Executive Summary

Fault Management Services for Reliable, Available, and Serviceable Systems (SRASS)

PNUM 12B

Background: The IEEE Portable Operating System Interface (POSIX) SRASS working group is in the process of establishing the IEEE POSIX 1003.1h standard. This draft standard is currently cited in the DoD Joint Technical Architecture (JTA) Document under emerging operating systems standards and these activities intend to elevate the POSIX 1003.1h standard to mandated status in the JTA.

The purpose of POSIX 1003.1h is to provide a set of APIs that allow a standard method for application software to create and access fault management information. Standard APIs allow application source code portability across POSIX-compliant operating systems (OSs) targeted to diverse platforms, resulting in reduced application software development and maintenance costs, as well as increased re-use and interoperability. POSIX standards can be used as technical architecture "building codes" for both information processing and weapon systems developed and used by the DoD.

The current draft of POSIX 1003.1h addresses the following areas:

- error/event logging and notification,
- core dump control,
- shutdown/reboot, and
- configuration space management.

Requirements: The requirements of these efforts are to further mature the POSIX 1003.1h standard and ensure that the standard can be used to satisfy DoD system requirements, particularly for weapon systems. The POSIX family of standards was originally written considering only commercial information processing systems. Extensions to the base standard, such as *POSIX Part 1: System Application Programmer Interface (API), Amendment 1: Real-time Extension* (POSIX 1003.1b) required DoD involvement for this working group to address real-time issues of concern in DoD weapon systems. Likewise, representatives of the DoD community must be involved in the development of other extensions to the base standard, such as POSIX 1003.1h, to ensure that the standards are written in such a way that does not preclude the fault management requirements of future DoD systems. Engineering data collected by the prototyping effort will be fed back into the POSIX 1003.1h working group to influence development of the standard, thereby making the standard more useful for DoD programs and possibly reducing the number of objections raised when the standard goes to ballot.

This effort encompasses three requirements:

1. Extend the configuration space management prototype developed under the OS-JTF funded Phase III of the Dynamic Reconfiguration Demonstration System (DRDS) program, and port the prototype to an open system testbed.
2. Support and provide feedback to the SRASS working group.
3. Generate a functional requirements specification for the fighter aircraft domain in support of the Virtual JTA effort.

The configuration space management prototype begun in Phase III of the DRDS program will be ported to a COTS testbed and expanded. The original prototype contained five of the twelve configuration space management operations defined in the draft standard. As part of this effort the remaining seven operations will be implemented and integrated with the existing operations. The configuration space management operations will be ported to an open system testbed. Results of this effort will be documented and presented to the SRASS working group to further refine the standard and provide a basis for eventual development of an Ada binding for the configuration space management operations.

Future Activities: The entire DRDS will be ported to a COTS testbed. Porting the DRDS technology to such a system will allow DRDS technology to be demonstrated in an open system environment containing COTS hardware and software. This new testbed can then be integrated into the F-18 cockpit simulation, thus providing

visual confirmation that the software can perform in a real-time environment. In parallel with the port to a COTS testbed, the DRDS will also be ported to Ada95. Demonstrating proposed open system standards on military testbeds is an important way to gain acceptance of open systems within the DoD community by actually demonstrating feasibility and utility of the standards. Documenting the lessons learned from these efforts will also benefit future programs in implementing an open system approach.