# A Risk Management Approach to CAIS Development

Hal Hart
Judy Kerner
Tony Alden
Frank Belz
Frank Tadman

TRW Defense Systems Group
Redondo Beach, California 90278

## ABSTRACT

The proposed DoD standard Common APSE Interface Set (CAIS) has been developed as a framework set of interfaces that will support the transportability and interoperability of tools in the support environments of the future. While the current CAIS version is a promising start toward fulfilling those goals and current prototypes provide adequate testbeds for investigations in support of completing specifications for a full CAIS, there are many reasons why the proposed CAIS might fail to become a usable product and the foundation of next-generation (1990's) project support environments such as NASA's Space Station software support environment. The most critical threats to the viability and acceptance of the CAIS include performance issues (especially in piggybacked implementations), transportability, and security requirements . To make the situation worse, the solution to some of these threats appears to be at conflict with the solutions to others.

TRW's CAIS development is a risk-managed approach planned to gather information early about critical threats, and, based on that information, to identify and pursue risk-reduction development approaches. This is an application of Barry Boehm's "Spiral Model" of the software development process, which integrates risk management into a generalization of systems development processes. Risk-managed approaches typically include prototyping to expedite acquisition of information in critical risk areas. TRW's initial assessment of risks led to a comprehensive design phase for the prototype before

coding based on two principal reasons:

1. the necessity to avoid a "narrow" prototype that accomplished some objectives while impeding others (or at least to reduce such conflicts in the initial implementation and to reduce and assess costs in expanding the prototype to serve broad risk-reduction objectives), and

2. incomplete information about how to accomplish that in a prototype (or even what the threats really were and hence what the objectives should be).

This prototype design phase was the first traversal of the Spiral Model. The near-term benefit of this approach is to direct initial prototyping activities toward areas with highest payoff in risk-reduction information while retaining compatibility with pursuance of other areas. The ultimate payoff of the TRW approach will not be in rapidness of prototype simulation of the initial CAIS, but in gathering information for specification and implementation of a viable 1990's CAIS (and perhaps even putting the CAIS prototype on the direct evolutionary path toward such a production-quality implementation).

Following are some of the risk-reduction directions determined by the TRW CAIS prototype design activities:

- **Performance:** a key fact is that the CAIS is more complex than typical 1980's operating systems, offering direct tool and user support in many areas not well (or directly) supported in most operating systems (e.g., configuration management support, inter-program communication and synchronization, access control, etc.). Early intense effort is needed in such key areas to develop efficient algorithms and/or architectures in these not-so-well-supported areas. Simulation has been identified as a time-saving approach to assess performance of newly developed CAIS algorithms or architectures without the complete expense of tool building or porting (and sometimes without completely implementing the CAIS algorithms). Additionally, the tough goal of piggybacked implementations (atop existing

operating systems) is aggravated by CAIS portability concerns

- **Transportability of CAIS Implementations:** the TRW CAIS design is based on a mapping of CAIS functionality directly to a machine-independent underlying model called the "tool portability layer". This means that most of the CAIS functionality can be implemented without regard to the underlying host. This approach isolates into the "inner portability layer" of the CAIS those functions that are most host-dependent. This ties in with the goal of efficiency by allowing development of host-dependent optimizations in the inner layer, and host-independent higher-level optimizations in the outer tool portability layer.

- **Security:** due to the time and expense of developing a certifiably secure CAIS (as on a bare machine), TRW's initial efforts will be investigations into using components from TRW's Army Secure Operating System (ASOS) project (scoped for A1) as a Trusted Computing Base upon which to implement the inner portability layer. This looks like a promising compromise between development costs of secure systems, and CAIS transportability and performance goals (because of reuse of the tool portability implementation layer and its optimizations).

As demonstrated in the list above, a risk-managed approach can find development strategies which simultaneously work toward solutions of the multiple critical threats to CAIS viability. A prototype implementation approach incorporating these is ongoing now, with a basic subset of the CAIS now implemented. Progress will be reviewed against the risk list later this year, at which time risks may be re-assessed, new alternative approaches hypothesized, and new directions selected based on information acquired in this phase of prototyping. This prototyping, risk re-assessment, and replanning will constitute another traversal of the Spiral Model.