Ada Semantic Interface Specification (ASIS)

Currie Colket

Chair, ACM/SIGAda/ASIS Working Group (ASISWG)
Chair, ISO/IEC JTC1/SC22 WG9 ASIS Rapporteur Group (ASISRG)

Space and Naval Warfare Systems Command
SPAWAR 332; 2451 Crystal Drive
Arlington, Virginia 22245-5200
Phone: (703) 602-1483; FAX: (703) 602-6805

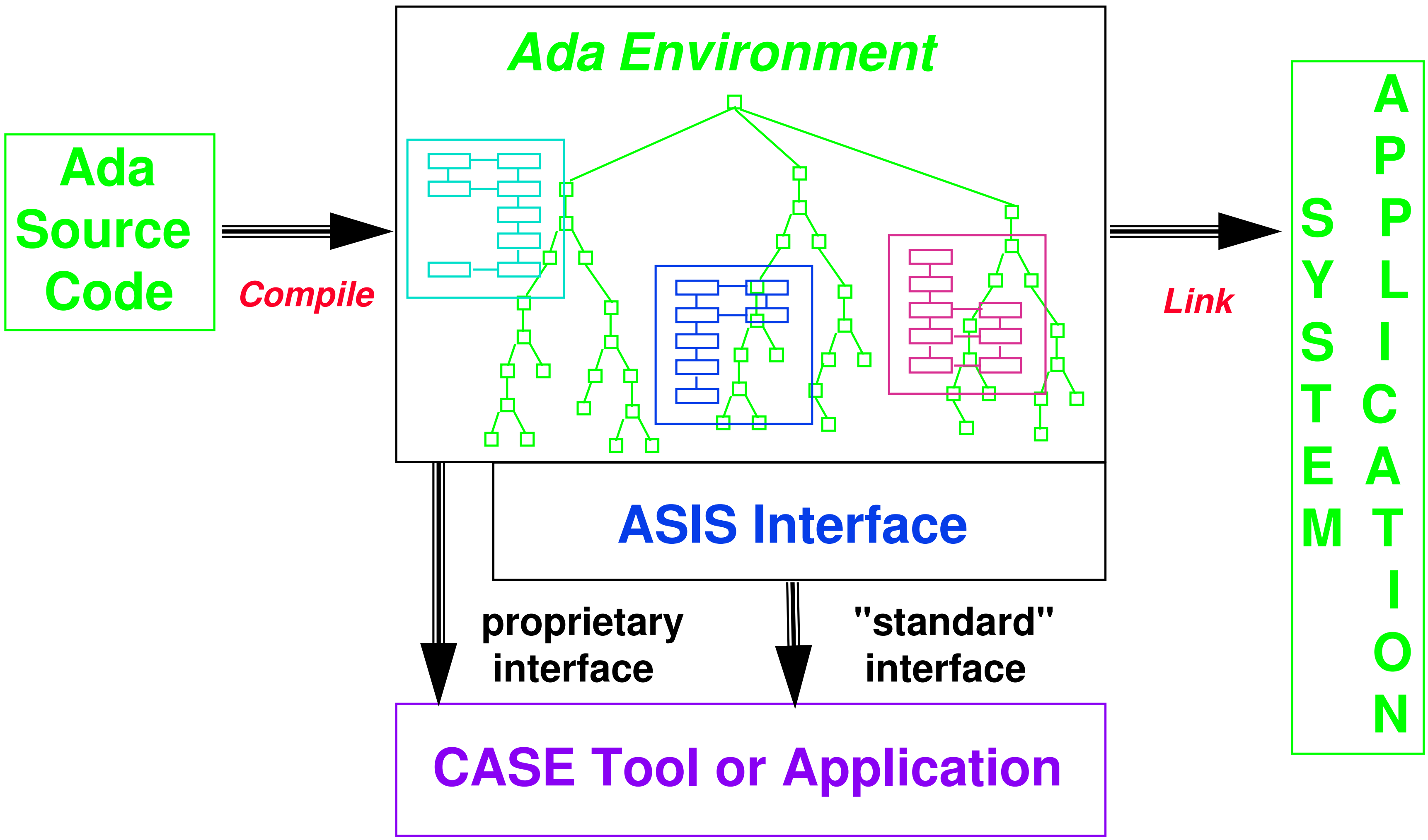
colket@smtp-gw.spawar.navy.mil
colket@sw-eng.falls-church.va.us

6 November 1995

Overview

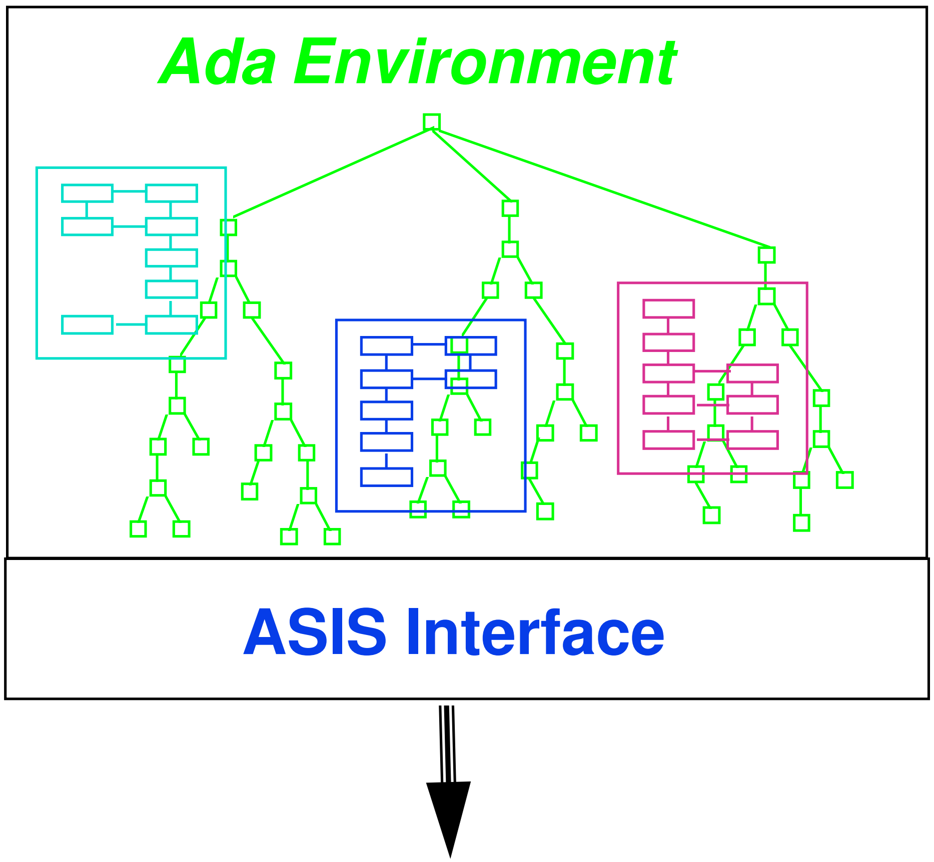
- What is ASIS? - Interface to the Ada Environment
  - Examples of Tools Built on ASIS
  - Examples of Applications Built on ASIS
- ASIS Functional Overview
- Secondary Abstractions
- Decoupling System Interfaces with ASIS
- ACM SIGAda's ASISWG
- ISO/IEC JTC1/SC22 WG9's ASISRG
- How to get ASIS
- ASIS GNAT Interface Implementation
- Expected ASISWG/ASISRG Schedule
- Final Notes

What is ASIS?



- Provides Syntactic and Semantic Information from
- Ada Environment using a "standard" interface

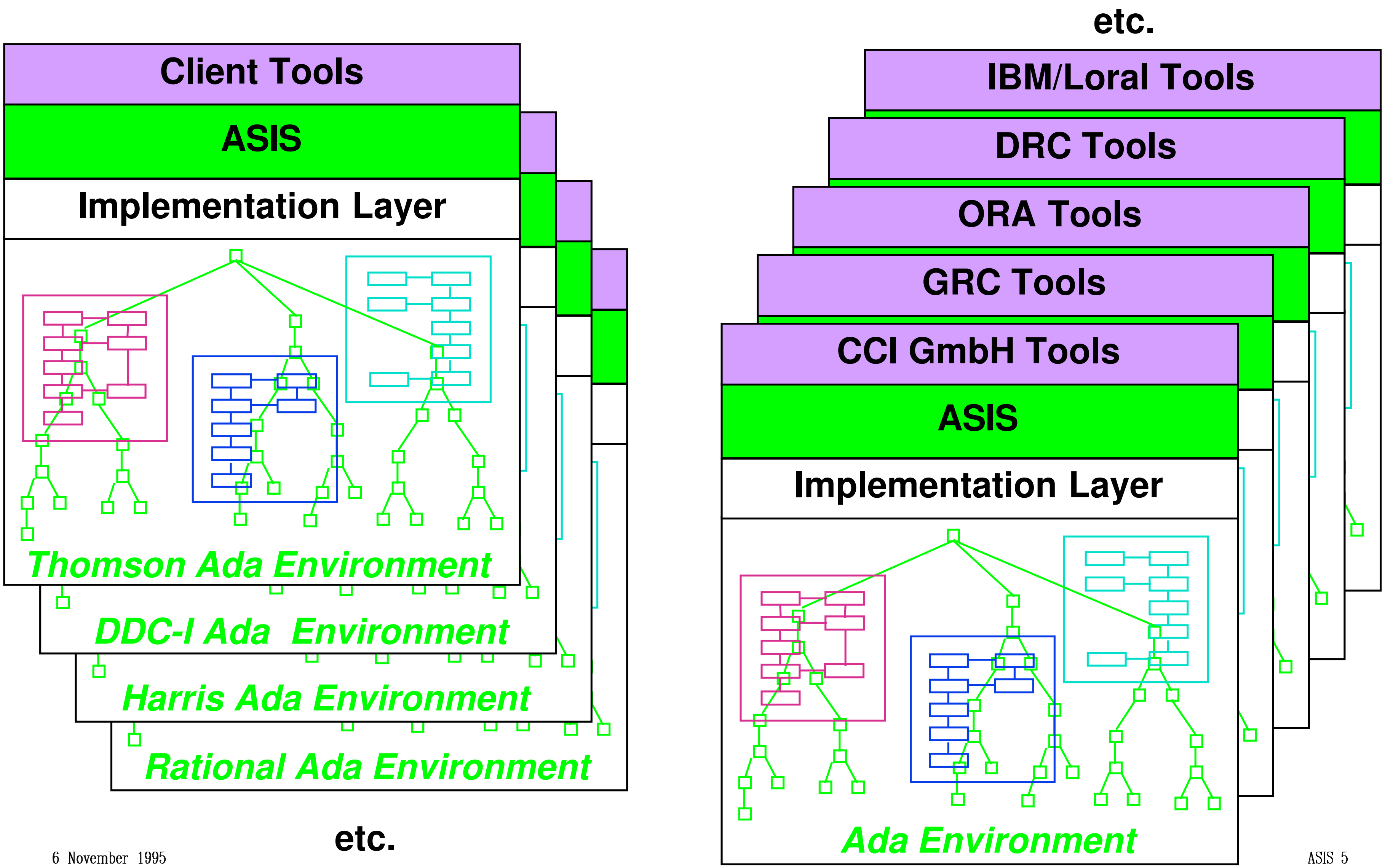
Examples of Tools Built on ASIS



Tools portable to Ada environments supporting ASIS Interface

- Code Restructuring Tools
- Code Browsing and Navigation Tools
- Coding Style and Standards Compliance Tools
- Cross Reference Tools
- Data Flow Analysis Tools
- Dependency Tree Analysis Tools
- Design Tools
- Document Generation Tools
- Invocation (Call) Tree Analysis Tools
- Language-sensitive Editing and Prettyprinting Tools
- Language Translation Tools
- Quality Assessment Tools
- Reverse Engineering Tools
- Re-Engineering Tools
- Safety & Security Compliance Tools
- Static Correctness Verifiers
- Tasking Analysis Tools
- Test-case Generation & Coverage Analysis Tools
- Usage, Quality, & Complexity Metrics Tools

ASIS Lets Client Tools "Snap On" to Compilation Systems



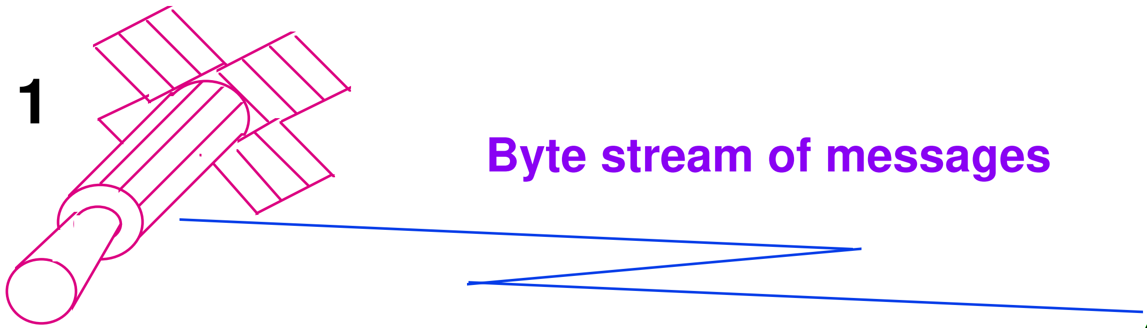
ASIS Usage

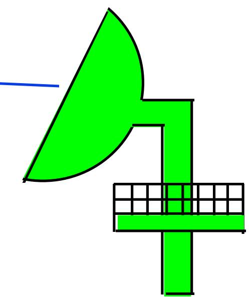
Compiler Vendors:	Tool Vendors:	End Users:	Countries:
DDC-I *	CCI GmbH	Air Force	Australia
Harris *	DRC	Boeing	Canada
Rational	Genoa Software *	IBM/Loral	China
Tomson	GRC	Logicon	France*
	Little Tree	Magnavox	Germany*
+	Consulting	MITRE	Japan*
	OC Systems	Navy	Russia
FSF GNAT *	ORA Corp	Unisys *	Switzerland
	SoHar		United Kingdom*
	WPL Labs *		United States
	Xinotech		

* Using in near future

Example of Application Built on ASIS-1

System 1

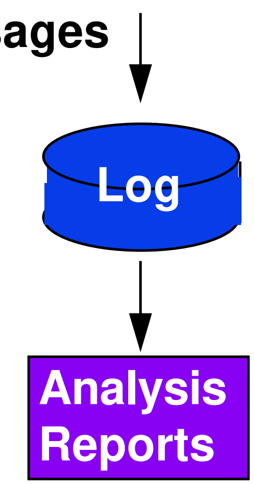




Problem

How do you delog data so an analyst can view data in a meaningful way?

Log Messages



Alternatives

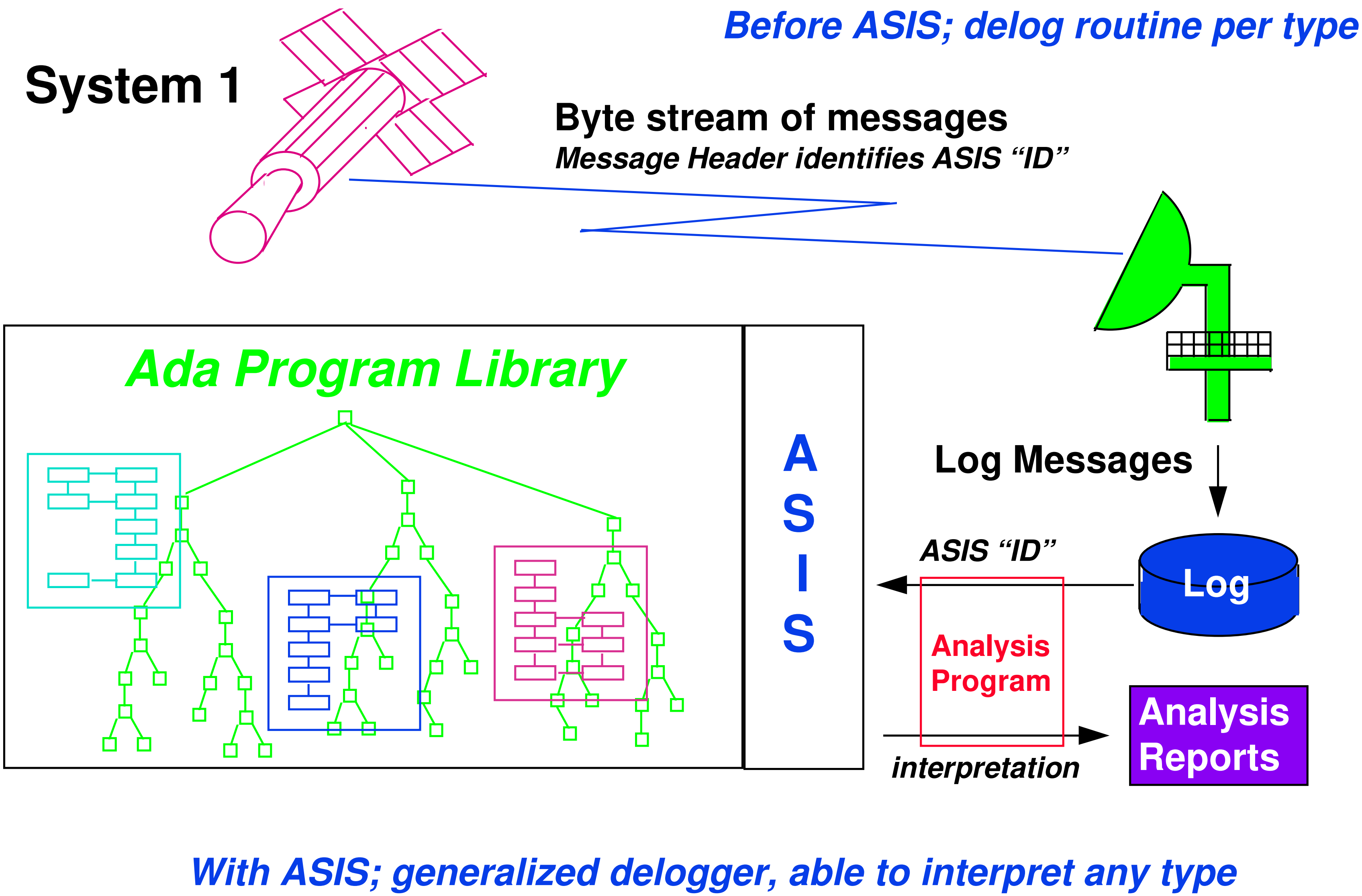
1. **Delog routine per message ID.**

   Expensive, inflexible, must be constantly updated

2. **Generalized delogger, able to interpret any message ID.**

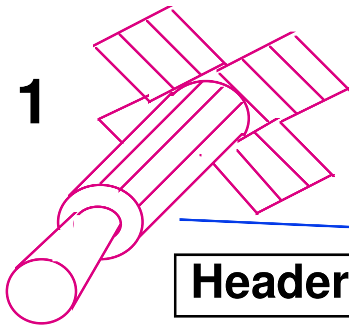
   Effective, highly flexible, never needs updating

Example of Application Built on ASIS-2



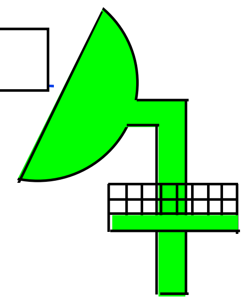
Example of Application Built on ASIS-3

System 1



Byte stream of messages
Message Header identifies ASIS “ID”

Header	**Byte_Stream (Array of Bytes)**

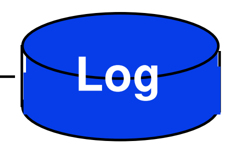


Log Messages

```
    type message_1 is
        record
            field_1 : array_type;
            field_2 : record_type;
            field_3 : private_enumeration_type;
        end record;
    type message_2 (discrim_2: integer) is private;

private
    subtype  range_type is positive range 10 .. 20;
    type varying_array_type is (integer range <> of float);
    type message_2  (discrim_2: integer);
        record
            field_1 : range_type;
            field_2 : varying_array_type (1 .. discrim_2);
        end record;
```

A S I S

ASIS “ID”



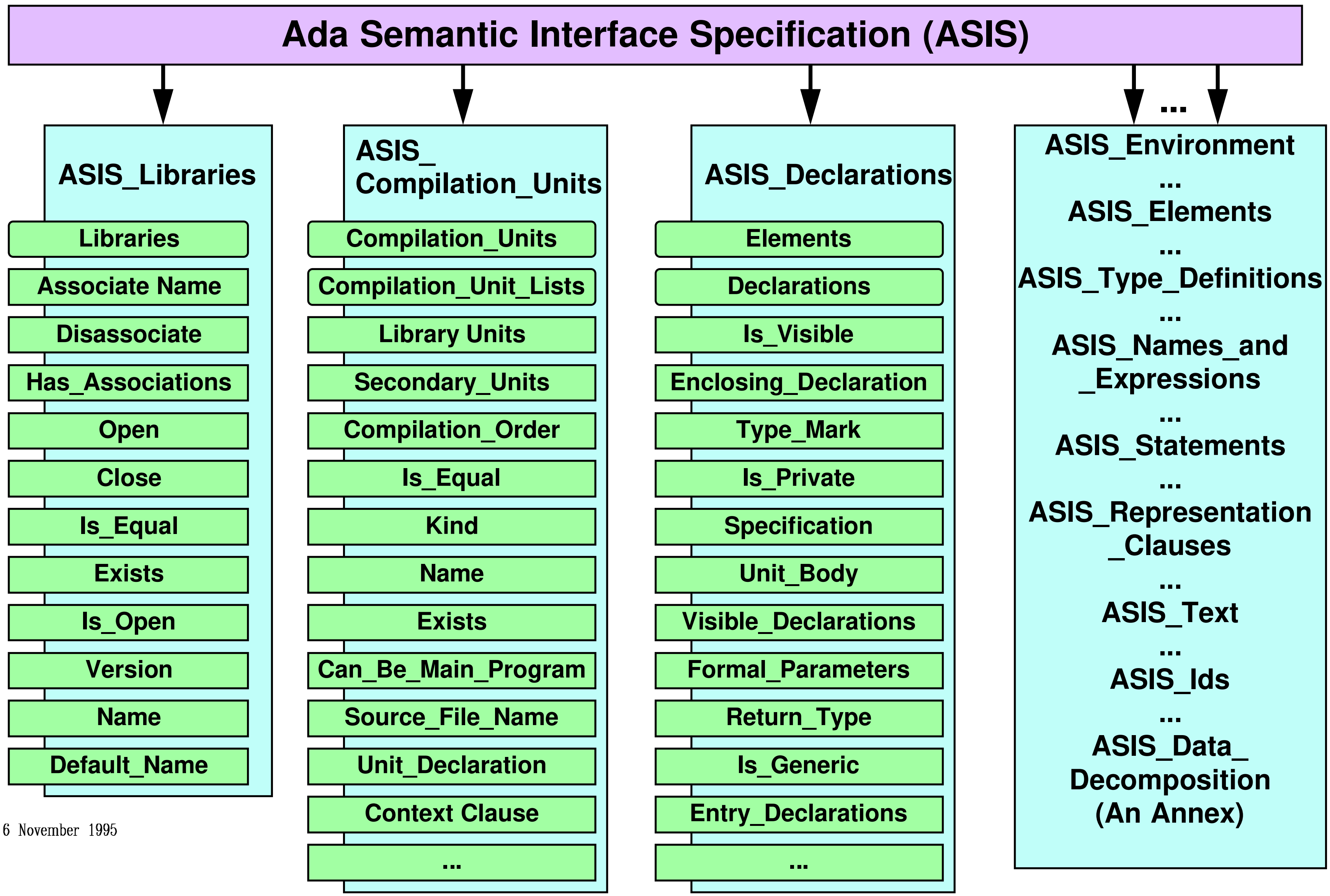
Analysis Program

Analysis Reports

interpretation services

ASIS provides ‘size, ‘position, ‘first_bit, ‘last_bit
Services index Byte_Stream to primitive types using Unchecked_Conversion
Services return values for analysis as variant record discriminated by ASIS Type_Kind

ASIS Functional Overview



ASIS Abstractions - Package ASIS

Ada Semantic Interface Specification (ASIS)

Provides common types for 12 ASIS packages

Encapsulates vendor dependencies in:

- **ASIS_Vendor_Primitives**

  Vendor dependent names and types are mapped to ASIS declarations

- **ASIS_Types_Primitives**

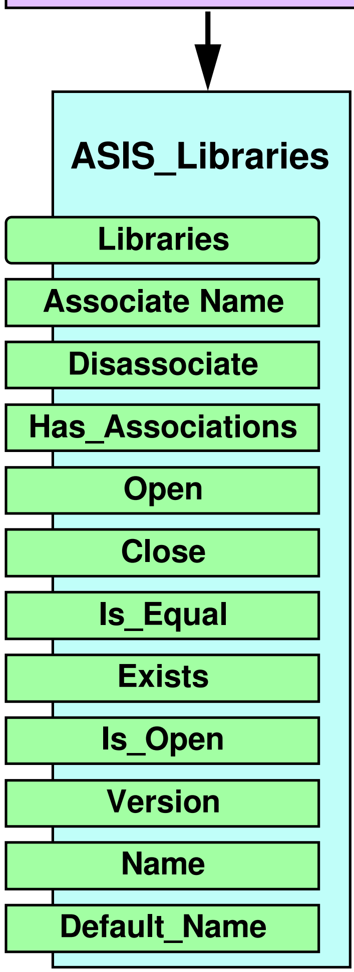
  Visibility is controlled by subtype declarations and renaming declarations exposing only vendor independent operations

Error conditions result in Ada Exceptions

- **Status & Diagnostic information available**

ASIS Abstractions - The Library

Ada Semantic Interface Specification (ASIS)



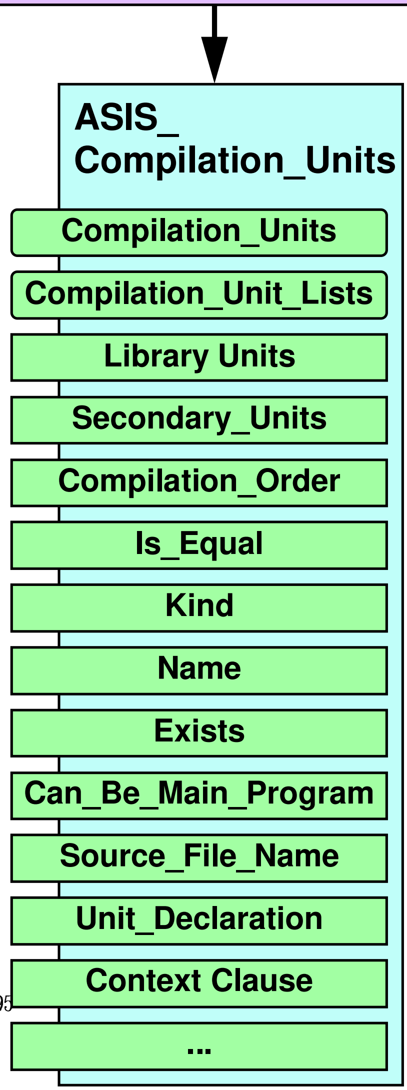
Library Model:

- **Associates name and parameters**
- **Open, Query, and Close**
- **Support for single or multiple libraries**
- **Free of vendor details and assumptions**

Ada Library is Read Only

ASIS Abstractions - Compilation Units

Ada Semantic Interface Specification (ASIS)



Provides external "Black Box" view

- **Attributes:**

  Date compiled, Text File, Compilation Options, Optimizations, Can Be a Main Program

- **Relationships:**

  Supporters, Dependents, Closure, Implicit Dependencies

Compilation Unit Model

- **Fetch by name**
- **Query of attributes & relationships**
- **Gateway to internal view using Elements**

ASIS Abstractions - Elements

Ada Semantic Interface Specification (ASIS)

...

ASIS_Environment
...
ASIS_Elements
...
ASIS_Type_Definitions
...
ASIS_Names_and _Expressions
...
ASIS_Statements
...
ASIS_Representation _Clauses
...
ASIS_Text
...
ASIS_Ids
...
ASIS_Data_ Decomposition (An Annex)

Provides internal “White Box” view

- **Logical handle to Ada elements:**

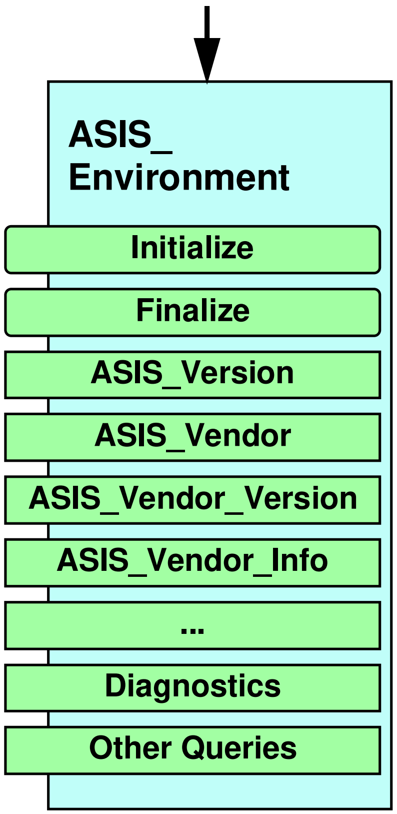
  Declarations, Statements, Expressions, Type Definitions, With Clauses ...

Element Model

- **Element kinds**
- **Specific element kinds**
- **Syntactical queries to classify, decompose, and recompose**
- **Semantic queries both forward & backward**
- **Elements know their enclosing Compilation Unit & Library**

ASIS Abstractions - Environment

Ada Semantic Interface Specification (ASIS)



Using ASIS:

```
ASIS_Environment.Initialize;          -- To setup environment
ASIS_Libraries.Associate (...);       -- To name Ada library
ASIS_Libraries.Open (...);            -- To gain library access
...
 -- Use various ASIS interfaces
...
ASIS_Libraries.Close (...);           -- To close access
ASIS_Dissociate (...);                -- To release library
ASIS_Environment.Finalize;            -- To release all resources
```

Simple ASIS Example

Ada Semantic Interface Specification (ASIS)

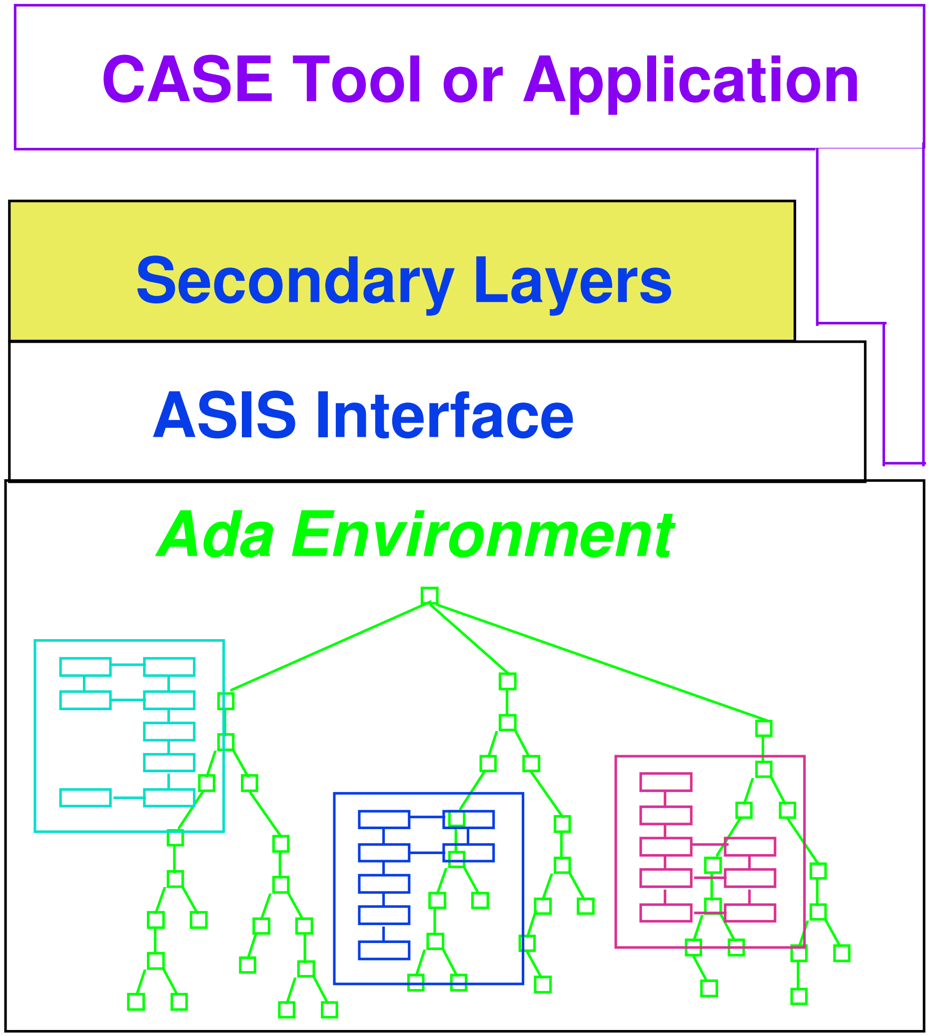
```
with text_io;
with ASIS;
with ASIS_Environment;
procedure ASIS_Environment_Example is

begin
    ASIS_Environment.Initialize;
    Text_io.Put_Line (“Is_Initialized: “ &
        Boolean’Image(ASIS_Environment.Is_Initialized);
    Text_io.Put_Line (“ASIS_Vendor: “ &
        ASIS.Strings.To_Standard_String(
            ASIS_Environment.ASIS_Vendor));
    Text_io.Put_Line (“ASIS_Vendor_Version: “ &
        ASIS.Strings.To_Standard_String(
            ASIS_Environment.ASIS_Vendor_Version));
    Text_io.Put_Line (“ASIS_Version: “ &
        ASIS.Strings.To_Standard_String(
            ASIS_Environment.ASIS_Version));
    ASIS_Environment.Finalize;
end ASIS_Environment_Example;
```

Output:

```
Is_Initialized: True
ASIS_Vendor: Alsys Inc.
ASIS_Vendor_Version:
  RISCAda SPARC Release 3
ASIS_Version: 1.1.1
```

ASIS Secondary Layers

Formerly called Program View Layers



Widget -like layer of abstractions that model commonly used views of Ada program structure

Each view provided in set of packages with operations for constructing and querying views

Offer CASE tool builders same kind of benefits enjoyed by X-users:

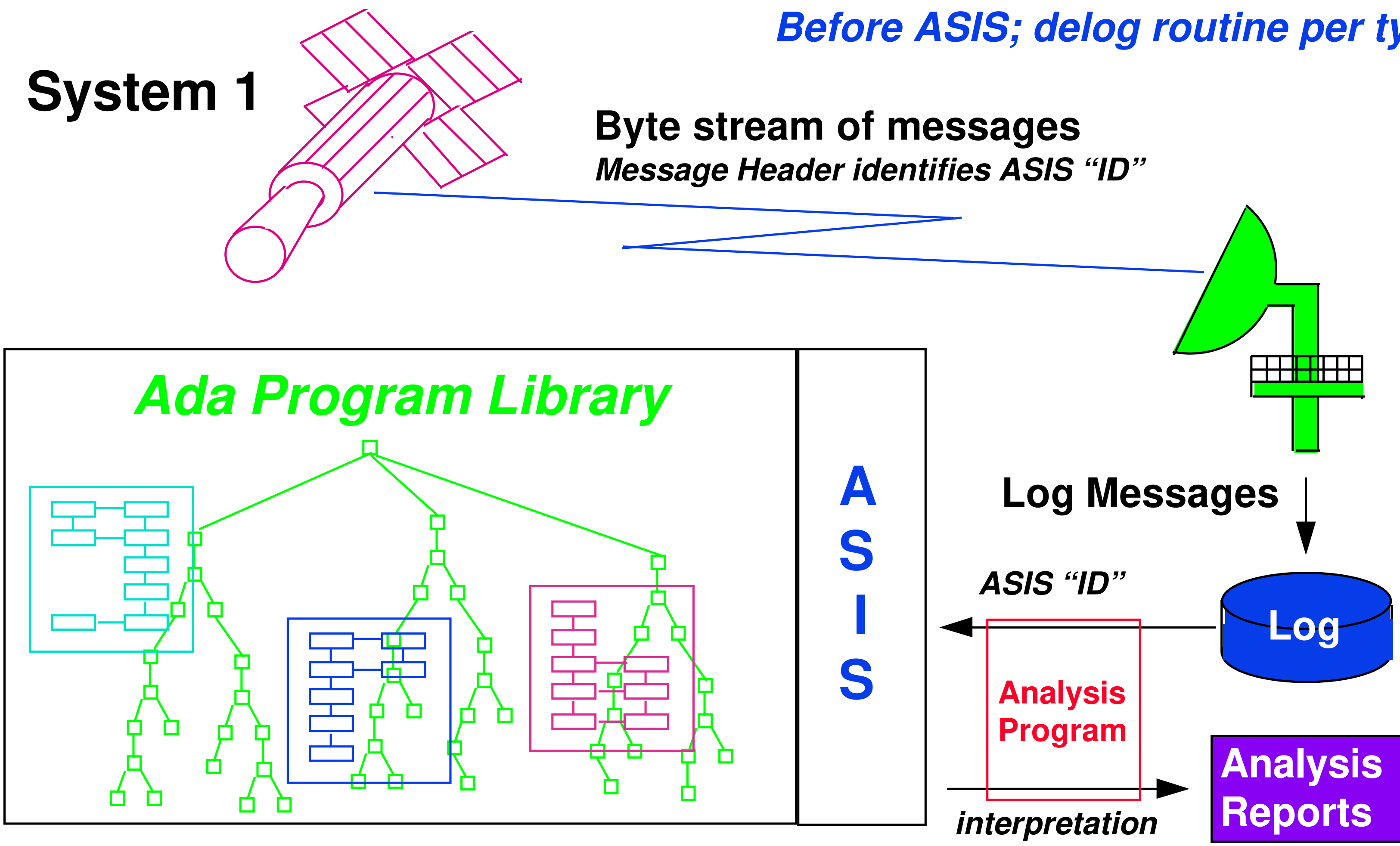
Productivity
Consistent results
Reliability
Portability

Secondary layers available for:

Scanners
Namespace View
Reference View
Region View
Control Flow View

Decoupling System Interfaces-1

Before ASIS; delog routine per type

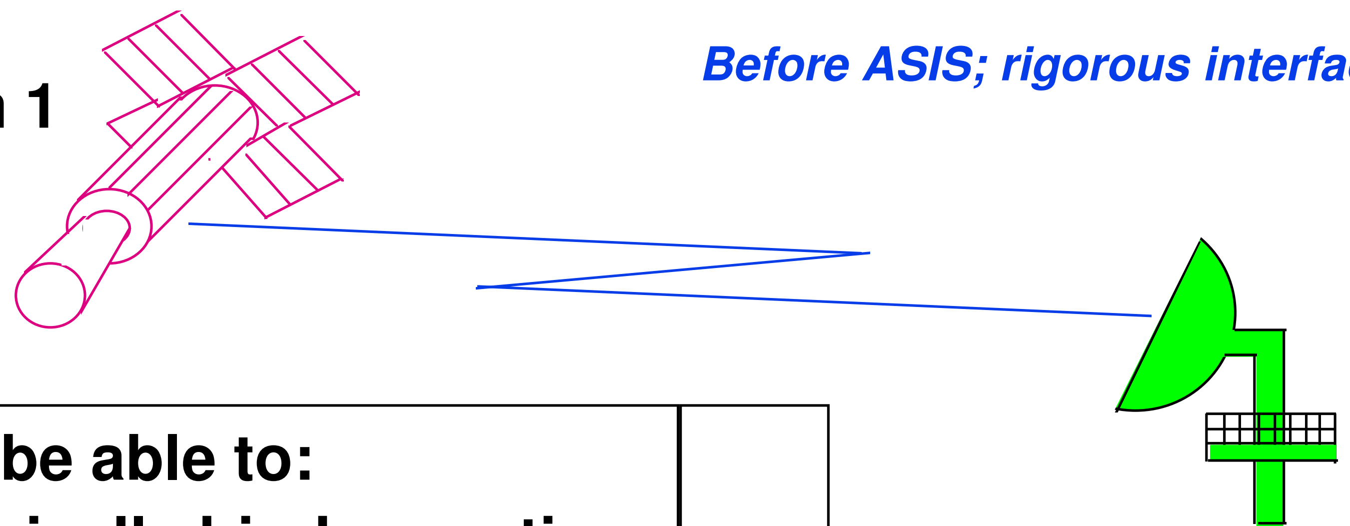


With ASIS; generalized delogger, able to interpret any type

Decoupling System Interfaces-2

System 1

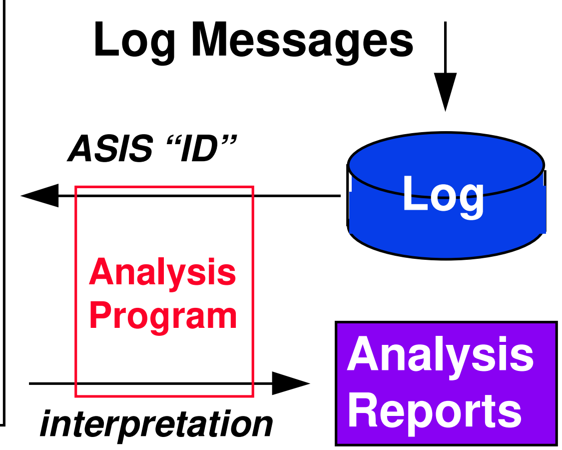
Before ASIS; rigorous interface



Like to be able to:

- Dynamically bind operations to message objects
- → Support true real-time interfaces
- Upgrade 2 systems asynchronously

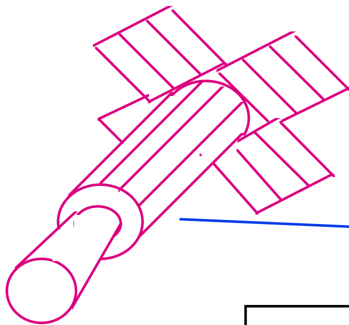
A
S
I
S



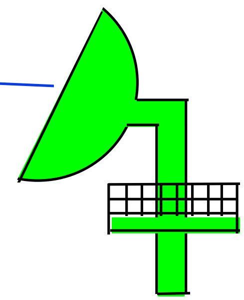
With ASIS; generalized delogger, able to interpret any type

Decoupling System Interfaces-3

Approach to dynamically bind operations to message objects



1. Initialize interface by sending each message
2. Transmit messages



Shared Object Type Declarations

A S I S

ASIS "ID"

1

ASIS provides 'size, 'position, 'first_bit, 'last_bit
Services index Byte_Stream to of shared object type into ground station object
Mapping operations could be dynamically bound during initialization to provide fast processing of subsequent messages

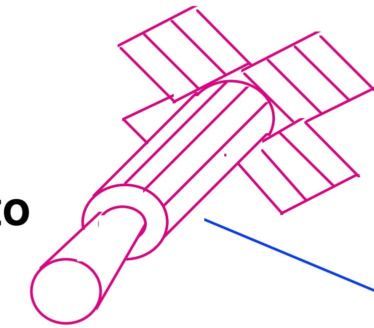
2

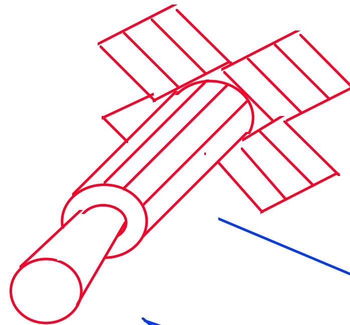
Real Time Processing of Messages

Dynamically bound operations to message objects could support true real-time interfaces

Vendor independent ASIS interface provides some capability to support heterogeneous interfaces

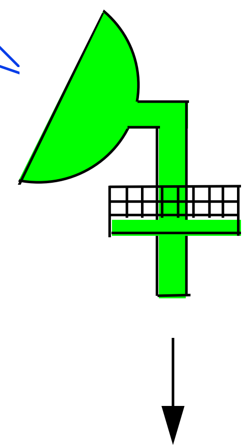
Decoupling System Interfaces-4





Ada95 allows OOP
- *Tagged Types (Class)*
- *Inherited Objects*
- *Extended Operations*

Approach to upgrade 2 systems asynchronously



Original Class Object:
- **Latitude**
- **Longitude**
- **Time of Observation**
- **Emitter Frequency**
- **Emitter PRI**

Extended Object:
- **Covariance Matrix**
- **Emitter Classification**

Operation
Fuse object
into system track

Perhaps ASIS could provide remote tag types for mapping into local tag types, if so

Systems can be upgraded asynchronously

ASIS Ada Board Resolution

Motion at the Ada Board 10-11 September 1992:

The Ada Board recognizes the potential benefits to the DoD Ada community of an ASIS standard and therefore it recommends that the AJPO director support by whatever means possible, the development of an ASIS standard and its submission to ISO/WG9 for publication.

Vote: Passed unanimously

ACM WG established to develop ASIS83 and ASIS95 with ISO WG9 Rapporteur Group for standardization of ASIS95

ACM's SIGAda's ASIS Working Group Charter

The purpose of this working group is to develop and participate in standardizing an implementation independent application programming interface to allow retrieval of the information stored within an Ada compiler's compilation database.

The initial goal of the working group was to further develop and promote the Ada Semantic Interface Specification (ASIS) for Ada 83. ASIS Version 1 is now complete and in the public domain. The working group is currently developing ASIS for Ada 9X, and will work towards its adoption as an international standard.

Membership in the working group is open to any interested party.

Current members include representatives of Ada compiler vendors, CASE tool developers, application developers, and other users.

Members are responsible for their own expenses.

ASIS defines a vendor independent interface for accessing both the syntactic and semantic information stored within an Ada compiler's library for an Ada compilation unit, including the information relating independently compiled units. The definition and operational semantics of ASIS are specified using Ada packages, with supporting commentary to describe the semantics and usage of operations. The Ada package specifications that compose ASIS are in the public domain, and each ASIS vendor is expected to provide a compliant implementation.

Original February 8, 1993, Updated June 28, 1994

ASISRG-1

ISO/IEC JTC1/SC22 WG9 ***voted unanimously to create ASISRG on 28 April 1995***

Scope

Standardize the interface between an Ada 95 compilation environment and tools/applications requiring information from this environment.

Purpose and Justification

The Ada Semantic Interface Specification (ASIS) is an interface between an Ada environment as defined by ISO/IEC 8652 and any tool requiring information from this environment. An Ada environment includes valuable semantic and syntactic information. ASIS is an open and published callable interface which gives CASE tool and application developers access to this information. ASIS has been designed to be independent of underlying Ada environment implementations, thus supporting portability of software engineering tools while relieving tool developers from having to understand the complexities of an Ada environment's proprietary internal representation.

ASISRG-1

Purpose and Justification (Continued)

As an international standard, ASIS will benefit the Information Technology community by facilitating the development of powerful CASE tools portable amongst the various environments provided by Ada vendors. This portability can only be achieved through the standardization of ASIS at the international level. A standardized ASIS will promote the development of powerful tools for the software engineering environment by providing access to important semantic information otherwise available only through proprietary interfaces. Further, ASIS will benefit the Information Technology community as a valuable resource for application development (e.g., decoupling system to system interfaces). The international standardization of ASIS will facilitate the use of this important capability in the development of system software applications.

ISO/IEC JTC1/SC22 is expected to approve ASISRG New Work Item in September 1995

ASISWG/ASISRG Officers

Chair ASISWG/ASISRG:	Currie Colket (SPAWAR)
Vice-chair ASISWG:	Steve Blake (Thomson)
Recorder ASISWG:	Clyde Roby (IDA)
Vice-recorder ASISWG:	Dan Cooper (Boeing)
Publicity/Meetings ASISWG:	Bill Thomas (MITRE)
ASISRG Co-Project Editors:	Steve Blake (Thomson) Clyde Roby (IDA)

How to Get ASIS - ASIS Artifacts

sw-eng **host on Internet as** ***sw-eng.falls-church.va.us***

ASIS directories: *As of 6 April 1995*

public/AdaIC/work-grp/asiswg
- **README**
- **asiswg-Charter**
- **ASIS.FAQ**
- **meetings/**
- **asis**
  - **issues/**
  - **V1.1.0/**
  - **V1.1.1/**
    - **ASIS83 files**
    - **testing/**
  - **V2.0.0/**
    - **ASIS95 files**

Mirrored in other media
- **AdaIC BB (modem)**
- **CD ROM**
- **Other US Internet nodes**
- **European Internet nodes**

ASIS83 is complete and ready for use (postscript & ASCII)

ASIS95 is under construction

Call +1 (703) 681-2466 for help in accessing ajpo host

ASIS Conformance Suite

Needed as a way to enhance tool-builder confidence that a particular ASIS implementation actually follows the standard

Valuable to vendor to ensure compliance

Suite of compliance tests is maintained with V1.1.1

- Complete with test harness
- ~ 1200 tests
- ~ 100 Mbytes
- Complete coverage for ASIS83

  (except for module decomposition)
- Input from users for tests important to user community

ASIS GNAT Interface Implementation

GNAT is an Ada95 compiler developed by NYU and distributed by the Free Software Foundation

- ASIS interface to GNAT is being developed by:
  - Dr. Sergey Rybin (Moscow State University)

    in conjunction with
  - Professor Alfred Strohmeier (Swiss Federal Institute of Technology)
- First free ASIS implementation prototype available in November 1995
- Partial implementation
- Should contain most of ASIS 2.0.Cstructural queries
- Limited to processing only one Compilation Unit at a time

How to get ASIS - WWW, BB, Internet

- ACM World Wide Web

  URL => http://www.acm.org/sigada/WG/asiswg/asiswg.html

- Ada Information Clearinghouse Bulletin Board:

  By modem auto-baud to 14,400: +1 (703) 681-2845

  Call +1 (703) 681-2466 for AdaIC BB voice help

  Name => *guest*   Password: *Ada4sw-eng*

  Parity = Even, Data Bits = 7, Stop Bits = 1, Xon/Xoff flowcontrol

- Ada Information Clearinghouse Internet via *ftp*:

  UNIX> ftp sw-eng.falls-church.va.us - Use ftp to sw-eng host

  Name: anonymous - Use *anonymous* login

  Password: <user ID> - Use user ID as password

  ftp> cd public/AdaIC/work-grp/asiswg - Change to asiswg sub-directory

  ftp> ls - List files in sub-directory

  ftp> get README - Retrieve asis README file

Call +1 (703) 681-2466 for AdaIC Internet voice help

Expected ASIS Schedule

Dec 93	AJPO recommends ASIS V1.1.0 (ASIS83) be used as interface to Ada83 Program Library
Mar 94	Design Goals for ASIS95 identified
Jun 94	ASISWG finalizes ASIS83 as V1.1.1 with test suite
Jun 94	Evaluate design approaches for ASIS95
Nov 94	Finalize approach for ASIS95
Mar 95	Skeleton ASIS95
Apr 95	ASISRG created unanimously by ISO/IEC JTC1/SC22 WG9
Jun 95	Skeleton ASIS95; new library/environment model
Fall 95	Public Review of ASIS95 initiated
Fall 95	ASISWG vote to submit ASIS95 as ISO Committee Draft
Fall 95	ASIS available for GNAT Ada95 Compiler
Fall 96	Balloting complete for ASIS95 Proposed Draft IS
Jun 97	Balloting for ASIS95 Draft International Standard
97-98	ASIS95 approved ISO Standard

Note: ASIS95 is an ASIS interface to Ada95

ASIS Interface - Funding

FY88 & FY89 Funding provided by STARS

Later in FY89, STARS initiated a policy not to fund any standardization efforts and ASIS funding was halted along with all other standardization activities

There has been no STARS, no AJPO, and no Navy funding for ASIS Standardization since FY89

All post FY89 ASIS standardization funding has resulted from industry investment - user community working closely with the compiler vendor community

Final Notes on ASIS

ASIS - Interface for obtaining information from Ada environments

- Provides both syntactic and semantic information
- Supports building powerful and highly portable CASE tools
- Supports many requirements for building systems
- Used in approximately 10 countries
- ASIS83 mature; evolving ASIS95, useful today; standard ~ 97-98

For ASIS W W W URL => http://www.acm.org/sigada/WG/asiswg/asiswg.html

To get involved with ASISWG and/or ASISRG, send email to:

asis-request@sw-eng.falls-church.va.us

to join asis@sw-eng.falls-church.va.us

asis-technical-request@sw-eng.falls-church.va.us

to join asis-technical@ sw-eng.falls-church.va.us or call

Currie Colket @ +1 (703) 602-1483 or Clyde Roby @ +1 (703) 845-6666