. Certainly, embracing this disruptive technology would dramatically change the way spacecraft are designed, developed, and deployed. By carefully evaluating the benefits of PnP and constraining the shift in the development process to meet the real, high level requirements of spacecraft GN&C systems, operationally responsive space can be achieved.

---

1 CIO Magazine: http://www.cio.com/archive/040101/disruption.html.

2 From Christensen's book, "The Innovator's Dilemma," extracted from the website: http://web.mit.edu/6.933/www/Fall2000/teradyne/clay.html.

OPERATIONALLY RESPONSIVE SPACECRAFT

Traditional spacecraft cost millions of dollars and are extremely tailored (i.e. optimized) to meet the specific requirements of each customer. The evolution of this process has been spiral in nature. As spacecraft cost more to develop, customers require more specific functionality, catering to their precise mission requirements, which in turn causes suppliers to optimize their designs for performance, based on those unique customer requirements. Eventually, each spacecraft is meticulously designed and developed to meet precise mission objectives, requiring a lot of time and money. This approach, of course, made technical and economic sense with low production volumes and high launch, bus, and payload costs.

Operationally responsive space is rooted in the concept that if many smaller and cheaper, essentially expendable spacecraft, can be produced, system performance can be "good enough," rather than optimal. In this incarnation of the spiral development process, the production of multiple, low-cost spacecraft implies the need for off-the-shelf building blocks or modular components, and modularity can be achieved at various levels. A spacecraft and payload, developed for a specific mission type and placed in inventory for future launch, might achieve rapid response. A standardized bus might be considered a modular component as it is paired with an "inventory" payload, which together are capable of meeting a mission objective when integrated and launched. Modularity of the spacecraft bus in terms of subsystems that are selected and combined to meet specific mission needs is another way to look at modularity. Finally, creating individual subsystem components that are modular in nature and thus can be plugged together to create the desired spacecraft functionality brings modularity to an even lower level.

Another attribute of responsive space is time-to-orbit. Certainly a mission can be designed where a single spacecraft can be built and deployed to achieve optimal time-to-orbit, but responsive space implies that the reduction of time-to-orbit is an on-going process. In this case, responsiveness is determined not only by the time-to-orbit for a single mission, but also in the re-usability of the system, subsystem or components and thus the future flexibility. At AFRL, Dr. James Lyke and Mr. Donald Fronterhouse have pioneered the Responsive Space Testbed (RST), an on-going activity.[3,4] As shown in Figure 2, Lyke and Fronterhouse have captured the essence of these issues of where modularity is best suited. They have posed the question "Where is good enough?" and quantify the question of modularity in terms of investment of time and money.

---

[3] Lyke, Jim, Don Fronterhouse, Scott Cannon, Denise Lanza, and Wheaton (Tony) Byers. 2005. "Space Plug-and-Play Avionics." AIAA Paper No. RS3-2005-5001, presented at the 3rd Responsive Space Conference, Los Angeles, CA April 25-28.

[4] Lyke, Jim, Scott Cannon, Don Fronterhouse, Denise Lanza, and Tony Byers. 2005. "A Plug-and-Play System for Spacecraft Components Based on the USB Standard. Paper No. SSC05-II-1, presented at the 19th Annual AIAA/USU Conference on Small Satellites, Logan UT, August 8-11.

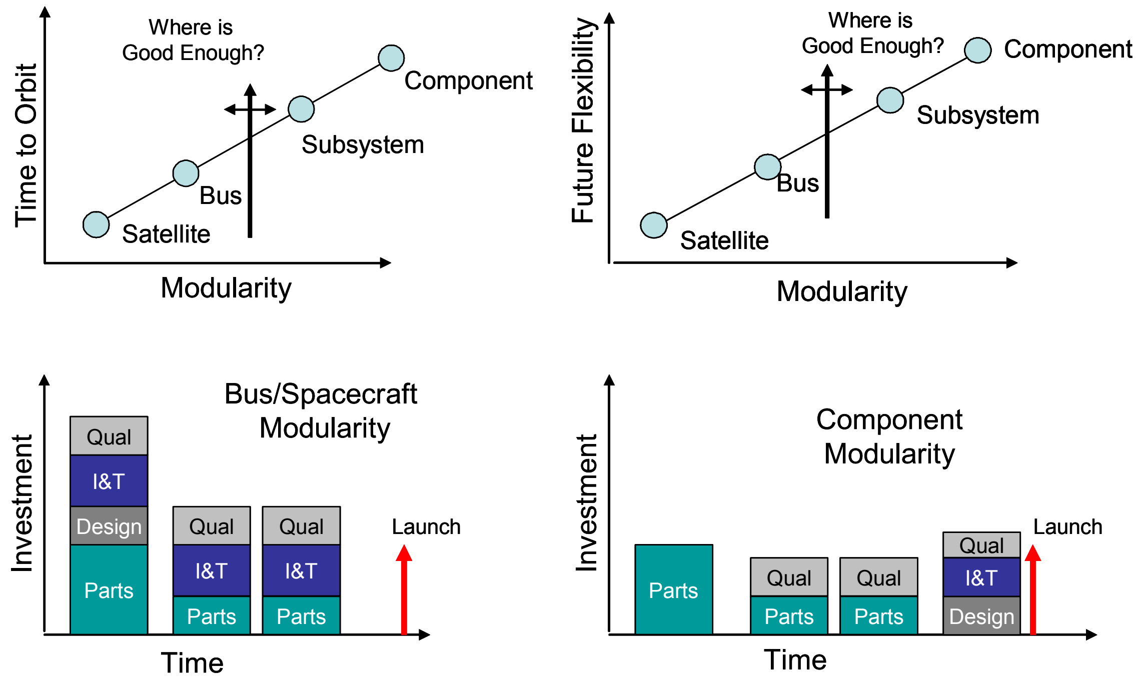


Figure 2 Modularity and Standardization are Keys to Responsiveness

Viewing system performance as good enough seems reasonable in the macro sense, but at the subsystem level, there are specific criteria that have always been used to establish the viability of the implementation. For GN&C, these criteria have included real-time performance with limited latency between sensor measurements and actuator operation.

GN&C provides a particularly challenging test case for modularity and PnP for a number of reasons:

- Mission criticality
- Real-time operation / Low tolerance for latency
- Very high availability and reliability requirements
- Increasing bandwidth and computational requirements
- Complexity
- Coordination requirements

It was not so many years ago that navigation was a problem for the ground segment to solve, and most spacecraft control laws were implemented with analog electronics. This situation led, by necessity, to highly integrated solutions in order to squeeze out the last increments in performance. Although software and fast digital computers have largely replaced the analog electronics, highly integrated and customized solutions are still the norm. Again, this norm makes economic sense at this point. For expensive space systems, there are typically considerable economic/performance benefits from customization and tight integration.

Now, however, with the potential of launch, bus, and payload each in the range of $5M to $15M, there is reason to believe that the economics will shift in favor of PnP and modular solutions. The solutions will start with modest performance, but dramatically lower design and integration costs, and drive up-market from there, bringing significant cost and schedule benefits to mainstream applications.

To reach an equilibrium between the desire for responsive space and the traditional criteria for developing a GN&C system, the Microcosm team established the following high level requirements for the implementation of a PnP architecture for GN&C applications:

1. Real-time (predictive, if not deterministic) performance
2. Support high availability, capable of fault tolerance
3. Scalable and extensible design (to both varying levels of network bandwidth and higher capability processors)
4. Low hardware overhead — size, weight, power
5. Compact software (to maximize the use of low cost microcontrollers)
6. Leverage existing technology (hardware, software, protocols)
7. Simplify system design and integration process
8. Maximize reuse [through well defined services and application program interfaces (APIs)]
9. Maximize portability (through partitioning of platform dependent code)
10. Create a freely available, no royalty, open design specification

First and foremost, for future implementation in a spacecraft system, the architecture has to support a real-time system environment. GN&C systems have evolved from strictly hard real-time analog (firmware) to a mixture of analog and digital (software) to today, when the heart of most GN&C systems is real-time embedded software. Additionally, the need to maintain a high level of testability and determinism is necessary to foster confidence in a mission critical, safety of flight, application. Where some real world applications are successful when they achieve "best effort" performance, a GN&C system, like a medical system, requires a higher and more robust level of operation. Finally, the architecture must include functional partitioning of components, interfaces, and algorithms designed to create software re-use, through the adherence to standard data interfaces at all levels of implementation.

AN EXAMPLE OF PnP APPLIED TO A GN&C IMPLEMENTATION

Microcosm, in conjunction with HRP Systems, under several SBIRs funded by AFRL/VSSE, has developed a rapid prototype implementation of PnP, at the component level, focused on a GN&C application. This prototype implementation has been used to demonstrate and evaluate various aspects of PnP, specifically as they apply to a spacecraft GN&C system. The Microcosm team has developed a software-based product that provides network dependent and independent components. By leveraging commercial off-the-shelf (COTS) networks to the greatest extent possible, a self-configuring, avionics network was created allowing Microcosm to place the emphasis of this work on the GN&C system. The self-configuring, avionics network, in conjunction with GN&C algorithms that operate on atomic level data (i.e., data at the lowest possible level of functional complexity, such as spacecraft rotation angles), form the basic system for a prototypical PnP system.

In this prototype environment, the components of a system are rapidly assembled with minimal need to write detailed, low-level code pertaining to the interface or usage of each element. Additionally, with standardized data interfaces, independent of the source and the algorithms, control law software and algorithms are quickly assembled and integrated for varying sensor and actuator suites. The resulting automation allows system designers to focus on the design of higher-level software in an object-oriented fashion, a process that itself might be automated under this concept.

The COTS leveraged networks created a system that includes non-homogeneous types, with bandwidth and protocol variations, and accommodates a multi-mastered system. The data were both pushed by the producers and pulled by the consumers, which yields the most robust implementation possible. Data flows from point-to-point among the various nodes or can be accessed through the resource and data managers. In any implementation, the network supports deterministic timing with real-time data rates of up to 1 megabit/second on the lower level sensor network and 12 megabits/second on the higher level mission network. Fault tolerance in the system is achieved both by redundancy of devices and by over determined data, which works with the GN&C application software because the data expected by the application software is at the atomic data level. The basic concepts presented in the prototype architecture are illustrated in Figure 3.

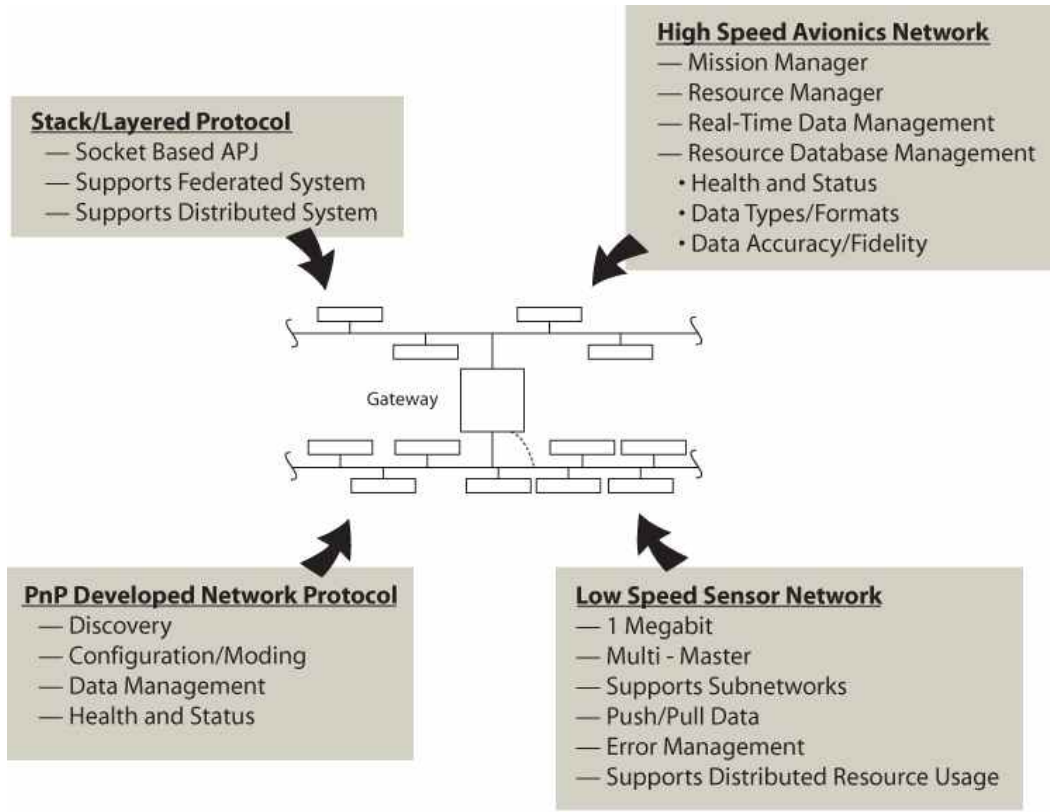


Figure 3 Demonstration Concepts as Applied to the Prototype Architecture

There are four primary components to the software product: mission manager, resource manager, network manager, and the GN&C application software. The Mission Manager (MM) is the component of the system that handles the mission objectives, requirements, and success criteria. It is in this software that decisions are made regarding the mission phase and the algorithms that must execute at any given time during a mission. The Resource Manager (RM) administrates the resource discovery process and maintains the information regarding data descriptions and the system configuration, as well as subsystem health and status. A component of the RM is the local RM that provides an API to resident software applications by abstracting (partitioning for easy reconfiguration) the physical activity needed to access the data while providing a mechanism for error handling and reporting. The Network Manager (NM) supervises network addressing, routing, protocol, and the interfaces associated with the medium over which data are being supplied. This is the component of the software that will be changed as different mediums and protocols are introduced. Finally, the GN&C application software understands the required sensor inputs, at an atomic level, and supports processing and outputs to the actuators (e.g., attitude control thrusters), based on mission mode control laws. As a subcomponent of the GN&C software, a Helper Application (HA) is created to provide input/output data handling, unpacking the native sensor data and sensor coordinate frame, creating atomic data as well as packing atomic data into actuator specific data formats. Each of these elements is needed to make the overall PnP system operational.

The key to making the implementation described above a true PnP capability for GN&C applications is reducing the data types down to their lowest functional or atomic levels. By evaluating each piece of data at the lowest level, the GN&C application software is no longer dependent on traditional subsystems or line replaceable units (LRUs), but rather focuses on the primary inputs and outputs that need to be generated. The actual number of input types needed for GN&C can be limited to four: time, rotation measurements, translation measurements, and third body angles, which in reality, equates to a finite number of atomic level data elements that need to be produced by the various sensors and sensor suites. More specifically, for our prototype demonstration(s), these data types include:

<u>Sensors</u>

1. Time — time stamp
2. Rotation measurements — rotation angles (3 components), rotation rates (3), rotation accelerations (3)
3. Translation measurements — translation position (3), translation rates (3), translation accelerations (3)
4. Third body angles — Earth angle (2 components), Earth angle rate (2), Sun angle (2), Moon angle (2), and star angle(s) (2 components for each star in the field-of-view).

<u>Actuators</u>

1. Requested thruster force (3 components)
2. Requested thruster torque (3)
3. Wheel momentum
4. Magnetic torquer duty cycle

The GN&C application software is required to be accessible by the RM, which resides below it in the overall layered network structure. The primary use of the GN&C application software within the context of this demonstration system is to test the capabilities and responsibilities of the proposed network and resource management software to meet the top-level requirements of a traditional GN&C system. The basic functionality of the GN&C application software is shown in Figure 4. The GN&C application software must take input data from any attitude or orbit determination device, convert that data into a common format that can be processed by the attitude and orbit control software via a HA, and execute actuator commands in actuator-specific format to properly control both orbit and attitude, again with the use of a HA. This software is flexible enough to accommodate any type of attitude or orbit determination sensor, as well as all types of actuators.

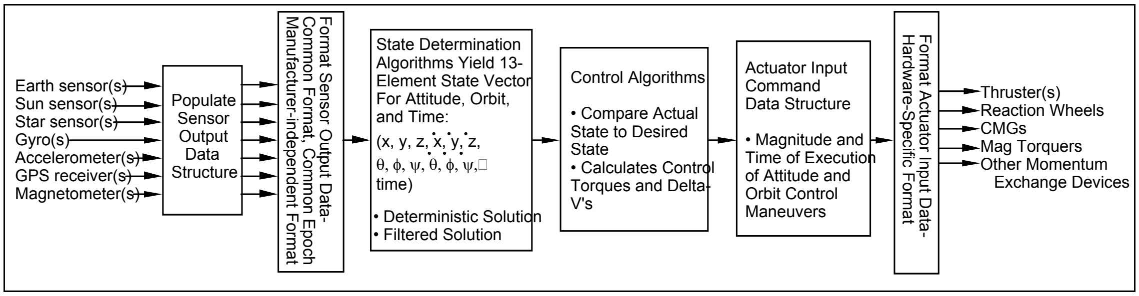


Figure 4 Spacecraft GN&C Software Structure and Primary Functions

For the demonstration and initial evaluation, to simplify the scenario, the GN&C software currently is hosted on a PC but can be quickly ported to most flight computers. The GN&C solution used a bang-bang controller that was not a closed-loop solution. When the controller believed that thruster firings were necessary, it attempted to fire a thruster, which lights LEDs on a separate 8051 processor board. Since a change in attitude does not occur, the controller continues to request thruster firings. Thus, the system is not closed-loop, but demonstrates the ability to achieve a viable solution with varying sensor inputs, which demonstrates the resource discover, management, and reconfiguration process at a low level.

The demonstration objectives and goals were focused on GN&C application software and systems, and included the following evaluation criteria:

1. Initial discovery and management of network resources (GN&C sensors and actuators)
2. GN&C algorithms operating with nominal data in a static environment
3. Graceful degradation of GN&C functionality through the loss of a sensor
4. Reconfiguration of the network to make use of data from alternate GN&C sensor sources
5. Adaptability of the GN&C algorithms as operations continue with data from various GN&C sensors and/or using different GN&C actuators
6. Real-time discovery as new GN&C sensors are added to the configuration

The demonstration configuration, shown in Figure 5, includes a simplified rate table to provide the mechanism to test the GN&C algorithms in a dynamic environment. The demonstration scenario includes an initialization period, where first-time discovery through self-announcing by components occurs and the RM database is populated. Next, the GN&C algorithms demonstrate the ability of the system to sense motion, based on atomic level sensor data, which will ultimately cause the firing of actuators or the energizing of magnetic torquers. Several aspects of reconfigurability and graceful degradation are demonstrated through the failure of sensors. The addition of a new, more accurate, sensor demonstrates the real-time discovery mechanism.

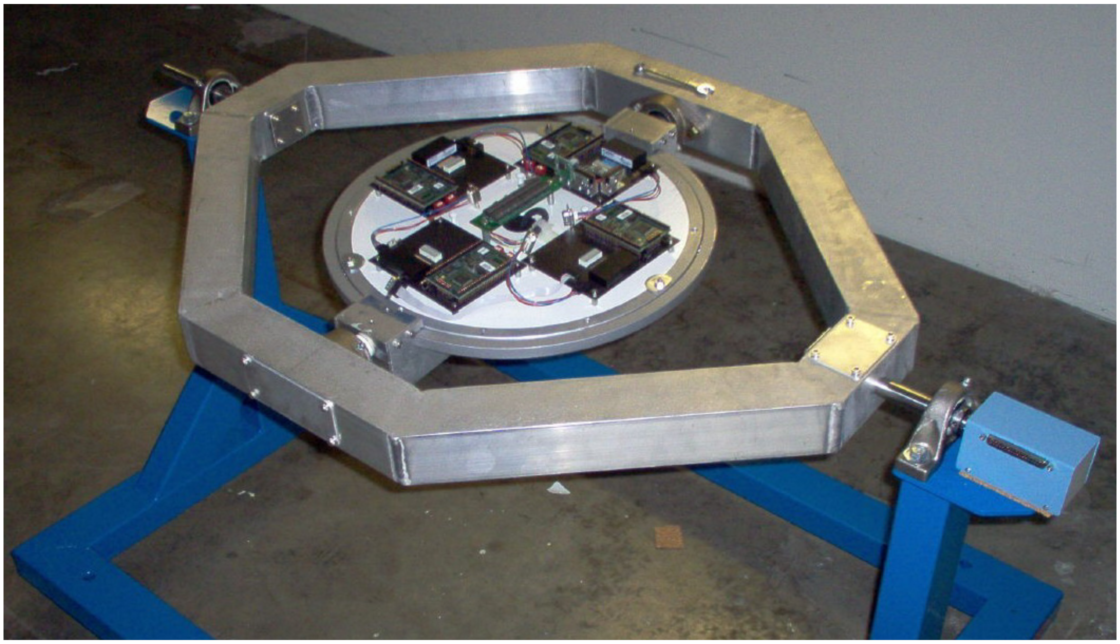


Figure 5 Microcosm PnP Demonstration Configuration

Following initialization, during which the RM is identified and other components on the networks "announce" themselves, the resource database is populated. This process demonstrates discovery and configuration, as well as resource management associated with the self-configuring network. The first step in the operational portion of the demonstration is to have the GN&C algorithms operate in a known, static state, making use of the "best available" sensor data on either of the networks. Next the attitude state is changed, using the rate table dynamics. The GN&C algorithms note a change in attitude, which prompts the use of actuators, lighting the light emitting diodes (LEDs). A second adjustment of attitude returns the rate table to the original static state. Once again, the GN&C algorithms note a change in attitude, eliminating the need to fire the actuators, demonstrated by turning off the LED(s), which demonstrates the adaptability of the GN&C algorithms based on atomic level GN&C sensor inputs.

To demonstrate reconfigurability of the GN&C algorithms, a sensor failure is simulated. The 2-axis gyro sensor is disconnected, alerting the RM that these data are no longer available. The GN&C algorithms automatically reconfigure the system to make use of data from a different source, namely, the single-axis micro-electro-mechanical systems (MEMS) gyros, demonstrating a graceful degradation of the overall system. While these data are less accurate, a solution is still achieved. The LED thrusters are fired momentarily to account for the perceived change in attitude, due to less accurate sensor inputs. The first attitude adjustment is re-applied, and the GN&C algorithms fire the LED thrusters, demonstrating the continued operation of the system in a degraded condition. The attitude is returned to the original nominal, static state.

Finally, the demonstration included aspects of "real-time" discovery as a new sensor, a three-axis gyro, is "plugged" into the system. The component announces itself to the RM. The GN&C algorithms note that a new, more accurate source of data is available and reconfigures to make use of these new data. Once again, the LED thrusters fire momentarily to account for the perceived change in attitude, due to more accurate sensor inputs. However, the system will quickly reach equilibrium, concluding the demonstration.

The Microcosm demonstration architecture, shown in Figure 6, includes actual hardware elements rather than simulations of the GN&C sensor data. The architecture provides a basis for demonstration, building on industry standard network medium and transceivers to allow focus on evaluation of the software-based products. By using COTS hardware and network implementations, driver and electrical protocols were leveraged to the fullest extent possible. Data formatting and protocols were developed to include salient aspects of both an avionics command response bus (i.e., Mil-Std-1553B) and an avionics point-to-point bus (i.e., ARINC 429). 8051 microprocessors were used to provide a front end to the MEMS sensors, providing the necessary value-added-processing to allow the components to "announce" themselves to the system.

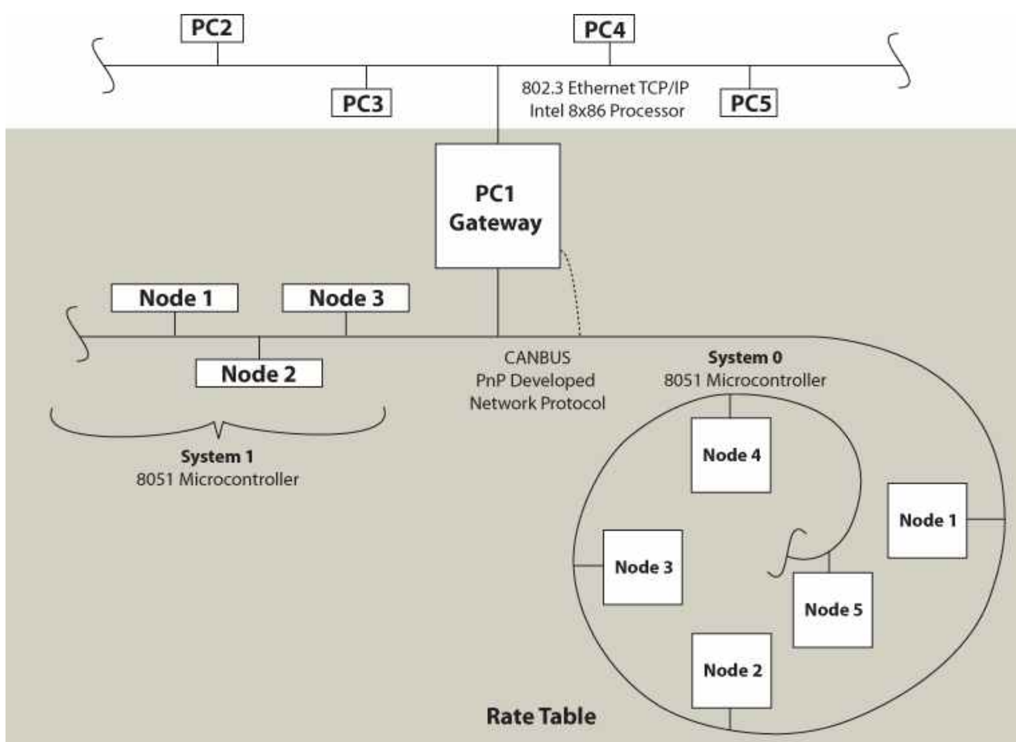


Figure 6 Microcosm Final Demonstration Architecture

In this implementation, the Microcosm team created a robust system that could accommodate the time synchronization and completely deterministic timing required for a GN&C implementation, while allowing for real-time discovery, resource management, and fault tolerance afforded by a PnP network. After evaluating the demonstration configuration, Microcosm believes that the innovative PnP approaches to GN&C algorithm partitioning and development are critical to the realization of operationally responsive space.

IMPLICATIONS FOR FUTURE GN&C SYSTEMS

As a result of studying the concept of PnP, applied to spacecraft systems, and developing a representative implementation of a PnP GN&C system, several implications for future GN&C systems can be derived. Key concepts of the PnP technology can be implemented within a GN&C subsystem to make the development process more modular. First, the use of standardized interfaces facilitates not only data transfer, but also discovery of data and selection of data elements, when multiple sources of like elements exist. Next, the partitioning of interface software, expanded to include network specific elements, protocol specific elements, component model/serial number data types and formatting, as well as component-to-spacecraft coordinate transformations, facilitate re-use and rapid integration. Additionally, this partitioning will facilitate the massaging of data to or from any component into atomic level data elements. While requiring some investment, not only by the spacecraft integrator, but also by suppliers of GN&C components, the components will evolve to increase their utility relative to the overall process. Finally, GN&C algorithms must be designed using an object oriented approach that abstracts the control laws from the actual source (sensor) or destination (actuator) of the data. This abstraction process is achieved by having the algorithms operate on atomic level data elements.

The use of standardized interfaces that allow for classic information transfer, as well as those additional components that provide necessary information for discovery, initialization, calibration, and data selection, allow GN&C components to be self-contained. This capability would include an interface definition for data, along with data representation and packing information, typically provided in a paper interface control document (ICD). Additionally, the inclusion of calibration data and component-to-spacecraft body coordinate transformation information, if known, allow the GN&C component to be operational, without the development of unique software for each component installation. Any information that is required for each atomic level data element to "announce" itself to the system should also be made available as part of a start-up or initialization interface definition. Included would be the quality and fidelity of the data or service being provided so that the application software (end user) can select the appropriate source or destination for its product, if multiple choices are available. Finally, the inclusion of other data that would typically be captured in a laboratory based engineering log book would help to enhance the functionality of the PnP concept in the spacecraft environment. All of the data can be captured in an XML extensible markup language (XML), or equivalent, structure that provides run-time parsing of the information. To attach a name to this electronic ICD the term extended transducer electronic data sheet (xTEDS) can be borrowed from instrumentation protocol.

Figure 7 illustrates PnP as applied to spacecraft GN&C components, highlighting that it is a disruptive technology. As a disruptive technology, the development of these PnP GN&C components may provide lower performance (good enough), but at lower cost. Over the course of time, through continued improvements, these PnP GN&C components will eventually replace traditional technologies, specifically mid-range components and with an emphasis on enhancing paper ICDs. Certainly, with respect to the traditional, deterministic arena of GN&C systems, PnP can be viewed as disruptive. The goal is to influence the process by demonstrating the "good enough" nature of a specific GN&C sensor component.

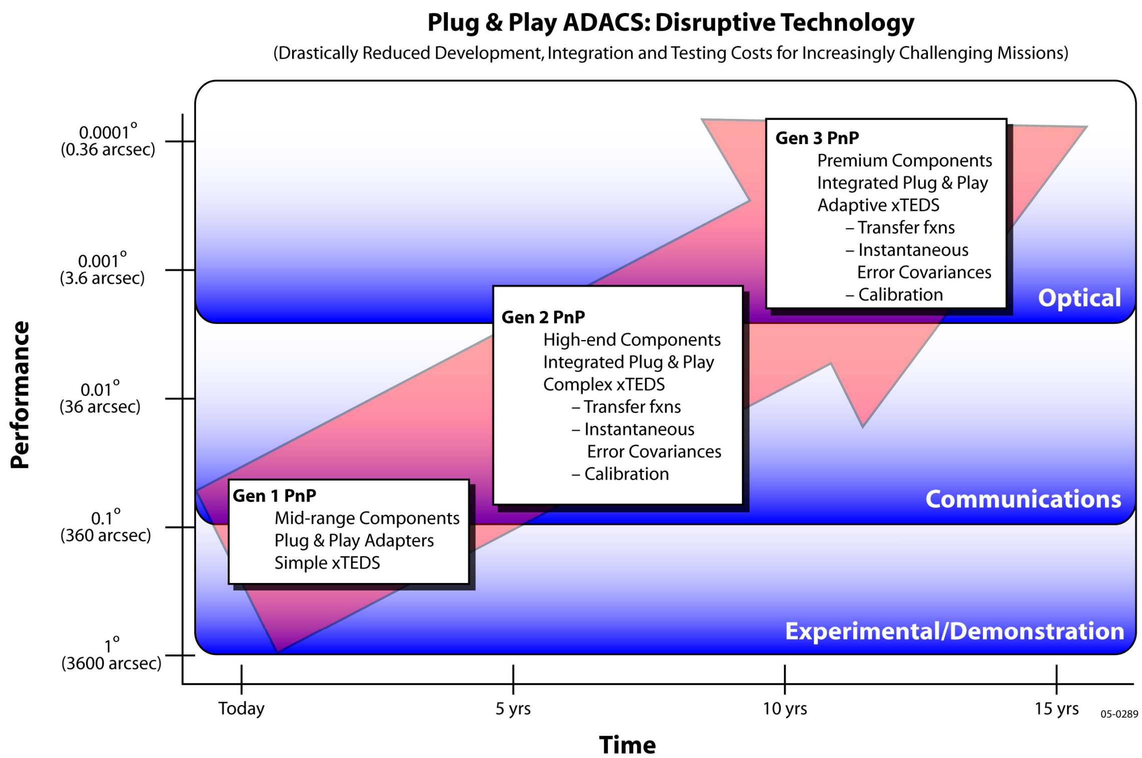


Figure 7 PnP, Specifically as it Applies to GN&C, is a Disruptive Technology that in Time, with Focused Effort, Will Realize its Full Potential.

All of these factors lead to new approaches to thinking about the design and integration of spacecraft systems, in particular, GN&C systems. Figure 8 illustrates a traditional GN&C architecture. Each of the sensors and actuators are independently selected, procured, and integrated, typically with entirely new input/output (I/O) handling software for each system, mission, or application. A filter and other sensor processing are created to optimize the data that are available, based on specific performance attributes that are provided to the engineer by the manufacture via an ICD. With each new spacecraft, different GN&C components, often from different suppliers or manufacturers, are selected to meet a unique characteristic of that specific mission. By-and-large, this approach has worked well in terms of performance and mission success; however, it is costly in terms of design, development, verification, integration, and test.

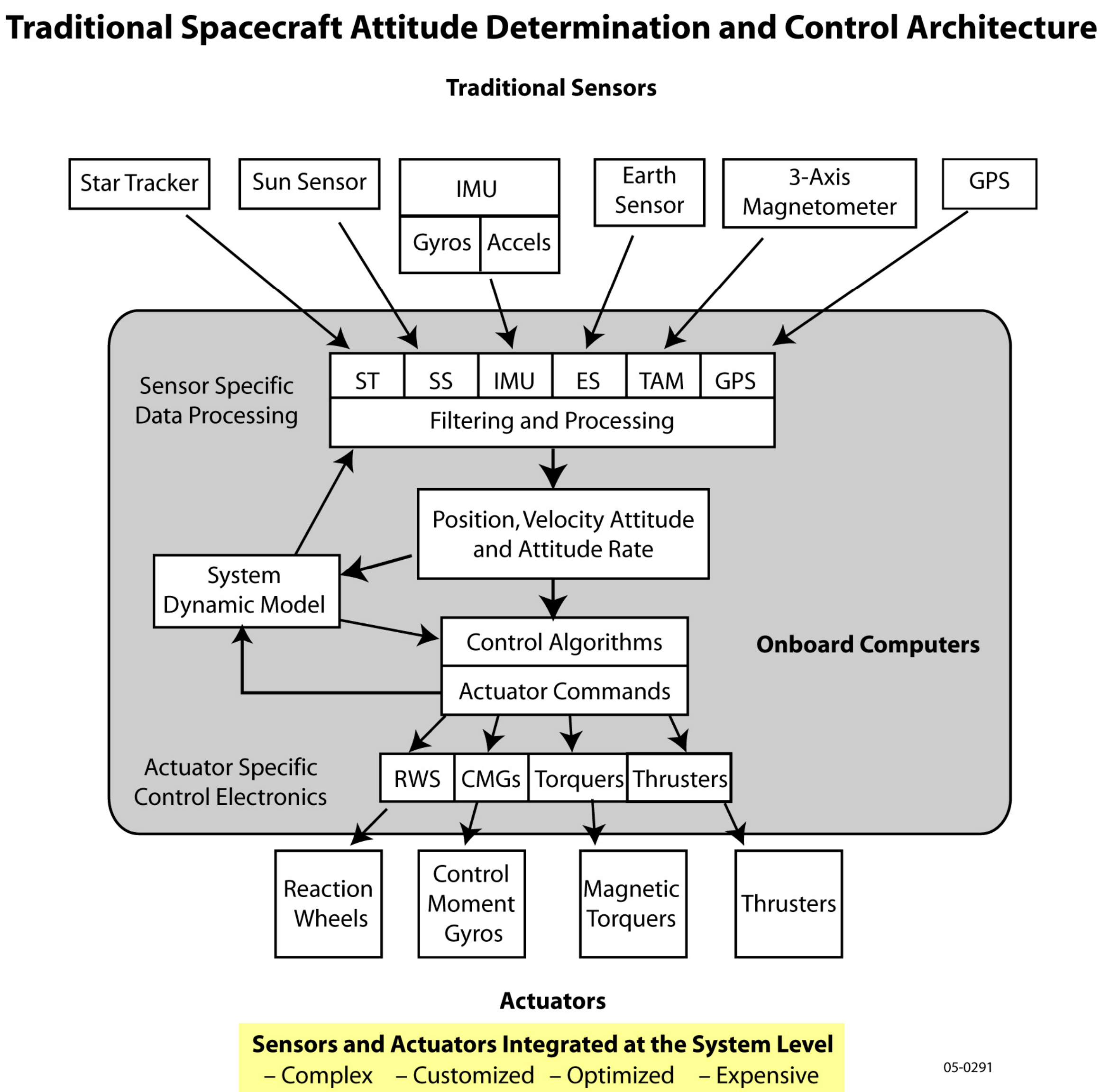


Figure 8 Traditional GN&C System Architecture with Custom I/O and Software

The economic drivers of PnP and modularity; truly low-cost launch, buses, and payloads supporting high utility missions, will change the way systems are designed and developed. To reduce overall costs, software must be reusable. In particular, GN&C algorithms must be able to support multiple components, from different suppliers, without requiring new development and most importantly, without requiring regression testing. The GN&C algorithms, encapsulated as software modules, need to be off-the-shelf (OTS), the same way the components are OTS. By abstracting the algorithms from the source of the data, using atomic level data elements, a PnP concept can make this reuse, and thus responsive space, a reality.

PnP GN&C approaches, like the one illustrated in Figure 9, make each software component stand-alone and facilitate the creation of helper application software that provides a mechanism for dealing with the network, protocol, component, or location specific elements aspects of the specific spacecraft configuration. This PnP GN&C approach makes make the entire GN&C architecture modular, enabling drastic reductions in cost and schedule by reducing the labor and scale of the design, development, verification, integration, and test efforts. Physical, network/messaging, and software interfaces are simplified and made routine, as are many of the verification and test processes.

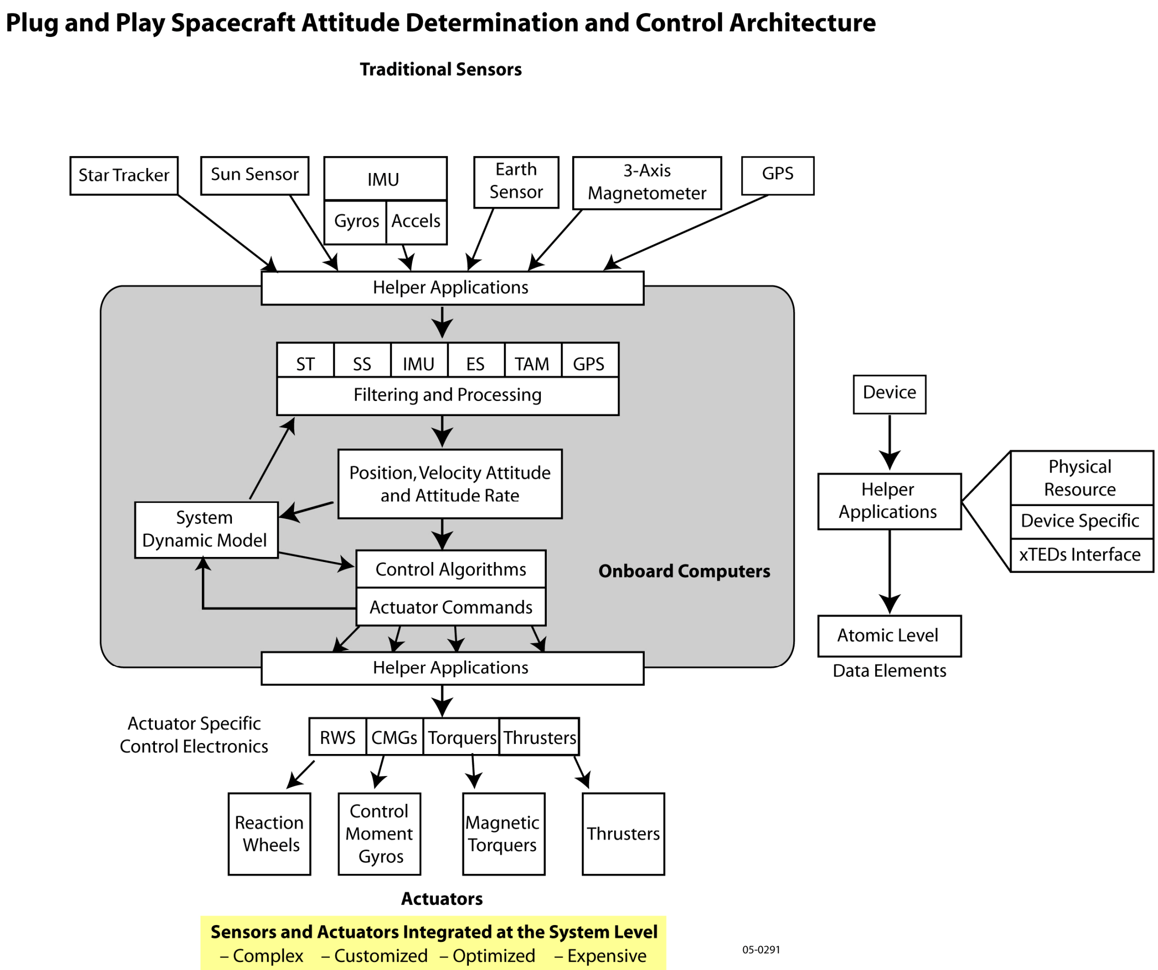


Figure 9 Modular GN&C System Architecture Supports Responsive Space

The process, shown previously in Figure 7, demonstrates a multi-generation development approach for achieving utility from PnP that initially provides “personality modules” to transform existing components into PnP components. When the traditional paper ICDs are replaced by XML-based xTEDS, and the HA software is created to abstract the core algorithms from the device select, the software can be “generic, yet adaptive, to the spacecraft mission and other selected components.

SUMMARY AND CONCLUSIONS

There is an emerging acceptance of responsive space as a means of allowing space-based assets to be quickly designed, developed, integrated, and deployed as evidenced by the development of AFRL's RST, the establishment of NRL's Operationally Responsive Space (ORS) / Integrated Systems Engineering Team (ISET), on-going TacSat activities, and the request for proposals for NASA's Commercial Orbital Transportation Services (COTS) demonstrations to provide rapid, continuous transportation between Earth and the International Space Station (ISS). Traditional spacecraft development cycles do not meet the goals and objectives of responsive space. A new paradigm is required that will challenge the traditional development processes, creating modular spacecraft elements (at the component or subsystem level), to enhance the viability of responsive space without compromising the performance of space-based assets. The application of PnP, a commercially accepted concept, to the process of spacecraft development can begin to refine the approach and methodologies that are used to produce these spacecraft. The Microcosm team has demonstrated, at a fundamental level, that PnP can be applied to GN&C spacecraft applications. It will take an increased level of effort to propel these concepts into the operational environment of spacecraft deployment, but the investment will enhance the probability of meeting the emerging requirements of responsive space.