Certification Authorities Software Team (CAST)

Position Paper
CAST-2

Guidelines for Assessing Software Partitioning/Protection Schemes

Completed February, 2001

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, <u>it does not constitute official policy or guidance from any of the authorities</u>. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**

Guidelines for Assessing Software Partitioning/Protection Schemes

Abstract:

DO-178B/ED-12B includes provisions for a concept called partitioning. However, partitioning is actually an implementation of a more general concept called protection. Section 2.3.1 of DO-178B/ED-12B states "If ***protection*** by partitioning is provided …". However, this term is not consistent throughout the document. In many cases, partitioning is used whereas protection is intended (e.g., Sections 2.5b and 11.1b). Furthermore, DO-178B/ED-12B does not provide any guidance on assessing partitioning/protection strategies. The purpose of this paper is to clarify the differences between protection and partitioning and to provide some guidelines for evaluation of software partitioning/protection.

Key words:

safety, partitioning, protection, software level

1.0 Introduction

DO-178B/ED-12B includes provisions for a concept called partitioning. However, partitioning is actually an implementation of a more general concept called protection. Section 2.3.1 of DO-178B/ED-12B states "If ***protection*** by partitioning is provided …". However, this term is not consistent throughout the document. In many cases, partitioning is used whereas protection is intended (e.g., Sections 2.5b and 11.1b). Furthermore, DO-178B/ED-12B does not provide any guidance on assessing partitioning/protection strategies. The purpose of this paper is to clarify the differences between protection and partitioning and to provide some guidelines in their evaluation.

Partitioning is just one means of implementing the general concept of protection. Partitioning is method of separating components to ensure protection (section 2.3.1 of ED12B/DO-178B). The real issue is whether two or more components are protected from the actions of each other. In order to be able to discuss these concepts, a set of definitions is proposed and discussed. Then a set of evaluation considerations for partitioning/protection is proposed. Finally, other known information on partitioning/protection is identified for reference purposes.

2.0 Definitions

- **Strict Protection** - Component X can be said to be strictly protected from Y if any behavior of Y has no effect on the operation of X. An example of this type of protection would be two components within a line replaceable unit (LRU) with no communication between them.

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, <u>it does not constitute official policy or guidance from any of the authorities</u>. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**

- **Safety Protection** - Component X can be said to be safely protected from Y if any behavior of Y has no effect on the safety properties of X. An example of this would be the use of a Cyclic Redundancy Code around data passed through a non-assured data link. The only safety property of importance would be the corruption of data. Loss of data could not be a safety property of interest in this example. This approach requires the identification of the safety properties which can be derived from the safety analysis/hazard analysis.
- **Two-way protection** - Component X is protected from Y, and Y is protected from X. An example of this type of protection would be two components within a line replaceable unit (LRU) with no communication between them **One-way protection** - Component X is protected from Y, but component Y is not protected from X. An example of this would be a computer which can only receive ARINC 429 data from the autoland system. In this case the autoland software could affect the maintenance software but the maintenance software would not be able to interfere with the autoland software.

NOTE: Strict or safety protection can either be one-way or two-way.

By providing a vocabulary to categorize with the different types of protections, the certification engineer and the applicant can limit their evaluation to only the essential items needed to make their case for protection.

3.0 Guidelines for Evaluation of Protection Claims

A component can effect the operation of other components by effecting the temporal (time) behavior or the data (space) of the other components. The applicant should first categorize the type of protection claimed according to the definitions specified above. If the applicant's approach to protection is to separate (partition) components in both time and space, then the applicant is required to demonstrate the partitioning in time and space between the two components to demonstrate either one way or two way strict protection. However if the applicant proposes to use safety protection, then the applicant must identify all the safety properties of time and space which could be affected and then demonstrate that the safety properties have not been violated.

In evaluating time properties the following items should be considered as appropriate to the design and claims of the applicant.

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, it does not constitute official policy or guidance from any of the authorities. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**

3.1 Time

The following items can effect the time parameters of a program and need to be investigated to demonstrate that they either have no effect or that their effect is acceptable based on the identified safety parameters. This list is not intended to be all inclusive.

- Interrupts and interrupt inhibits (software and hardware)
- Loops (e.g. infinite loops)
- Real time correspondence:
  - frame overrun
  - interference with real time clock
  - counter/timer corruption
  - pipeline and caching

- Control Flow defects (timing aspects):
  - incorrect branching into a partition or protected area
  - corruption of a jump table (double duty?)
  - corruption of the processor sequence control
  - corruption of return addresses
  - unrecoverable hardware state corruption (e.g., mask and halt)

- Memory, I/O contention
- Data flags
- Software traps:
  - divide by zero
  - un-implemented instruction
  - specific software interrupt instructions
  - unrecognized instruction
- Recursion termination
- Indirect non terminating call loops
- Holdup commands (performance hedges)

DO-248B/ED-94B (to be published in mid-2001) will contain a discussion paper on partitioning/protection. (Note: this wording is based on SC-190/WG-52 plenary approved paper – it could have editorial changes.) The paper states that for partitioning/protection for shared computer resources, the following time aspects should be considered:

- *"Protection of the processing and communication assigned to a partition.*
- *The consistent order of execution between communicating partitions.*
- *Deterministic scheduling (processor and communication).*
- *Guaranteed access for each software partition to a prescribed set of hardware resources for a prescribed period of time and at a prescribed rate and, if necessary, at a prescribed point in time."*

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, <u>it does not constitute official policy or guidance from any of the authorities</u>. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**

3.2 Space

The following items can affect the space parameters of a program and need to be investigated to demonstrate that they either have no effect or that their effect is acceptable based on the identified safety parameters. This list is not intended to be all inclusive.

- Loss of input or output data
- Corruption of input or output data
- Corruption of internal data:
  - direct or indirect memory writes
  - table overrun
  - incorrect linking
  - calculations involving time
- Delayed data
- Program overlays
- Buffer sequence (double jeopordy)
- External device interaction (e.g. displays):
  - loss of data (e.g. overwritten)
  - delayed data
  - incorrect data (unlikely across systems)
  - protocol halts (e.g. ack nacks)
- Control Flow defects (space aspects):
  - incorrect branching into a partition or protected area
  - corruption of a jump table (double duty?)
  - corruption of the processor sequence control
  - corruption of return addresses
  - unrecoverable hardware state corruption (e.g., mask and halt)

The DO-248B/ED-94B discussion paper on partitioning/protection states that the following space aspects should be considered for partitioning/protection of shared computer resources (note: this wording is based on SC-190/WG-52 plenary approved paper – it could have editorial changes):

- "*Protection of code memory, data memory, registers, and input/output buffers.*
- *Persistent storage locations (e.g., data memory), assigned to a software partition, write-able only by that partition.*
- *Context data (e.g., processor registers, CPU-caches) used by a task preserved or flushed as appropriate when control is transferred to another partition.*
- *Data flow and communications between partitions.*"

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, <u>it does not constitute official policy or guidance from any of the authorities</u>. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**

3.3 Partitioning/Protection Violations

Section 11.10j of ED-12B/DO-178B indicates that the design must address the potential breaches of protection/partitioning. Adequate error control should be provided to contain breaching errors. DO-178B/ED-12B Section 6.3.3f and objective 13 of Table A-4 address verification of the partitioning/protection integrity.

4.0 Research Information

The FAA sponsored a research effort through NASA Langley to establish necessary and sufficient conditions for the evaluation of protection. Dr. John Rushby (of SRI International) completed a report in 1999 entitled, "Partitioning in Avionics Architecture: Requirements, Mechanism, and Assurance." The report can be found on both the SRI and NASA web-sites. The abstract from the report is included below:

"Automated aircraft control has traditionally been divided into distinct functions that are implemented separately (e.g., autopilot, auto-throttle, flight management); each function has its own fault-tolerant computer system, and dependencies among different functions are generally limited to the exchange of sensor and control data. A by-product of this "federated" architecture is that faults are strongly contained within the computer system of the function where they occur and cannot readily propagate to affect the operation of other functions.

More modern avionics architectures contemplate supporting multiple functions on a single, shared, fault-tolerant computer system where natural fault containment boundaries are less sharply defined. Partitioning uses appropriate hardware and software mechanisms to restore strong fault containment to such integrated architectures.

This report examines the requirements for partitioning, mechanisms for their realization, and issues in providing assurance for partitioning. Because partitioning shares some concerns with computer security, security models are reviewed and compared with the concerns of partitioning."

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, <u>it does not constitute official policy or guidance from any of the authorities</u>. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**

5.0 RTCA/EUROCAE Information

As previously alluded to, RTCA Special Committee #190 (SC-190) and EUROCAE Working Group #52 (WG-52) completed a discussion paper entitled "Partitioning Aspects in DO-178B/ED-12B." The paper will be published in DO-248B/ED-94B, the final SC-190/WG-52 report, in mid-2001. The paper provides some additional insight into the partitioning/protection subject.

Additionally, RTCA SC-182 and WG-48 completed a document entitled, "Requirements Specification for Avionics Computer Resource (ACR)" (DO-255/ED-96). This document provides some guidance regarding the need for partitioning/protection in ACRs (reference sections 1.2.2.1, 2.1, 2.5, and 3.11 of DO-255/ED-96).

NOTE: **This position paper has been coordinated among the software specialists of certification authorities from the United States, Europe, and Canada. However, <u>it does not constitute official policy or guidance from any of the authorities</u>. This document is provided for educational and informational purposes only and should be discussed with the appropriate certification authority when considering for actual projects.**