



**Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom**

TELEPHONE:
INTERNATIONAL:
FAX:
E-MAIL:

**Cambridge (01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk**

Training

ANSAwise - Introduction to CORBA and OMG

Chris Mayers

Abstract

Needing to select a environment within which to procure and build open distributed systems, organizations find it difficult to compare their features and benefits. One important standard is CORBA, specified by the OMG.

This module of the ANSAwise training programme describes the OMG's Common Object Request Broker Architecture (CORBA) both in the way it is standardized, and also in its overall structure and content. Since interoperability is a major concern, this topic is covered in detail.

[This module is a variant of APM.1322, but this module covers only CORBA, not DCE.]

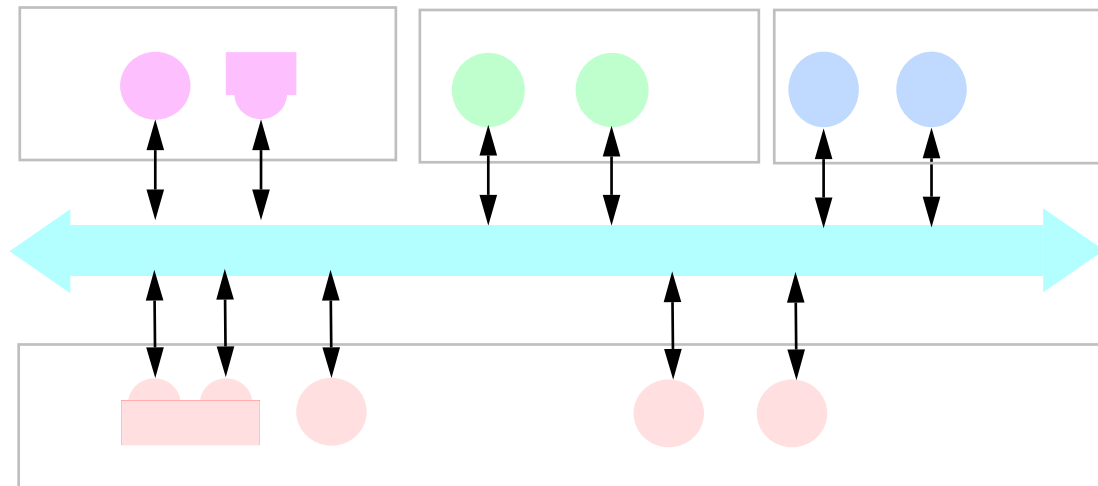
APM.1536.04

Approved
Briefing Note

13th May 1996

Distribution:
Supersedes:
Superseded by:

Introduction to CORBA and OMG



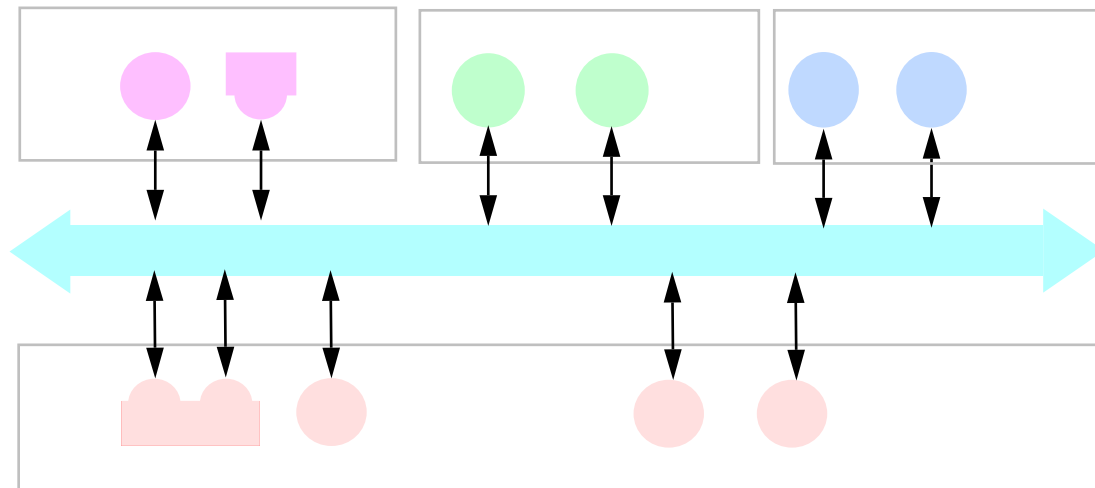


In this session

- Describe the key features of a major open environment for distributed computing
 - CORBA (Common Object Request Broker Architecture)
- Indicate its place in your distributed systems strategy

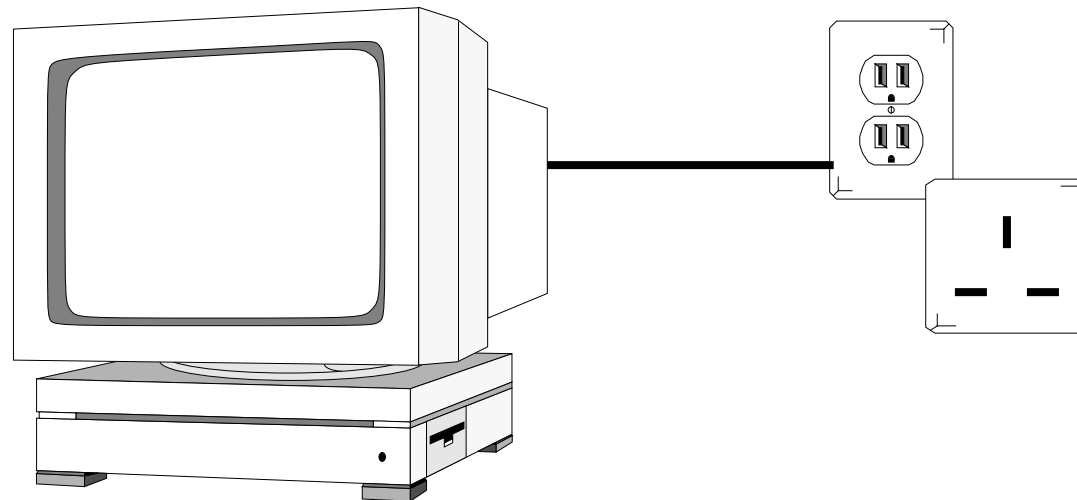
CORBA (Common Object Request Broker Architecture)

- Architecture standardized by the Object Management Group (OMG)



The OMG's Vision

- **The Global Information Appliance**



- **“It should be as easy to plug a computer into a world of computing services as it is to plug a computer into the world power grid”**



The OMG's Focus

- **Application integration**
 - the same thing as distributed processing
- **Constructing information-sharing distributed systems from diverse sources**
 - heterogeneous
 - networked
 - physically disparate
 - multi-vendor



The OMG's view of previous approaches

- They were too low-level
 - excellent building blocks, but not at the level the application developer is interested in
- There was no standardized integrating framework for applications



The OMG's Approach

- **Create consensus based on commercially-available software...**
 - proposals for standards must describe technology that is *imminently available*
 - paper-only standards are inadequate
- **... Create a marketplace for off-the-shelf standards-compliant software**



The OMG's Method of Operation

- **Not-for-profit company**
- **Small staff; no internal development**
- **Object World subsidiary spreads the word through shows, conferences, market studies, seminars,...**
- **Now has more than 600 members**



How OMG Standards are Made

- **OMG selects interfaces**
 - through competitive selection from industrial proposals
- **OMG publishes interfaces**
 - the specifications are freely available to anyone who wants them
- **OMG controls interfaces**
 - they belong to OMG, not the submitter; OMG controls their evolution
- **OMG liaises with standards bodies**
 - so that specifications become standards

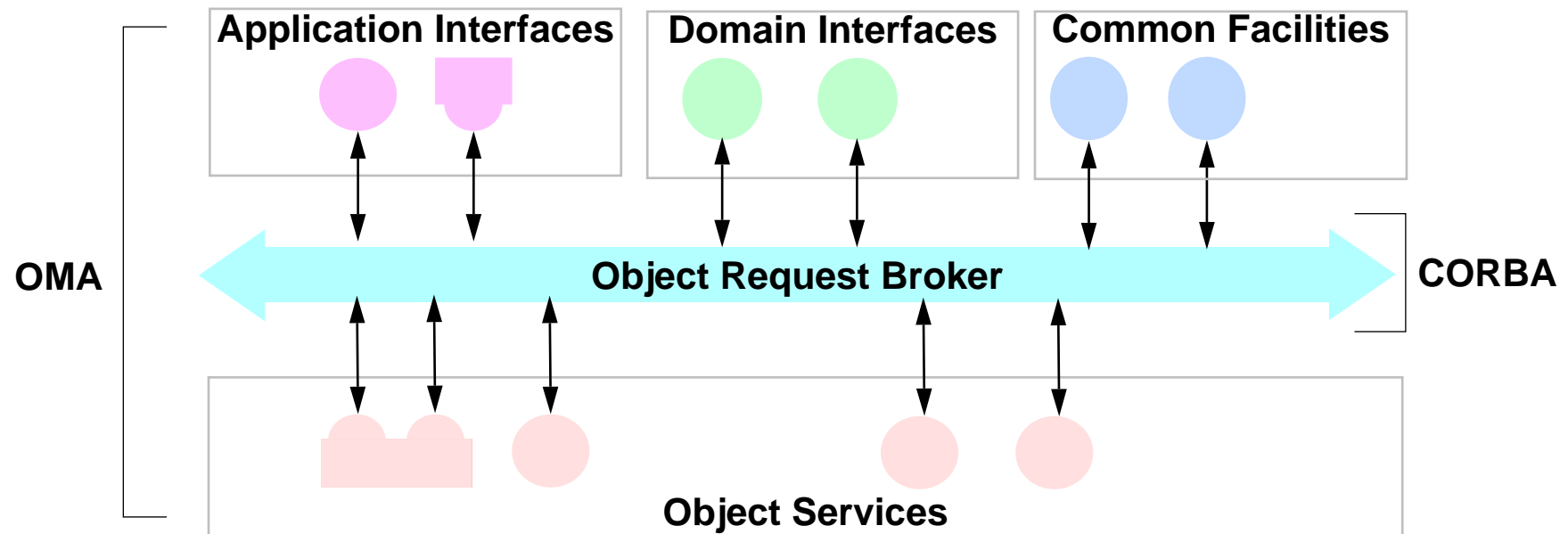


OMG is Neutral

- **OMG itself does not deal with implementations**
 - **it does not create, sell, or resell implementations**

- **OMG does not test implementations**
 - **this is done by X/Open, who have a strong reputation in conformance testing**

The Object Management Architecture



- **Consists of the Object Request Broker (ORB), plus objects**
 - **Objects are Object Services, Common Facilities, Domain Interfaces, or Application Interfaces**

The Object Request Broker (ORB)

- **Objects request the services of other objects via the ORB**
 - the ORB is responsible for locating the object implementation, and all the communications mechanisms that support the request
 - client and object implementation can be written in different languages, and run on different types of machines





Object Services (CORBAservices)

- (Common) Object Services are basic 'system-level' services
 - but they are themselves *objects*
 - they have interface specifications, just like all objects...
 - so alternative implementations are possible, with different quality-of-service, say
- Example Object Services
 - Transactions
 - Concurrency
 - Events



Common Facilities (CORBAfacilities)

- **Common Facilities are application-level objects that can be shared between applications**
- **For example:**
 - **e-mail**
 - **printing**
 - **compound documents**
- **Common Facilities support interoperation of third-party products**



Domain Interfaces

- **Domain interfaces are specific to vertical markets**
- **For example:**
 - **geo-spatial data processing**
 - **system management**



Application Interfaces

- **Application Interfaces are application-specific**
 - provided by third-party ISV (Independent Software Vendor)
 - provided by end-user
- **Application Interfaces are therefore not specified by the OMG**



Platform Technology Committee

- **Is the plenary for platform-related technology adoption**
 - **Object Services**
 - **Object Request Broker**
 - **Common Facilities**
 - **Analysis and Design**
- **Is umbrella for related Special Interest Groups**
 - **Internet**
 - **Realtime**
 - **Electronic Commerce**



Domain Technology Committee

- **Is the plenary for domain-specific technology adoption, via Task Forces**
 - **CORBAfinancials: Financial domain**
 - **CORBAmed: Healthcare domain**
 - **CORBAmanufacturing: Manufacturing domain**
 - **CORBAtel: Telecommunications domain**
 - **Business Objects**
- **Is umbrella for related Special Interest Groups**
 - **Transportation**
 - **Interactive Multimedia and Commerce**



Architecture Board

- **Maintains the overall quality, consistency, and coherence of OMG specifications**
 - **maintains the OMA**
 - **makes sure the adopted specifications are consistent with the OMA and each other**
 - **does no technology adoption itself, but vets the requests from, and responses to, the other Committees**



Portability in CORBA - Platforms

- **CORBA is portable**
 - implementations now exist for most platforms and operating systems...
 - ...mainframe, minicomputers, micros,...
- **For practical portability, you must take into account**
 - operating system version
 - compiler version and vendor
 - processor type



Portability in CORBA - Programming Languages

- CORBA can be used with a range of programming languages
 - for each programming language, a *mapping* must be standardized
- Currently C, C++, Smalltalk, and Ada are standardized (COBOL and others to follow)



CORBA - Programming Interfaces

- **CORBA is object-oriented**
- **CORBA requires a commitment to object-oriented principles...**
 - **...but you can use a non-object-oriented programming language (for example C)**



Portability, Diversity, and Interoperability

- Remember that you are not constrained to use the same platform and programming language everywhere
 - a client application can be written in Smalltalk on a PC...
 - ... the server can be written in COBOL on a mainframe
- You can change platform and programming language later
- This relies on interoperability between distributed systems implementations from different vendors



Interoperability - a key issue

- **How can distributed systems interoperate? Two approaches**
 - **a common protocol**
 - **protocol gateways**



Interoperability in CORBA

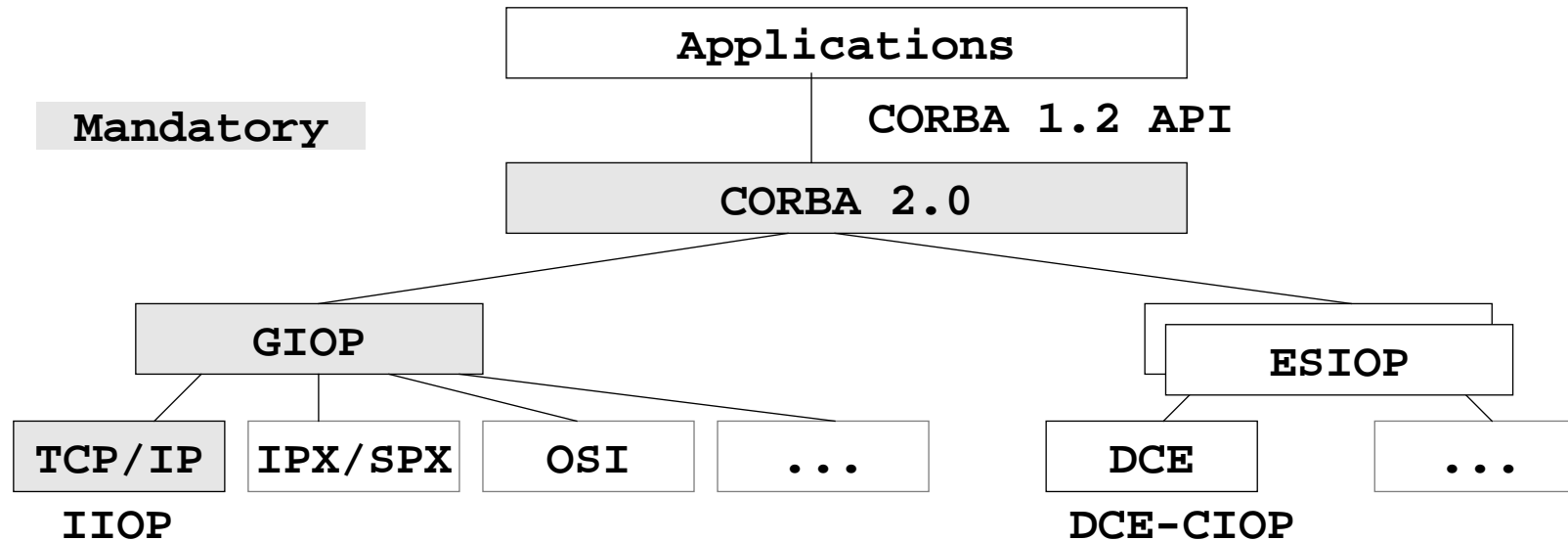
- **OMG regarded interoperability as essential**
 - but this was not handled in CORBA 1.x
- **In CORBA 2.0, OMG did not select a common protocol because**
 - no single protocol could give optimal performance in all situations
 - there were existing implementations to consider



Interoperability in CORBA 2.0

- **CORBA 2.0 interoperability comprises**
 - **an overall architecture for CORBA-CORBA communications**
 - **an API for building bridges**
 - **a general multi-transport message format (GIOP: General Inter-ORB Protocol)**
 - **an API for gateways using other message formats (ESIOP: Environment-Specific Inter-ORB Protocols)**

CORBA Interoperability Architecture



- **The IIOP (Internet Inter-ORB Protocol) is mandatory**
 - guarantees interoperability between any ORBs
- **The DCE-CIOP (Common Inter-ORB Protocol) is optional**
 - if provided, guarantees interoperability with likewise ORBs



GIOP and IIOP Goals

- **Widest possible availability**
- **Simplicity**
- **Scalability**
- **Low cost**
- **Generality**
- **Architectural neutrality**



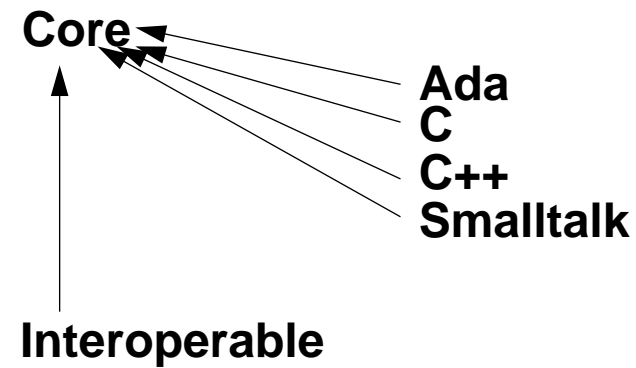
What is CORBA 2?

- **CORBA 2 is a family of component specifications**
 - vendors can support some components, but not others
- **The component specifications are the compliance components**
 - each component will have an associated branding stamp
 - if a component is supported, it must be supported exactly as specified



The CORBA 2 Component Family

- Extra CORBA2/... components will continue to be added





CORBA Implementations - 1

- **Digital (DEC): ObjectBroker**
- **Chorus: COOL/ORB**
- **Expersoft: PowerBroker**
- **Hewlett-Packard: Distributed Smalltalk**
- **IBM: SOMobjects**
- **ICL: DAIS**
- **IONA: Orbix**



CORBA Implementations - 2

- **NeXT: PDO/OpenStep**
- **Object-Oriented Technologies: DOME**
- **PostModern Computing: ORBeline**
- **SunSoft: NEO**



CORBA Implementation Trends

- **Some ORB implementations will be bundled with complete programming environments and tools**

- **Some ORB implementations may be bundled with the operating system in future**



Summary

- Take care to assess your CORBA needs, and compare like with like
- For more on CORBA
 - see *CORBA* (Object Management Group Inc.)...
 - ... see also *First Class* magazine, published by OMG