Product Standard

Object Management:

Common Object Request Broker Architecture (CORBA®) Version 2.3

The Open Group

Product Standard

Object Management:
Common Object Request Broker Architecture (CORBA®) Version 2.3

Document: X01OM

Published in the U.K. by The Open Group (June 2001)

Any comments relating to the material 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 shall include an implementation of the OMG abstract object model that contains the following:

- Object Request Broker (ORB), including interfaces that allow access to the ORB and interoperability between ORBs
- Interface Definition Language (IDL), which is used to describe the interfaces that client objects call and object implementations provide
- Portable Object Adapter (POA), which is used to support the management, configuration and invocation of objects
- 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

- Value Type Semantics, which describes the ability to pass object parameters to other objects by value
- Abstract Interface Semantics, which defers determination of whether an object is passed by reference or by value until runtime
- Dynamic Management of **any** Values, which provides a portable way of using **any** values by a receiving object
- Interface Repository (IR), which describes the component of the ORB that manages and provides access to a collection of object definitions

The Technical Standards specifying the functionality in CORBA 2.3-compliant products, including specification of the CORBA 2.3 architecture and components, are found in *The Common Object Request Broker: Architecture and Specification, Revision 2.3.1* and the associated *Language Mapping Specifications*. Hereinafter, these documents are referred to as "the Specification".

The Common Object Request Broker Architecture and Specification, Revision 2.3.1, October 1999, is available at: http://www.omg.org/ as document:

CORBA http://www.omg.org/cgi-bin/doc?formal/99-10-07

In addition to the core ORB technology, conforming products provide 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 12.3.4 of the Specification.

Each implementation must include one or both of the following language mappings of OMG IDL, which are aligned with CORBA 2.3. The reference documents may be found at the OMG document server URL noted above.

- C++ http://www.omg.org/cgi-bin/doc?formal/99-07-41
- Java http://www.omg.org/cgi-bin/doc?formal/99-07-53

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

- DCE ESIOP (Chapter 16)
- COM/CORBA interworking functionality (Chapters 17, 18, 19 and 20)
- Interceptors (Chapter 21)
- Any specific object facilities or services

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.

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 vendor provides 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 as noted below. 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 unless otherwise noted below.

Note:

> Interfaces that are defined in the Specification as implementation-defined do not meet the criteria for inclusion as conformance requirements.

Human-Computer Interface

Not Applicable.

Portability Interface

1. Object Request Broker (ORB): Chapters 1, 2, and 4 of the Specification.

   Note:

   Implementation of Management of Policy Domains (Section 4.10) is not required.

2. Value Type Semantics: Chapter 5 of the Specification.
3. Abstract Interface Semantics: Chapter 6 of the Specification.
4. Dynamic Invocation Interface (DII): Chapter 7 of the Specification.
5. Dynamic Skeleton Interface (DSI): Chapter 8 of the Specification.
6. Dynamic Any: Chapter 9 of the Specification.

   Note:

The comment in Specification section 4.2 ("Dynamic Any related operations deprecated and removed from primary list of ORB operations") refers to a previous specification of Dynamic Any and not to the material in Chapter 9.

7. Interface Repository (IR): Chapter 10 of the Specification.

Note:

create_recursive_sequence_tc is deprecated in the Specification and therefore not required. The generation of RMI Hashed Format (Section 10.6.2) is excluded as it does not apply to all bindings.

8. Portable Object Adapter (POA): Chapter 11 of the Specification.

Programming Language Environment

OMG Interface Definition Language (IDL): Chapter 3 of the Specification, and one or both of the C++ and Java language mappings contained in the associated Language Mapping documents of the Specification, with the following exception:

1. The **fixed** and **long double** extended data types; TypeCode interfaces and operations required to support these extended data types; and language mapping support for these extended data types are not required. However, IDL compilers must parse these extended data type keywords and associated phrases properly, and may post "unimplemented" warning messages.

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.

Interoperability

GIOP and IIOP: Chapters 12, 13, 14 and 15 of the Specification.

Note:

IIOP 1.2 support is required. The requesting_principal field in the GIOP request header is deprecated for GIOP version 1.0 and 1.1; this field is not present in GIOP version 1.2. Although OMG has registered ports 683 (CORBA-IIOP) and 684 (CORBA-IIOP-SSL) with IANA, there is no requirement at present to monitor these specific ports.

• 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 adhering 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 which includes the C++ Language mapping, a C++ compiler that supports all the features described ISO/IEC 14882: 1998 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 which includes the Java Language mapping, a Java compiler that supports all the features described in the *Java Platform Core API*[1], JDK Level 1.1 or higher, is required to be supported.

PORTABILITY ENVIRONMENT

There are no dependencies on Operating System Interfaces.

[1] The Java Application Programming Interface (Java Series), J. Gosling, F. Yellin, and The Java Team, published by Addison-Wesley, 1996 (ISBN: 0-201-63453-8).

OVERRIDING STANDARDS

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

INDICATORS OF COMPLIANCE

A Test Report from the currently authorized release of the VSOrb (for C++ Implementations) or the VSJOrb (for Java Implementations) 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 release of VSOrb and VSJOrb contains test coverage in the following areas:

- Object Request Broker (ORB) APIs and Semantics
- Value Type Semantics
- Abstract Interface Semantics
- Dynamic Invocation Interface (DII)
- Dynamic Skeleton Interface (DSI)
- Dynamic **any**
- Interface Repository (IR)
- Portable Object Adapter (POA)
- Interoperability (GIOP and IIOP)
- Interface Definition Language (IDL)
- (VSOrb only) C++ Language Mapping
- (VSJOrb only) Java Language Mapping

For products which offer a 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.

----- ooooo -----