

NOSC
NAVAL OCEAN SYSTEMS CENTER San Diego, California 92152-5000

Technical Document 1121
July 1987

DoD Requirements and Design Criteria for the Common APSE Interface Set (CAIS)

TRW Defense Systems Group



Approved for public release.
distribution is unlimited

The views and conclusions contained in this report are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Naval Ocean Systems Center or the U.S. Government

NAVAL OCEAN SYSTEMS CENTER

San Diego, California 92152-5000

E. G. SCHWEIZER, CAPT, USN
Commander

R. M. HILLYER
Technical Director

ADMINISTRATIVE INFORMATION

This work was performed for the U.S. Air Force, Ada Joint Program Office, 1211 S. Fern, Arlington, VA 22209, under program element 63226F. Contract N66001-86-D-0156 was carried out by TRW Defense Systems Group, One Space Park, Redondo Beach, CA 90278, under the direction of R.H. Mumm, Code 423, Naval Ocean Systems Center, San Diego, CA 92152-5000.

Released by
R.A. Wasilausky, Head
Software Engineering Technology Branch

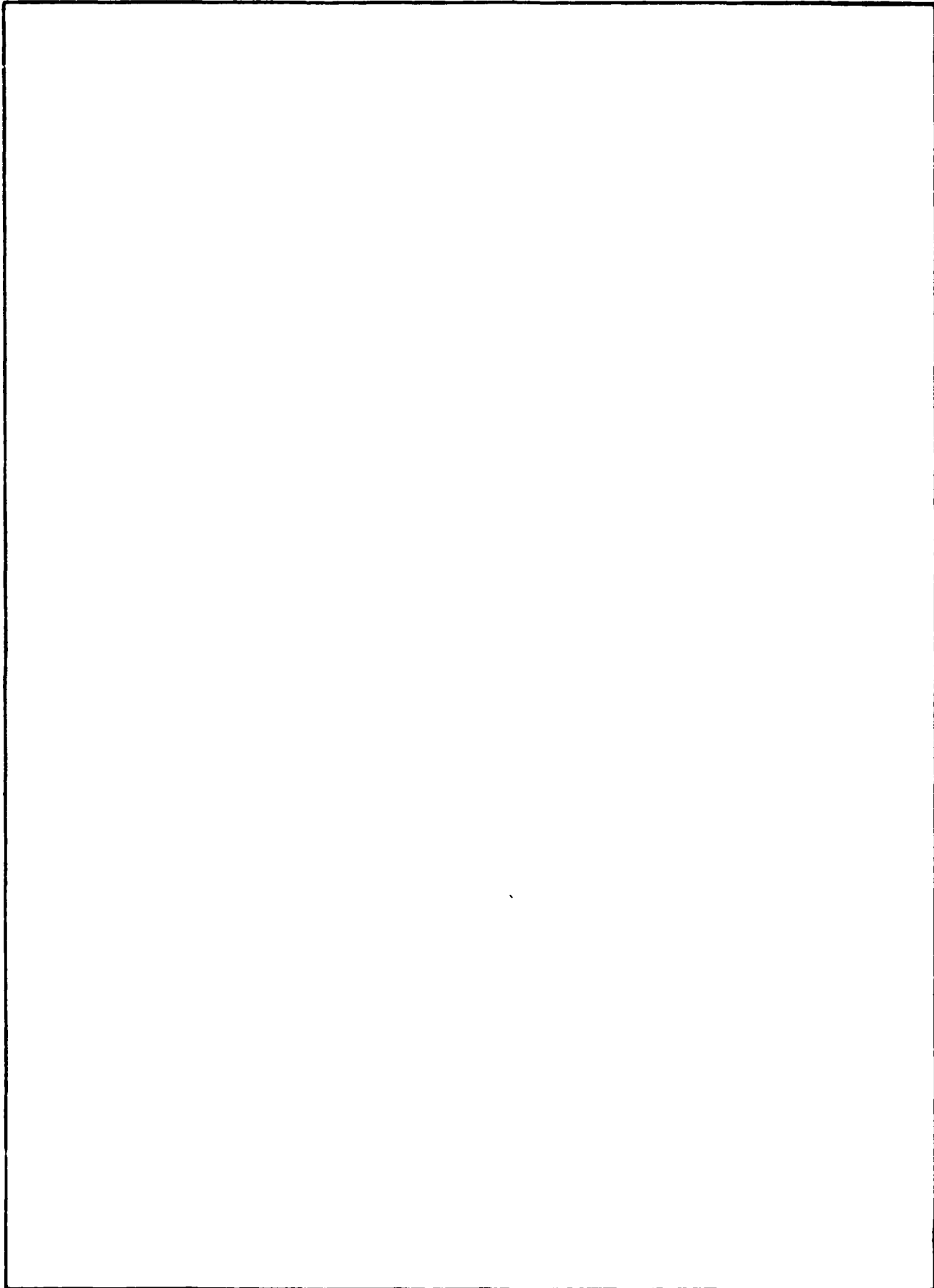
Under authority of
J.A. Salzmann, Head
Information Systems Division

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b RESTRICTIVE MARKINGS	
2a SECURITY CLASSIFICATION AUTHORITY		3 DISTRIBUTION AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.	
2b DECLASSIFICATION/DOWNGRADING SCHEDULE			
4 PERFORMING ORGANIZATION REPORT NUMBER(S)		5 MONITORING ORGANIZATION REPORT NUMBER(S) NOSC TD 1121	
6a NAME OF PERFORMING ORGANIZATION TRW Defense Systems	6b OFFICE SYMBOL <i>if applicable:</i>	7a NAME OF MONITORING ORGANIZATION Naval Ocean Systems Center	
6c ADDRESS (City, State and ZIP Code) One Space Park Redondo Beach, CA 90278		7b ADDRESS (City, State and ZIP Code) Software Engineering Technology Branch San Diego, CA 92152-5000	
8a NAME OF FUNDING SPONSORING ORGANIZATION U.S. Air Force	8b OFFICE SYMBOL <i>if applicable:</i> AJPO-NA	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N66001-86-D-0156	
8c ADDRESS (City, State and ZIP Code) Ada Joint Program Office 1211 S. Fern Arlington, VA 22209		10 SOURCE OF FUNDING NUMBERS	
		PROGRAM ELEMENT NO 63226F	PROJECT NO RDAF
		TASK NO 420-CS22	AGENCY ACCESSION NO DN288 534
11 TITLE (Include Security Classification) DoD Requirements and Design Criteria for the Common APSE Interface Set (CAIS)			
12 PERSONAL AUTHOR(S)			
13a TYPE OF REPORT Final	13b TIME COVERED FROM _____ TO Oct 86	14 DATE OF REPORT (Year, Month, Day) July 1987	15 PAGE COUNT 33
16 SUPPLEMENTARY NOTES			
17 DDC/SOAT CODES		18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB GROUP	
		Ada programming support environments (APSE)	
		Kernel APSE (KAPSE)	
		KAPSE interface team (KIT)	
19 ABSTRACT (Continue on reverse if necessary and identify by block number) <p>This document refines some of the DoD "Stoneman" requirements for Ada programming support environments (Buxton80) and imposes them upon a CAIS specification. The DoD "Steelman" requirements for high order computer programming languages (Fisher78) and the several sets of ANSI "OSCR" requirements and design objectives for operating system command and response languages (OSCR82) have also influenced this document.</p>			
20 CONTRIBUTION AVAILABILITY STATEMENT <input type="checkbox"/> UNCLASSIFIED <input checked="" type="checkbox"/> INTERNAL USE ONLY <input type="checkbox"/> FOR OFFICIAL USE ONLY		21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a NAME OF RESPONSIBLE PERSONAL R.H. Mumm		22b TELEPHONE (Include Area Code) 619-225-7401	22c OFFICE SYMBOL Code 423

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)



DD FORM 1473, 84 JAN

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DoD

Requirements

and

Design Criteria

for the Common APSE Interface Set (CAIS)

4 October 1986

Prepared by the
KAPSE Interface Team (KIT)
and the
KIT-Industry-Academia (KITIA)

Prepared for the
Ada¹ Joint Program Office (AJPO)
The Pentagon
1211 Fern 3D139
Washington, DC 20301-3081

Under NOSC Contract No. N66001-86-D-0156

¹Ada is a Registered Trademark of the U.S. Government, Ada Joint Program Office

Table of Contents

	PREFACE	vii
1. INTRODUCTION		1-1
1.1 Scope.		1-1
1.2 Terminology.		1-1
1.3 Relationship to Specifications & Implementations.		1-2
1.4 Reference Documents.		1-2
2. GENERAL DESIGN OBJECTIVES		2-1
2.1 Scope of the CAIS.		2-1
2.2 Basic Services.		2-1
2.3 Implementability.		2-1
2.4 Modularity.		2-1
2.5 Extensibility.		2-1
2.6 Technology Compatibility.		2-1
2.7 Uniformity.		2-2
2.8 Security.		2-2
2.9 Visible Distribution of CAIS Facilities		2-2
2.9A Reference.		2-2
2.9B Reallocation.		2-2
3. GENERAL SYNTAX AND SEMANTICS		3-1
3.1 Syntax		3-1
3.1A General Syntax.		3-1
3.1B Uniformity.		3-1
3.1C Name Selection.		3-1
3.1D Pragmatics.		3-1
3.2 Semantics		3-1
3.2A General Semantics.		3-1
3.2B Repeatability.		3-2
3.2C Exceptions.		3-2
3.2D Consistency.		3-2
3.2E Cohesiveness.		3-2
3.2F Pragmatics.		3-2
4. ENTITY MANAGEMENT SUPPORT		4-1
4.1 Entities, Relationships, and Attributes		4-2
4.1A Data.		4-2
4.1B Elementary Values.		4-2
4.1C System Integrity.		4-2
4.2 Typing		4-2
4.2A Types.		4-2
4.2B Rules about Type Definitions.		4-2
4.2C Type Definition.		4-3
4.2D Changing Type Definitions.		4-3

4.2E Triggering.	4-3
4.3 Identification	4-3
4.3A Exact Identities.	4-3
4.3B Identification.	4-3
4.3C Identification Methods.	4-3
4.4 Operations	4-4
4.4A Entity Operations.	4-4
4.4B Relationship Operations.	4-4
4.4C Attribute Operations.	4-4
4.4D Exact Identity Operations.	4-4
4.4E Uninterpreted Data Operations.	4-5
4.4F Synchronization.	4-5
4.4G Access Control.	4-5
4.4H Version (Revision and Variation) Support.	4-5
4.4I History Mechanism.	4-5
4.5 Transactions.	4-5
4.5A Transaction Mechanism.	4-5
4.5B Transaction Control.	4-5
4.5C System Failure.	4-5
4.6 Robustness and Restoration.	4-5
4.7 Common External Form.	4-6
5. PROGRAM EXECUTION FACILITIES	5-1
5.1 Activation of Program	5-1
5.1A Activation.	5-1
5.1B Unambiguous Identification.	5-1
5.1C Activation Data.	5-1
5.1D Dependent Activation.	5-1
5.1E Independent Activation.	5-1
5.2 Termination	5-1
5.2A Termination.	5-1
5.2B Termination of Dependent Processes.	5-2
5.2C Termination Data.	5-2
5.3 Communication	5-2
5.3A Data Exchange.	5-2
5.4 Synchronization	5-2
5.4A Task Waiting.	5-2
5.4B Parallel Execution.	5-2
5.4C Synchronization.	5-2
5.4D Suspension.	5-2
5.4E Resumption.	5-2
5.5 Monitoring	5-2
5.5A Identify Reference.	5-2
5.5B RTS Independence.	5-3
5.5C Instrumentation.	5-3

6. INPUT/OUTPUT	6-1
6.1 Tool-Device Driver Interfaces	6-1
6.1A Specified Interfaces.	6-1
6.1B Text Interfaces.	6-1
6.1C Graphical Tool Interfaces.	6-2
6.1D Device Driver Visibility.	6-2
6.1E Unsupported Features.	6-2
6.1F Device Driver Connection.	6-2
6.1G Device Driver Creation and Deletion.	6-2
6.1H Data Length.	6-2
6.1I Buffering.	6-2
6.1J Data Modifications.	6-2
6.1K Input Sampling.	6-2
6.1L Type-Ahead.	6-3
6.1M Echoing.	6-3
6.1N Control Input Traps.	6-3
6.1O Telecommunications Support.	6-3
6.2 Interfaces Supporting Device Drivers	6-3
6.2A Sufficiency.	6-3
6.2B Device Independence.	6-3
6.2C Basic Functions.	6-3
6.2D Exclusive Access.	6-3
6.2E Input Output Sequencing.	6-3
6.2F Transmission Characteristics.	6-4
6.2G Timeout.	6-4
6.2H Data Link Control.	6-4
7. GLOSSARY OF TERMS	7-1

PREFACE

The KAPSE Interface Team (KIT), and its companion Industry-Academia team (KITLA), were formed by a Memorandum of Agreement (MOA) signed by the three services and the Undersecretary of Defense in January, 1982. Their purpose is to contribute to the achievement of Interoperability of environment databases (of applications software) and Transportability of software development tools ("I&T"). These are economic objectives identified at the outset of the DoD common language initiative in the mid-1970's. Progress toward fulfilling these objectives is now acknowledged to require a level of commonality among Ada Programming Support Environments (APSEs), in addition to the standard language Ada [Ada83]. The core of the KIT/KITLA strategy to fulfill I&T objectives is to define a standard set of APSE interfaces ("CAIS" for "Common APSE Interface Set"), which augment the Ada language with the functionality needed to implement tools, thus improving the ability to share tools and databases between conforming APSEs. Note that a number of these interfaces are at the Kernel APSE (KAPSE) level, while others address a higher level of functionality. This document establishes requirements and design objectives (called "criteria") on the definition of a CAIS.

This document refines some of the DoD "Stoneman" Requirements for Ada Programming Support Environments [Buxton80] and imposes them upon a CAIS specification. The DoD "Steelman" Requirements for High Order Computer Programming Languages [Fisher78] and the several sets of ANSI "OSCRL" requirements and design objectives for Operating System Command and Response Languages [OSCRL82] have also influenced this document.

1. INTRODUCTION

1.1 Scope. This document provides the Department of Defense's requirements and design criteria for the definition and specification of a Common APSE Interface Set (CAIS) for Ada Programming Support Environments (APSEs).

1.2 Terminology. Precise and consistent use of terms has been attempted throughout the document.

Potentially ambiguous terms used in the document are defined in the Glossary of KIT/KITLA Terminology [KK85]. Some definitions tailored to the context of this document are provided in the last section of the document.

Additionally, the following verbs and verb phrases are used throughout the document to indicate where and to what degree individual constraints apply. Any sentence not containing one of the following verbs or verb phrases is a definition, explanation or comment.

"SHALL" indicates a requirement on the definition of the CAIS; sometimes "shall" is followed by "provide" or "support," in which cases the following two definitions supersede this one.

"SHALL PROVIDE" indicates a requirement for the CAIS to provide interface(s) with prescribed capabilities.

"SHALL SUPPORT" indicates a requirement for the CAIS to provide interface(s) with prescribed capabilities or for CAIS definers to demonstrate that the capability can be constructed from CAIS interfaces.

"SHOULD" indicates a desired goal (design criterion) but one for which there is no objective test.

1.3 Relationship to Specifications & Implementations. This document specifies functional capabilities which are to be provided in the semantics of a CAIS specification and are therefore to be provided by conforming CAIS implementations. In general, the specifications of software fulfilling those capabilities (and decisions about including or not including CAIS interfaces for certain capabilities as suggested by the "shall support" definition in the previous section) are delegated to the CAIS definers. If a CAIS implementor determines that it is feasible, then the CAIS implementor may provide a particular specified CAIS facility by reusing other CAIS facilities, thereby achieving a "layered implementation" of the CAIS. Therefore, the realization of a specific CAIS implementation is the result of intentionally divided decision-making authority among 1) this requirements document, 2) CAIS definers, and 3) CAIS implementors.

1.4 Reference Documents.

MILITARY STANDARDS

[Ada83] Reference Manual for the Ada Language, ANSI/ML-STD-1815A, January 1983.

OTHER GOVERNMENT DOCUMENTS

[Buxton80] "Stoneman" DoD Requirements for Ada Programming Support Environments, February 1980.

[Fisher78] "Steelman" DoD Requirements for High Order Computer Programming Languages, June 1978.

[KK85] Glossary of KIT/KITIA Terminology, draft 1985.

[TCSEC83] Trusted Computer System Evaluation Criteria, CSC-STD-001-83, DoD Computer Security Center, August 15, 1983.

NON-GOVERNMENT DOCUMENTS

[OSCRL82] Operating System Command and Response Languages, proposed ANSI standard drafts, 1982.

2. GENERAL DESIGN OBJECTIVES

2.1 Scope of the CAIS. The CAIS shall provide interfaces sufficient to support the use of APSEs for wide classes of projects throughout their lifecycles and to promote I&T of tools and data between APSEs.

2.2 Basic Services. The CAIS should provide simple-to-use mechanisms for achieving common, simple actions. Facilities which support needs of less frequently used tools should be given secondary consideration.

2.3 Implementability. The CAIS specification shall be machine independent and implementation independent. The CAIS shall be implementable on bare machines and on machines with any of a variety of operating systems. The CAIS shall contain only interfaces which provide facilities which have been demonstrated in existing commercial or military software systems. CAIS features should be chosen to have a simple and efficient implementation in many machines, to avoid execution costs for unneeded generality, and to ensure that unused portions of a CAIS implementation will not add to execution costs of a non-using tool. The measures of the efficiency criterion are, primarily, minimum turnaround time for CAIS basic services used by APSE tools and, secondarily, consumption of resources.

2.4 Modularity. Interfaces should be partitioned such that the partitions may be understood independently and there are no undocumented dependencies between partitions.

2.5 Extensibility. The design of the CAIS should facilitate development and use of extensions of the CAIS; i.e., CAIS interfaces should be reusable so that they can be combined to create new interfaces and facilities.

2.6 Technology Compatibility. The CAIS shall adopt existing standards where applicable. For example, recognized standards for device characteristics are provided by ANSI, ISO, IEEE, and DoD.

2.7 Uniformity. A small number of unifying conceptual models should underlie the CAIS. All CAIS features should uniformly address aspects such as status returns, exceptional conditions, parameter types, and options.

2.8 Security. The CAIS shall provide interfaces to allow tools to operate within a Trusted Computer System (TCS) that meets the Class B3 criteria as defined in [TCSEC83]. Specifically:

- a. It shall be possible to implement the CAIS within a TCS.
- b. When implemented within a TCS, the CAIS shall support the use of the security facilities provided by the Trusted Computing Base (TCB) to applications programs.
- c. When not implemented within a TCS, the CAIS interfaces sensitive to security shall operate as a dedicated secure system (i.e., all data at a single security level, and all subjects cleared to at least that level).

2.9 Visible Distribution of CAIS Facilities

2.9A Reference. The CAIS shall provide a means for tools to refer to distinct physical resources (both computational and storage) that are used to implement specific CAIS facilities.

2.9B Reallocation. The CAIS shall provide a means to control (or influence) the manner in which the physical resources are associated with specific CAIS facilities.

3. GENERAL SYNTAX AND SEMANTICS

3.1 Syntax

3.1A General Syntax. The syntax of the CAIS interfaces shall be expressed in Ada package specifications. The syntax of the CAIS interfaces shall conform to the character set as defined by the Ada standard (section 2.1 of ANSI/MIL-STD-1815A [Ada83]).

3.1B Uniformity. The CAIS should employ uniform syntactic conventions and should not provide several notations for the same concept. CAIS syntax issues (including, at least, limits on name lengths, abbreviation styles, other naming conventions, relative ordering of input and output parameters, etc.) should be resolved in a uniform and integrated manner for the whole CAIS.

3.1C Name Selection. The CAIS should avoid coining new words (literals or identifiers) and should avoid using words in an unconventional sense. Ada identifiers (names) defined by the CAIS should be natural language words or industry accepted terms whenever possible. The CAIS should define Ada identifiers which are visually distinct and not easily confused (including, at least, that the CAIS should avoid defining two Ada identifiers that are only a 2-character transposition away from being identical). The CAIS should use the same name everywhere in the interface set, and not its possible synonyms, when the same meaning is intended.

3.1D Pragmatics. The CAIS should impose only those restrictive rules or constraints required to achieve I&T. CAIS implementors will be required to provide the complete specifications of all syntactic restrictions imposed by their CAIS implementations.

3.2 Semantics

3.2A General Semantics. The CAIS shall be completely and unambiguously defined. The specification of semantics should be both precise and understandable. The semantic specification of each CAIS interface shall include a precise statement of assumptions (including execution-time preconditions for calls), effects on global data and packages, and interactions with other interfaces.

3.2B Repeatability. Every time a CAIS interface is called under the same circumstances, it should return the same response.

3.2C Exceptions. The CAIS interfaces shall employ the mechanism of Ada exceptions to report exceptional situations that arise in the execution of CAIS facilities. The CAIS specification shall include exceptions (with visible declarations) for all situations that violate the preconditions specified for the CAIS interfaces. The CAIS specification shall include exceptions (with visible declarations) that cover all violations of implementation-defined restrictions.

3.2D Consistency. The description of CAIS semantics should use the same word or phrase everywhere, and not its possible synonyms, when the same meaning is intended.

3.2E Cohesiveness. Each CAIS interface should provide only one service.

3.2F Pragmatics. The CAIS specification shall enumerate all aspects of the meanings of CAIS interfaces and facilities which must be defined by CAIS implementors. CAIS implementors will be required to provide the complete specifications for these implementation-defined semantics.

4. ENTITY MANAGEMENT SUPPORT

Access controls and security rights will apply to all CAIS facilities required in this section.

The general requirements for the CAIS entity management support are the following.

- a. There shall be a means for retaining data.
- b. There shall be a way for retaining relationships among and properties of data.
- c. There shall be a way of operating upon data, deleting data, and creating new data.
- d. There shall be a means for defining certain operations and conditions as legal, for enforcing the definitions, and for accepting additional definitions of legality.
- e. There shall be a means to describe data, and there shall be a means to operate upon such descriptions. Descriptions of the data shall be distinguished from the data described.
- f. There shall be a way to develop new data descriptions by inheriting (some of) the properties of existing data descriptions.
- g. The relationships and properties of data shall be separate from the existence of the data instances.
- h. The descriptions of data and the instances of data shall be separate from the tools that operate upon them.
- i. The data facilities shall be sufficient to support Ada program libraries.

This characterization (subsections 4.1 - 4.7) of Entity Management Support is based on the Stoneman [Buxton80] requirements for a database, using a model based on the entity-relationship concept. Although a CAIS design meeting these requirements is expected to demonstrate the characteristics and capabilities reflected here, it is not necessary that such a design directly employ this entity-relationship model.

The entity-relationship model, for which definitions and requirements follow in 4.1 - 4.7, fulfills these requirements, and any alternative data model shall fulfill these requirements and shall also fulfill the equivalent of the requirements in 4.1 through 4.7.

4.1 Entities, Relationships, and Attributes

4.1A Data. The CAIS shall provide facilities for representing data using entities, attributes or binary relationships. The CAIS may provide facilities for more general N-ary relationships, but it is not required to do so.

4.1B Elementary Values. The CAIS shall provide facilities for representing data as elementary values.

4.1C System Integrity. The CAIS facilities shall ensure the integrity of the CAIS-managed data. Some of these facilities are access control, concurrency control, database backup and restoration, and transactions.

4.2 Typing

4.2A Types. The facilities provided by the CAIS shall enforce typing by providing that all operations conform to the type definitions. Every entity, relationship and attribute shall have one and only one type.

4.2B Rules about Type Definitions. The CAIS type definitions shall

- specify the entity types and relationship types to which each attribute type may apply
- specify the type or types of entities that each relationship type may connect and the attribute types allowed for each relationship type
- specify the set of allowable elementary values for each attribute type
- specify the relationship types and attribute types for each entity type
- permit relationship types that represent either functional mappings (one-to-one or many-to-one) or relational mappings (one-to-many or many-to-many)
- permit multiple distinct relationships among the same entities
- impose a lattice structure on the types which includes inheritance of attributes, attribute value ranges (possibly restricted), relationships and allowed operations.

4.2C Type Definition. The CAIS shall provide facilities for defining new entity, relationship and attribute types.

4.2D Changing Type Definitions. The CAIS shall provide facilities for changing type definitions. These facilities shall be controlled such that data integrity is maintained.

4.2E Triggering. The CAIS shall provide a conditional triggering mechanism so that prespecified procedures or operations (such as special validation techniques employing multiple attribute value checking) may be invoked whenever values of indicated attributes change. The CAIS shall provide facilities for defining such triggers and the operations or procedures which are to be invoked.

4.3 Identification

4.3A Exact Identities. The CAIS shall provide exact identities for all entities. The CAIS shall support exact identities for all relationships. The exact identity shall be unique within an instance of a CAIS implementation, and the CAIS shall support a mechanism for the utilization of exact identities across all CAIS implementations.

4.3B Identification. The CAIS shall provide identification of all entities, attributes and relationships. The CAIS shall provide identification of all entities by their exact identity. The CAIS shall support identification of all relationships by their exact identity.

4.3C Identification Methods. The CAIS shall provide identification of entities and relationships by at least the following methods:

- identification of some "start" entity(s), the specification of some relationship type and the specification of some predicate involving attributes or attribute types associated with that relationship type or with some entity type. This method shall identify those entities which are related to the identified start entity(s) by relationships of the given relationship type and for which the predicate is true. Subject to the security constraints of section 2.8, all relationships and entities shall be capable of identification via this method, and all attributes and attribute types (except uninterpreted data) shall be permitted in the predicates.
- identification of an entity type or relationship type and specification of some predicate on the value of any attribute of the entity type or relationship type. This method shall identify those entities or relationships of the given

type for which the predicate is true. Subject to the security constraints of section 2.8, all attributes (except uninterpreted data) shall be permitted in the predicates.

4.4 Operations

4.4A Entity Operations. The CAIS shall provide facilities to:

- create entities
- delete entities
- examine entities (by examining their attributes and relationships)
- modify entities (by modifying their attributes)
- identify entities (as specified in Section 4.3)

4.4B Relationship Operations. The CAIS shall provide facilities to:

- create relationships
- delete relationships
- examine relationships (by examining their attributes)
- modify relationships (by modifying their attributes)
- identify relationships (as specified in Section 4.3)

4.4C Attribute Operations. The CAIS shall provide facilities to:

- examine attributes
- modify attributes

4.4D Exact Identity Operations. The CAIS shall provide facilities to:

- pass exact identities between processes
- compare exact identities

4.4E Uninterpreted Data Operations. The CAIS shall provide that use of the input-output facilities of the Ada language (as defined in Chapter 14 of ANSI/MIL-STD-1815A [Ada83]) results in reading/writing an uninterpreted data attribute of an entity. The facilities of Section 6 shall then apply.

4.4F Synchronization. The CAIS shall provide dynamic access synchronization mechanisms to individual entities, relationships and attributes.

4.4G Access Control. The CAIS shall provide selective prohibition of operations on entities, relationships, and attributes being requested by an individual.

4.4H Version (Revision and Variation) Support. The CAIS shall provide a mechanism for specifying and implementing policies of versioning, in which multiple versions of entities are maintained and access by default to a preferred or current version is allowed.

4.4I History Mechanism. The CAIS shall support a mechanism for collecting and utilizing history. The history mechanism shall provide sufficient information to support comprehensive configuration control.

4.5 Transactions.

4.5A Transaction Mechanism. The CAIS shall support a transaction mechanism. The effect of running transactions concurrently shall be as if the concurrent transactions were run serially.

4.5B Transaction Control. The CAIS shall support facilities to start, end and abort transactions. When a transaction is aborted, all effects of the designated sequence of operations shall be as if the sequence were never started.

4.5C System Failure. System failure while a transaction is in progress shall cause the effects of the designated sequence of operations to be as if the sequence were never started.

4.6 Robustness and Restoration. The CAIS shall support facilities which ensure the robustness of and ability to restore CAIS-managed data. The facilities shall include at least those required to support the backup and archiving capabilities provided by modern operating systems.

4.7 Common External Form.

The CAIS shall specify a representation on external media of a set of related data entities (as described in section 4), to be known as the Common External Form. The CAIS shall support the transfer of data from the entity system of Section 4 to external media in this form. All information (including relationships and values) in the part of the entity system transferred shall be preserved on the external medium in the Common External Form. The CAIS shall support the transfer of data from an external medium in the Common External Form to the entity management system of Section 4. The CAIS shall preserve information on such a transfer to the maximum extent, given the possibility of different representations of data on the systems involved.

5. PROGRAM EXECUTION FACILITIES

Access controls and security rights will apply to all CAIS facilities required by this section.

5.1 Activation of Program

5.1A Activation. The CAIS shall provide a facility for a process to create a process for a program that has been made ready for execution. This event is called activation.

5.1B Unambiguous Identification. The CAIS shall provide facilities for the unambiguous identification of a process at any time between its activation and deactivation; one such capability shall be as an indivisible part of activation. This act of identification establishes a reference to that process. Once such a reference is established, that reference will refer to the same process until the reference is dissolved. A reference is always dissolved upon termination of the process that established the reference. A terminated process may not be deactivated while there are references to that process.

5.1C Activation Data. The CAIS shall provide a facility to make data available to a program upon its activation.

5.1D Dependent Activation. The CAIS shall provide a facility for the activation of programs that depend upon the activating process for their existence.

5.1E Independent Activation. The CAIS shall provide a facility for the activation of programs that do not depend upon the activating process for their existence.

5.2 Termination

5.2A Termination. The CAIS shall provide a facility for a process to terminate a process. There shall be two forms of termination; the voluntary termination of a process (termed completion) and the abnormal termination of a process. Completion of a process is always self-determined, whereas abnormal termination may be initiated by other processes.

5.2B Termination of Dependent Processes. The CAIS shall support clear, consistent rules defining the termination behavior of processes dependent on a terminating process.

5.2C Termination Data. The CAIS shall provide a facility for termination data to be made available. This data shall provide at least an indication of success or failure for processes that complete. For processes that terminate abnormally the termination data shall indicate abnormal termination.

5.3 Communication

5.3A Data Exchange. The CAIS shall provide a facility for the exchange of data among processes.

5.4 Synchronization

5.4A Task Waiting. The CAIS shall support task waiting.

5.4B Parallel Execution. The CAIS shall provide for the parallel execution of processes.

5.4C Synchronization. The CAIS shall provide a facility for the synchronization of cooperating processes.

5.4D Suspension. The CAIS shall provide a facility for suspending a process.

5.4E Resumption. The CAIS shall provide a facility to resume a process that has been suspended.

5.5 Monitoring

5.5A Identify Reference. The CAIS shall provide a facility for a process to determine an unambiguous identity of a process and to reference that process using that identity.

5.5B RTS Independence. CAIS program execution facilities should be designed to require no additional functionality in the Ada Run-Time System (RTS) from that provided by Ada semantics. Consequently, the implementation of the Ada RTS shall be independent of the CAIS.

5.5C Instrumentation. The CAIS shall provide a facility for a process to inspect and modify the execution environment of another process. This facility is intended to promote support for portable debuggers and other instrumentation tools.

6. INPUT/OUTPUT

Access controls and security rights will apply to all CAIS facilities required by this section.

The requirements specified in this section pertain to input/output between/among processes, data entities, communication devices, and storage devices, unless otherwise stated.

Input and output are defined in terms of device drivers. The following requirements are divided between those required of interfaces to tools, and those required by implementors of device drivers.

6.1 Tool-Device Driver Interfaces

6.1A Specified Interfaces. The CAIS shall specify tool-driver interfaces for at least the following logical devices:

- magnetic tape in ANSI format
- paper tape including precise hole placement
- serial text
- positional text
- graphical write-once
- graphical rewritable
- window manager

6.1B Text Interfaces. The text interfaces shall support control of at least the following attributes of text: margins, page width, page length, boldness, slant, justification, underlining, type size, color, and line spacing. The positional text interface shall permit locator input.

6.1C Graphical Tool Interfaces. The graphical interfaces shall support at least the description by tools of geometrical figures, and of complete pixel-detailed illustrations. There shall be specific interfaces for line drawings in geometrical form, with mechanisms for area fill. There shall be specific interfaces for including text strings in graphical output. The graphical rewritable interface shall permit locator input.

6.1D Device Driver Visibility. The CAIS shall provide interfaces which enable a tool to determine if a device driver is available.

6.1E Unsupported Features. The CAIS shall provide interfaces to control the consequences when the underlying device does not have all of the features required by the device driver.

6.1F Device Driver Connection. The CAIS shall provide interfaces by which a tool can connect to a device driver. This shall permit at least static, and may permit dynamic association between a tool and a device driver.

6.1G Device Driver Creation and Deletion. The CAIS shall provide interfaces which permit the addition and removal of device drivers.

6.1H Data Length. The CAIS shall specify reasonable limits on the length of data items to be communicated across the interfaces it specifies, and shall require all implementations to support to these limits.

6.1I Buffering. The CAIS shall support the clearing of any output buffers, both with and without forced processing of their contents. The CAIS shall support the clearing of input buffers. The CAIS shall support the input of character data such that each character is received when it is transmitted, without waiting for any other condition.

6.1J Data Modifications. The CAIS shall support control of character insertion (padding), character deletion (filtering), and character replacement (modification) in its text interfaces.

6.1K Input Sampling. The CAIS shall support sampling an input device for data without waiting due to an absence of data.

6.1L Type-Ahead. The CAIS shall support device driver interrogation and control of type-ahead.

6.1M Echoing. The CAIS shall support device driver interrogation and control of echoing.

6.1N Control Input Traps. The CAIS shall support the identification of a text device as a control device, i.e. a device for which certain sequences of input data represent a control input trap. The CAIS shall support selection of the sequences and their consequences.

6.1O Telecommunications Support. The CAIS shall support a telecommunications interface for data transmission.

6.2 Interfaces Supporting Device Drivers

6.2A Sufficiency. The interfaces provided for the implementation of device drivers shall be sufficient for the implementation of all of the device driver interfaces required by section 6.1.

6.2B Device Independence. The specifications of the interfaces required by this section shall not be dependent on particular devices.

6.2C Basic Functions. The CAIS shall provide interfaces by which a device driver can send and receive data, including bit maps and characters, to and from devices, and receive and exercise asynchronous control.

6.2D Exclusive Access. The CAIS shall provide a means for a device driver to obtain exclusive access to a device, either a physical device or a device driver. Such exclusive access does not require the exclusion of processes which, in a particular installation or implementation, cannot be prevented from intruding.

6.2E Input Output Sequencing. The CAIS shall provide facilities to enable a device driver to ensure the servicing of output requests in the order of their invocation, the processing of input in temporal order, and the proper sequencing of input and output.

6.2F Transmission Characteristics. The CAIS shall support device driver inquiry and control of at least baud rate, parity, number of bits, and full/half duplex.

6.2G Timeout. The CAIS shall provide facilities to permit timeout on input and output operations.

6.2H Data Link Control. The CAIS shall support facilities for the dynamic control of data links, including, at least, self-test, automatic dialing, hang-up, and broken-link handling.

7. GLOSSARY OF TERMS

ACTIVATE	to create a CAIS process. The activation of a program binds that program to its execution environment, which are the resources required to support the process's execution, and includes the program to be executed. The activation of a process marks the earliest point in time which that process can be referenced as an entity within the CAIS environment.
ARCHIVE	a subset of the CAIS-managed data that has been relegated to backing storage media while retaining the integrity, consistency and availability of all information in the entity management system.
ATTRIBUTE	an association of an entity or relationship with an elementary value.
BACKUP	a redundant copy of some subset of the CAIS-managed data. The subset is capable of restoration to active use by a CAIS implementation, particularly in the event of a loss of completeness or integrity in the data in use by implementation.
CONSUMER	an entity that is receiving data units via a datapath.
DATA UNIT	a representation of a value of an Ada discrete type.
DATAPATH	the mechanism by which data units are transmitted from a producer to a consumer.
DEACTIVATE	to remove a terminated process so that it may no longer be referenced within the CAIS environment.
DEVICE DRIVER	a computer program fragment responsible for converting a device independent information representation or protocol to a device dependent representation or protocol.
ELEMENTARY VALUE	one of two kinds of representations of data: interpreted and uninterpreted.
ENTITY	a representation of a person, place, event or thing.

EXACT IDENTITY

a designation of an entity (or relationship) that is always associated with the entity (or relationship) that it designates. This exact identity will always designate exactly the same entity (or relationship), and it cannot be changed.

FACILITY

a service obtained by using one or more CAIS interfaces.

HISTORY

a recording of the manner in which entities, relationships and attribute values were produced and of all information which was relevant in the production of those entities, relationships or attribute values.

IDENTIFICATION

a means of specifying the entities, relationships and attributes to be operated on by a designated operation.

INTEGRITY

preservation of conformance of the structure and contents of data to rules established by a particular APSE as defined by the CAIS specification, implementation-defined CAIS values and parameters, APSE administrators, and users.

INTERFACE

a specification of an individual service which is both provided by the CAIS and directly usable by APSE tools.

INTERPRETED DATA

a data representation whose structure is controlled by CAIS facilities and may be used in the CAIS operations.

MONITOR

to observe (or measure) the behavior or value of a process, operation, or data.

PROCESS

the CAIS facility used to represent the execution of any program.

PRODUCER

an entity that is transmitting data units via a datapath.

PROGRAM

a set of compilation units, one of which is a subprogram called the "main program." Execution of the program consists of execution of the main program, which may invoke subprograms declared in the compilation units of the program.

RELATIONSHIP

an ordered connection or association among entities. A relationship among N entities (not necessarily distinct) is known as an "N-ary" relationship.

RESOURCE	any capacity which must be scheduled, assigned, or controlled by the operating system to assure consistent and non-conflicting usage by programs under execution. Examples of resources include: CPU time, memory space (actuals and virtual), and shared facilities (variables devices, spoolers, etc.).
RESUME	to resume the execution of a suspended process.
SUSPEND	to stop the execution of a process such that it can resumed. In the context of an Ada program being executed, this implies the suspension of all tasks, and the prevention of the activation of any task until the process is resumed. It specifically does not imply the release of any resources which a process has assigned to it, or which it has acquired, to support its execution.
TASK WAIT	delay of the execution of a task within a process until a CAIS service requested by this task has been performed. Other tasks in the same process are not delayed.
TERMINATE	to stop the execution of a process such that it cannot be resumed.
TRANSACTION	a grouping of operations, including a designated sequence of operations, which requires that either all of the designated operations are applied or none are; e.g., a transaction is uninterruptible from the user's point of view.
TYPE-AHEAD	the ability of a producer to transmit data units before the consumer requests the data units.
TYPING	an organization of entities, relationships and attributes in which they are partitioned into sets, called entity types, relationship types and attribute types, according to designated type definitions.
UNINTERPRETED DATA	a data representation whose structure is not controlled by CAIS facilities and whose structure is not used in the CAIS operations.

RAC Comment Form

!section:

!RAC version: **4 October 1986**

!submitter:

!date:

!1-line topic/subject:

!extended comment or recommendation:

!rationale for recommendation:

!disposition by RACWG:

[Send on MILNET to *POberndorf@Ada20.ISI.EDU* & *HMumm@Ada20.ISI.EDU*
or via U.S. Mail to "Patricia Oberndorf/Hans Mumm,
Code 423, NSSC, San Diego, CA 92152"]

**END
DATE
FILMED**

8-12-87