Product Standard

Object Management:

Common Object Request Broker Architecture (CORBA)

The Open Group

Product Standard

Object Management: Common Object Request Broker Architecture (CORBA)

Document Number: X98OR

Published in the U.K. by The Open Group, October 1998.

Any comments relating to the material contained in this document may be submitted to:

The Open Group
Apex Plaza
Forbury Road
Reading
Berkshire RG1 1AX
U.K.

Or by email to:

OGSpecs@opengroup.org

Product Standard

NAME

Common Object Request Broker Architecture (CORBA)

LABEL FOR LOGO

CORBA®

DESCRIPTION

The Object Management Group has developed an architecture and specification for object technology use, management, interworking (the exchange of data between object models), and interoperability (the means of message exchange between object request brokers). Products which are developed in compliance with this architecture and specification include an Object Request Broker (ORB) which enables objects to transparently make and receive requests and responses in a distributed environment. In addition to the ORB, products may also supply object services that applications may share. These object services are a collection of services (interfaces and objects) that support basic functions for using and implementing objects, as well as common facilities.

Products that conform to this Product Standard include an implementation of the OMG abstract object model which includes the following:

- Object Request Broker (ORB), including interfaces that allow access to the ORB without dependencies on the Basic Object Adapter (BOA)
- Interface Definition Language (IDL), which is used to describe the interfaces that client objects call and object implementations provide
- Dynamic Invocation Interface (DII), which describes the client side of the interface that allows dynamic creation and invocation of requests to objects
- Dynamic Skeleton Interface (DSI), which describes the server-side interface that can deliver requests from an ORB to an object implementation that does not have compile-time knowledge of the type of object it is implementing
- Interface Repository (IR), which describes the component of the ORB that manages and provides access to a collection of object definitions

The detailed Technical Standard specifying the functionality in CORBA-compliant products, including specification of the CORBA architecture and components, is detailed in *The Common Object Request Broker: Architecture and Specification*[1] and the associated *IDL/Java Language*

1. The Common Object Request Broker: Architecture and Specification, OMG, Revision 2.0: July 1995, updated July 1996, Revision 2.1: August 1997, available via FTP at:

   ftp://ftp.omg.org/pub/docs/formal/97-09-01.pdf

Mapping.[2] Hereafter, these documents are referred to as ‘‘the Specification’’.

In addition to the core ORB technology, these products supply an implementation of the ORB interoperability architecture that supplies the framework for ORB interoperability, including the General Inter-ORB Protocol (GIOP) and the Internet Inter-ORB Protocol (IIOP). Products described by this Product Standard correspond to *Interoperability-compliant ORBs*, as defined by Section 9.3.4 of the Specification.

Each implementation also includes one or more of the following language mappings of OMG IDL:

- C Language
- C++ Language
- Java
- Smalltalk
- COBOL
- ADA

This Product Standard includes core ORB functionality, DII, DSI, IR, IIOP Version 1.0, and at least one of the OMG IDL language mappings.

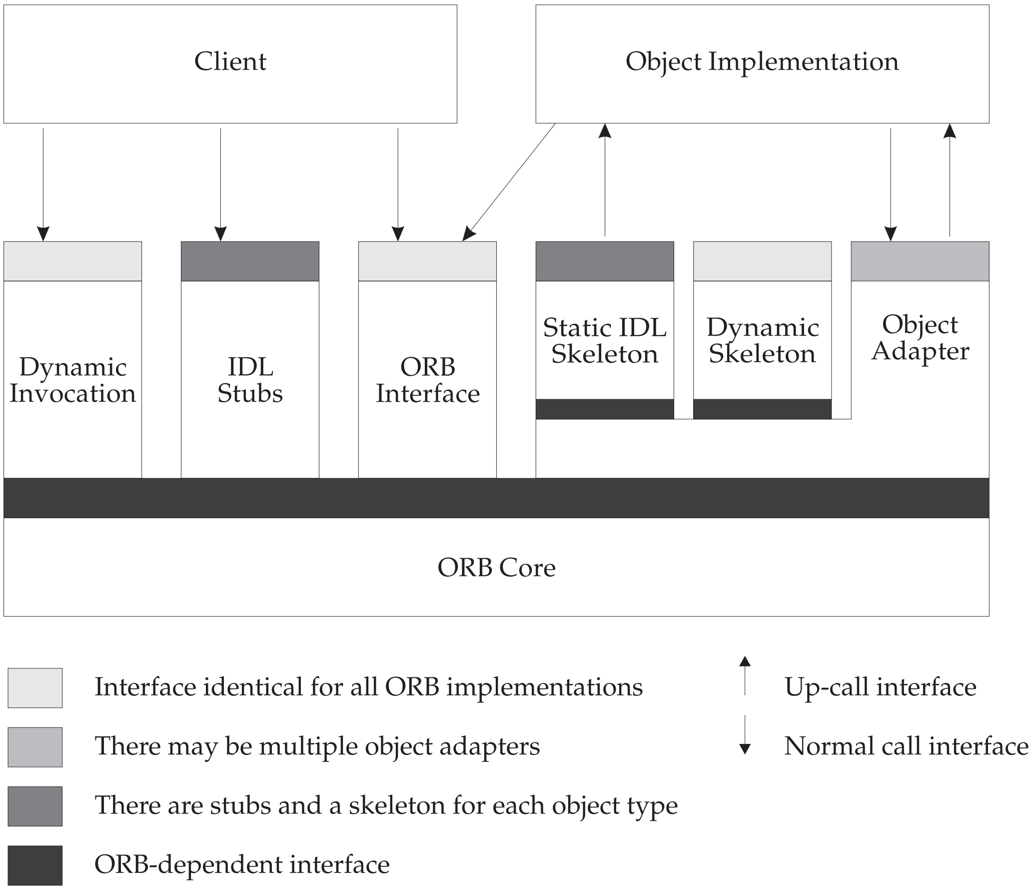
This Product Standard does not require inclusion of the following components of the CORBA specification:

- Basic Object Adapter (BOA)
- Portable Object Adapter (POA)
- COM/CORBA interworking functionality
- Any specific object facilities or services

Support for IIOP Version 1.0 is required. Support for IIOP Version 1.1 is optional.

2. IDL/Java Language Mapping, OMG, currently available as OMG TC Document orbos/98-01-16 via FTP at:
ftp://ftp.omg.org/pub/docs/orbos/98-01-16.pdf

The CORBA conceptual architecture is shown in the following figure:



CONFORMANCE REQUIREMENTS

A conformant product consists of a particular combination of software supported on certain specified hardware platforms, and the specific implementation of the product to be registered must therefore be uniquely identified in the Conformance Statement.

A single configuration of the product shall meet all mandatory conformance requirements. In addition, products may implement optional facilities. Conformant products may also provide additional standards-based or proprietary services, as long as the mandatory conformance requirements continue to be met, or the Conformance Statement contains instructions on how the user may configure the environment such that the product meets the conformance requirements.

A single configuration of the product shall meet all the conformance requirements defined in the indicated chapter of the Specification. It shall support all the definitions of those chapters including all functions and headers, all protocol messages, all externally stored or transmitted data formats, and all application/service capabilities, with the following exceptions:

1. The Basic Object Adapter (BOA) interface, and any other interfaces or operations on other interfaces that are dependent on the BOA.
2. The extended data types **wchar**, **wstring**, **fixed**, **long long**, and **long double**; TypeCode interfaces and operations required to support these extended data types; language

mapping support for these extended data types; and support for codeset negotiation.

Note: Interfaces that are defined in the Specification as implementation-defined do not meet the criteria for inclusion as conformance requirements. Implementation-defined behavior is detailed in the Conformance Statement.

Human-Computer Interface

Not applicable.

Portability Interface

Object Request Broker (ORB): Chapters 1, 2, and 5 of the Specification.

Note: *get_current*( ), *get_service_information*( ), *get_implementation*( ), *get_policy*( ), and *get_domain_managers*( ) are deprecated in the Specification and therefore not required.

Dynamic Invocation Interface (DII): Chapter 4 of the Specification.

Dynamic Skeleton Interface (DSI): Chapter 6 of the Specification.

Interface Repository (IR): Chapter 7 of the Specification.

Programming Language Environment

OMG Interface Definition Language (IDL): Chapter 3 of the Specification and one or more of the language mappings listed below.

Note: For clarification, Section 3.2 of the Specification details character set ISO/Latin 1 requirements; however, full ISO/Latin 1 support is not required.

C Language Mapping: Chapter 17 of the Specification.

C++ Language Mapping: Chapter 18 of the Specification.

Smalltalk Language Mapping: Chapter 19 of the Specification.

COBOL Language Mapping: Chapter 20 of the Specification.

ADA Language Mapping: Chapter 21 of the Specification.

Java Language Mapping: *IDL/Java Language Mapping*.

Interoperability

Membership in the *Interoperability Program*[3] is required.

GIOP and IIOP: Chapters 9, 10, 11, and 12 of the Specification.

Conformance with IIOP Version 1.0 is required. Support of IIOP Version 1.1 is optional.

Products registered as conformant to this Product Standard may optionally claim conformance to IIOP Version 1.1 (Chapter 12 of the Specification). To conform to IIOP Version 1.1, the product must support the IIOP Version 1.0 conformance requirements, as well as additionally supporting:

---

3. The Interoperability Program is defined in Open Group Brand Program Documentation, September 1998, Interoperability Program (ISBN: 1-85912-285-X, X983).

Fragment message (message format)
Non-byte-oriented data types

There are no additional specific interoperability assurance requirements for CORBA products.

- Data Interchange Formats

  None.

- Communications Interfaces and Protocols

  The IIOP services in a conformant product require reliable, end-to-end connections for their execution. Consequently, they are defined to be implemented on top of the Transmission Control Protocol (TCP). TCP defines a method for connection-oriented, end-to-end communications to take place over Internet Protocol-based networks.

  Products registered as conformant to this Product Standard, and the respective environments in which their conformance has been established, shall conform to the following:

  IETF RFC 793, Transmission Control Protocol
  IETF RFC 791, Internet Protocol
  IETF RFC 950, Internet Standard Subnetting Procedure
  IETF RFC 919, Broadcasting Internet Datagrams
  IETF RFC 922, Broadcasting Internet Datagrams in the Presence of Subnets
  IETF RFC 792, Internet Control Message Protocol
  IETF RFC 1112, Host Extensions for IP Multicasting

OPERATIONAL ENVIRONMENT

For a conformant product including the C Language mapping, a C compiler compliant with *ISO/IEC 9899: 1990* is required to be supported.

For a conformant product including the C++ Language mapping, a C++ compiler that supports all the features described in the *Annotated C++ Reference Manual (ARM)*[4] is required to be supported. Further, the C++ compiler must support templates and must supply native exception handling. Namespace and RTTI support are not required.

For a conformant product including the Java Language mapping, a Java compiler that supports all the features described in the *Java Platform Core API*[5] is required to be supported.

For a conformant product including the COBOL Language mapping, a COBOL compiler compliant with *ISO 1989: 1985* is required to be supported.

For a conformant product including the ADA Language mapping, an ADA compiler compliant with *ISO/IEC 8652: 1995* is required to be supported.

4. Annotated C++ Reference Manual (ARM), Margaret A. Ellis and Bjarne Stroustrup, published by Addison-Wesley, June 1, 1990 (ISBN: 0-201-51459-1).

5. The Java Application Programming Interface described in one of two volumes by J. Gosling, F. Yellin, and The Java Team, published by Addison-Wesley, 1996 (ISBN: 0-201-63453-8).

PORTABILITY ENVIRONMENT

There are no dependencies on Operating System interfaces.

OVERRIDING STANDARDS

The Object Management Group's *The Common Object Request Broker: Architecture and Specification* and the associated *IDL/Java Language Mapping* (see the DESCRIPTION).

INDICATORS OF COMPLIANCE

A Test Report from the currently authorized release of the VSOrb or the VSJOrb Test Suite. Refer to *http://www.opengroup.org/testing* to ascertain the currently authorized version of the test suites, and to *Testing and the Open Brand* for detailed testing requirements.

The initial release of VSOrb and VSJOrb contains test coverage in the following areas:

- Object Request Broker (ORB) APIs and Semantics
- Dynamic Invocation Interface (DII)
- Dynamic Skeleton Interface (DSI)
- Interface Repository (IR)
- Interoperability (GIOP and IIOP)
- Interface Definition Language (IDL)
- (VSOrb only) C Language Mapping
- (VSOrb only) C++ Language Mapping
- (VSJOrb only) Java Language Mapping

For products which offer a C Language or C++ Language binding, the IIOP tests must run successfully in a multi-system, multi-vendor conformant ORB product configuration.

For products which offer a Java Language binding, the IIOP tests must run successfully in a multi-system, multi-vendor conformant ORB product configuration.

MIGRATION

None.

INTEROPERABILITY PROGRAM

In maximizing the involvement of the Overriding Standards owner (the Object Management Group), the following alterations to the standard Interoperability Program processes take precedence in any invocation related to this Product Standard:

- Having decided to proceed with the invocation of the Conciliation process, The Open Group will, within five days of the decision to proceed, gather information from the Requester and the Vendors involved in the issue, and submit a description of the issue with an accompanying draft Interoperability Recommendation to the Object Management Group's Urgent Clarification Process.
- The Requester, if not a Vendor, is not required to participate in the creation of the draft Interoperability Recommendation.
- All discussions conducted and resolutions reached as part of the Conciliation or Mediation processes will be handled as an Urgent Clarification as described in the Object Management

Group Policies and Procedures document.[6]

- The Open Group will, where appropriate, represent the Requester's interests in the Object Management Group processes and procedures.
- The Object Management Group will return an approved Interpretation or Interoperability Recommendation to The Open Group within ten working days of receipt of the initial draft Interoperability Recommendation from The Open Group.
- Although it is recommended that all Vendors are part of the Object Management Group and their appropriate working groups and task forces, Vendors of Registered Products are not required to participate in the Object Management Group's processes and procedures.
- The Object Management Group may extend the period for reaching a resolution regarding the draft Interoperability Recommendation after consultation with the Requester and The Open Group.
- The Requester may request that The Open Group does not publish the approved Interoperability Recommendation provided by the Object Management Group.

6. *Policies and Procedures of the Object Management Group Technical Process* (pp/98-08-01, Version 1.7, August 4th 1998).