



Sistemi Distribuiti

Corso di Laurea in Ingegneria


Prof. Paolo Nesi
Parte: 3 – CORBA

Department of Systems and Informatics
University of Florence
Via S. Marta 3, 50139, Firenze, Italy
tel: +39-055-4796523, fax: +39-055-4796363

Lab: DISIT, Sistemi Distribuiti e Tecnologie Internet
nesi@dsi.unifi.it, paolo.nesi@unifi.it
www: <http://www.dsi.unifi.it/~nesi>





Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 1



CORBA



- CORBA Architecture
- General Concepts
- ORB Structure
- Client and Server in CORBA
- Object Adapter
- CORBA for WEB applications
- Usage of CORBA
- Single and Multithread CORBA



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 2

CORBA
Common Object Request Broker Architecture

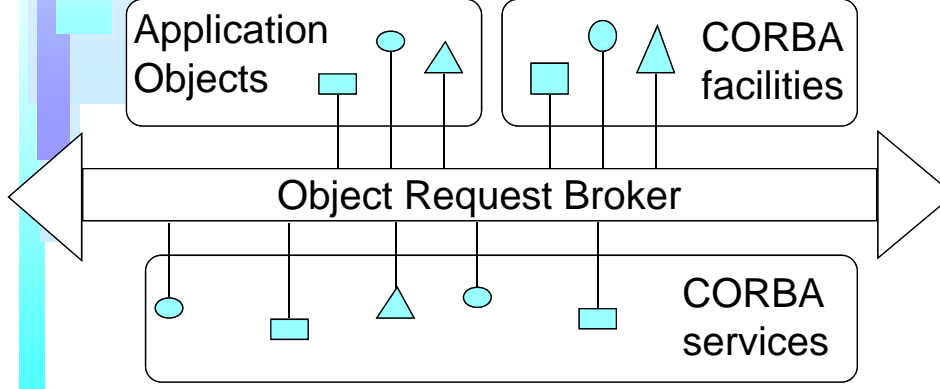
- **OMG's** (Object Management Group) specification for interoperability between distributed computing nodes (1989)
- **ORB**: middleware that establishes requester-provider relationship
- **Goal:**
 - ♣ Usage of OO programming in Distributed Systems
 - ♣ Allow heterogeneous environments communicating at object level
 - ♣ regardless of implementation of the endpoints
 - Different languages in the applications
 - Different implementations of the ORB
- CORBA 1 (1990), CORBA 2 (1996)



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 3

CORBA
CORBA, Common Object Request Broker Architecture

- Defined by the Object Management Group nel 1991
- Object Management Architecture





The diagram illustrates the CORBA architecture. At the top, there are two boxes: 'Application Objects' on the left and 'CORBA facilities' on the right. Each box contains three icons: a square, a circle, and a triangle. Below these boxes is a horizontal bar labeled 'Object Request Broker'. The 'Object Request Broker' bar has two large arrows pointing outwards to the left and right. Below the 'Object Request Broker' bar is another box labeled 'CORBA services'. This box contains five icons: a circle, a square, a triangle, a circle, and a square. Lines connect the icons in the 'Application Objects' and 'CORBA facilities' boxes to the 'Object Request Broker' bar, and lines connect the icons in the 'CORBA services' box to the 'Object Request Broker' bar.

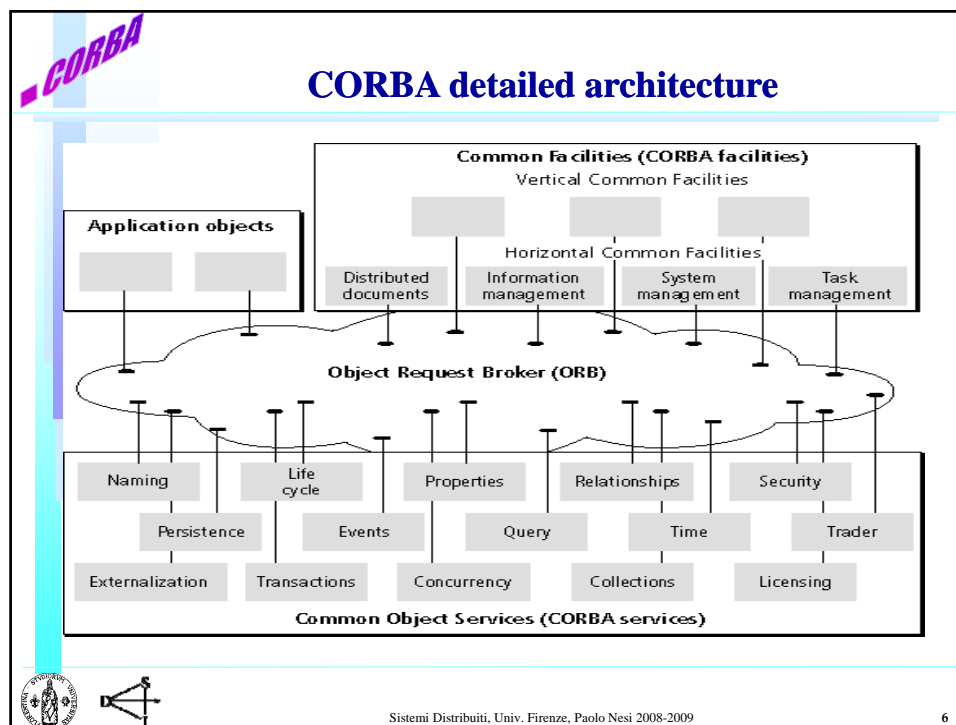
Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 4

CORBA

- **Object Request Broker (ORB)**
 - ♣ The libraries, processes, and other infrastructure in a distributed environment that enable CORBA objects to communicate with each other.
 - ♣ The ORB connects objects requesting services to the objects providing them.
- **Naming service**
 - ♣ to allow CORBA objects to be named by binding a name to an object reference.
 - ♣ The name binding may be stored in the naming service, and
 - a client may supply the name to obtain the desired object reference.

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 5



CORBA

4 Componenti di CORBA

- **ORB, Object Request Broker**
 - ♣ Distributed application
 - ♣ rende trasparente la locazione fisica degli oggetti, naming
 - ♣ unmarshal-marshall, e invocazione dei metodi
- **CORBA Services**
 - ♣ Security, time, etc..
 - ♣ persitency, events, transactions, etc..
- **CORBA Facilities**
 - ♣ Servizi di base condivisi da molte applicazioni
 - ♣ Non vitali come i CORBA Services, OS esteso....
 - ♣ E.g.: amministrazione sistema, mail, etc.
- **Application Objects**
 - ♣ Objects basati su CORBA

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 7

CORBA

The main components of the CORBA architecture

- The CORBA architecture is designed to allow clients to invoke methods in CORBA objects
 - ♣ clients and objects can be implemented in a variety of programming languages
 - ♣ it has additional components wrt a generic MiddleWare
 - object adapter, implementation repository and interface repository

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 8

CORBA

The main components of the CORBA architecture

- ORB core
- role similar to that of a communication module
- provides an interface that includes operations to:
 - ✦ enable it to be started and stopped;
 - ✦ convert between remote object references and strings, naming;
 - ✦ provide argument lists for requests using dynamic invocation.

client: client program, proxy for A, ORB core
 implementation repository, interface repository
 Request
 Reply
 server: ORB core, object skeleton adapter, Servant A
 or dynamic invocation, or dynamic skeleton

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 9

CORBA

The main components of the CORBA architecture

- Client stubs/proxies
- Written in the client language.
- The IDL compiler for the client language uses an IDL interface to generate one of the following:
 - ✦ for object-oriented languages the class of a proxy
 - ✦ for procedural languages a set of stub procedures.
- the client stubs/proxies marshal the arguments in invocation requests and unmarshal exceptions and results in replies

client: client program, proxy for A, ORB core
 implementation repository, interface repository
 Request
 Reply
 server: ORB core, object skeleton adapter, Servant A
 or dynamic invocation, or dynamic skeleton

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 10

CORBA

The main components of the CORBA architecture

- Dynamic invocation interface
- In some applications (e.g. browsers), a client without the appropriate proxy class may need to invoke a method in a remote object.
- CORBA does not allow classes for proxies to be downloaded at run time as in Java RMI.
- The dynamic invocation interface is CORBA's alternative. (see the Interface Repository)

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009

11

CORBA

The main components of the CORBA architecture

- **Skeletons**
 - skeleton classes (for OO languages) are generated in the language of the server by the IDL compiler.
 - remote method invocations are dispatched via the appropriate skeleton to a particular servant,
 - the skeleton unmarshals the arguments in request messages and marshals exceptions and results in reply messages.

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009

12

CORBA

The main components of the CORBA architecture

- **Object adapter**
 - bridges the gap between
 - ♣ CORBA objects with IDL interfaces and
 - ♣ the programming language interfaces of the corresponding servant classes.
 - it does the work of the remote reference and dispatcher modules

client: client program, proxy for A, ORB core
 implementation repository, interface repository
 server: ORB core, object adapter, skeleton, Servant A
 Request, Reply
 or dynamic invocation, or dynamic skeleton

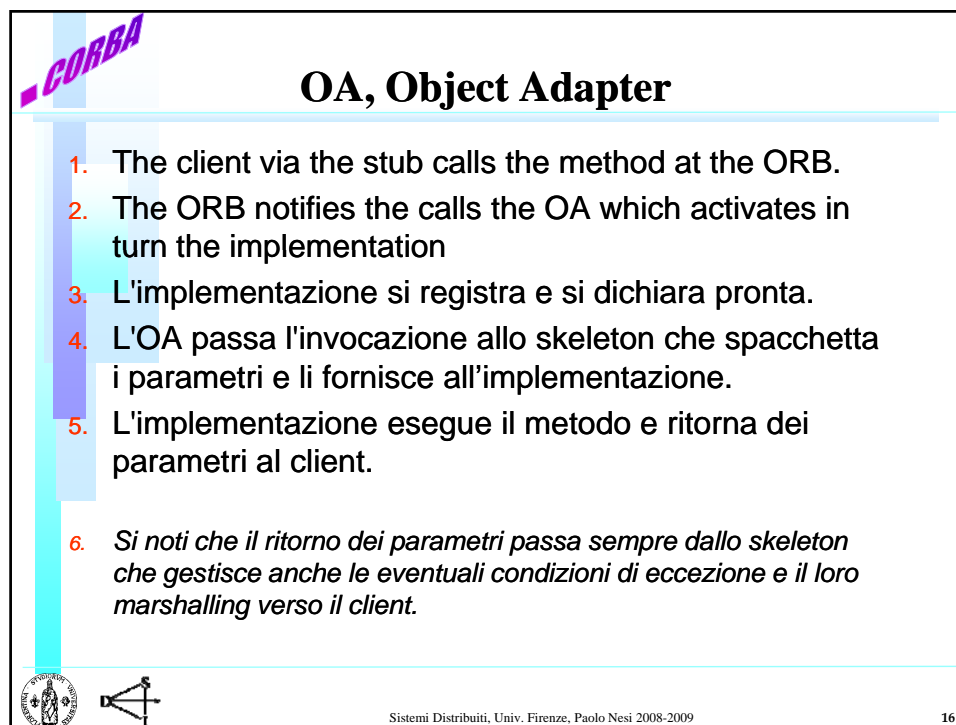
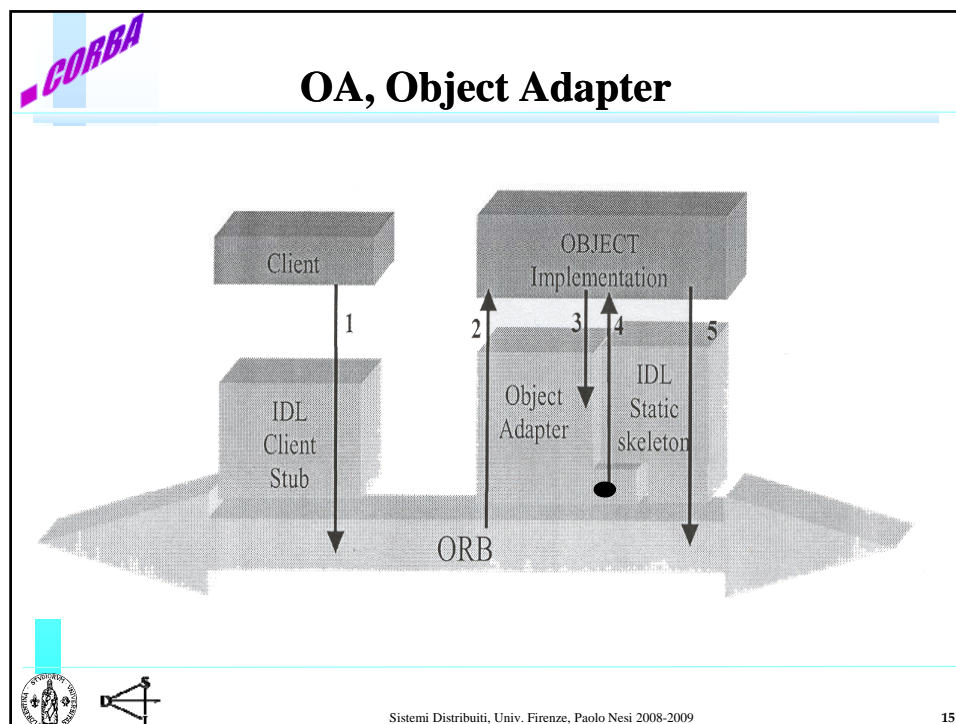
Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 13

CORBA

Object Adapter

- **Object Adapter (Portable object adapter)**
 - ♣ provides ORB services to particular groups of object implementations
- **Services and duties/activities**
 - ♣ generation and interpretation of object references, mapping object references to implementations, and registration of implementations.
 - ♣ method invocation (dispatching) via a skeleton, object and implementation activation and deactivation
 - ♣ security of interactions (method access control, etc.)

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 14



CORBA

OA, Object Adapter

- ♣ **The Object Adapter (OA)**
 - to interface an object's implementation with its ORB.
- ♣ **Three Object Adapters.**
 - **Basic Object Adapter (BOA)**
 - Provides CORBA objects with a common set of methods.
 - CORBA object's interface to the ORB
 - available in every ORB implementation
 - Includes user authentication, object activation, object persistence ,etc...
 - **Library Object Adapter (LOA)**
 - **Object-Oriented Database Adapter (OODA)**
 - Both LOA and OODA are useful for accessing objects in persistent storage

17

CORBA

The main components of the CORBA architecture

- **Interface repository**
 - ♣ the interface repository provides information about registered IDL interfaces to clients and servers that require it.
- **Implementation repository**
 - ♣ activates registered servers on demand and locates running servers
 - ♣ uses the object adapter name to register and activate servers.
 - ♣ more about this later

The diagram illustrates the CORBA architecture components and their interactions. It is divided into two main sections: 'client or dynamic invocation' and 'server or dynamic skeleton'.



- Client (or dynamic invocation):** Contains a 'client program' which interacts with a 'proxy for A' and an 'ORB core'.
- Server (or dynamic skeleton):** Contains an 'ORB core', an 'object adapter', a 'skeleton', and a 'Servant A'.
- Repositories:** An 'implementation repository' and an 'interface repository' are positioned between the client and server.
- Interactions:**
 - A 'Request' arrow points from the client's ORB core to the server's ORB core.
 - A 'Reply' arrow points from the server's ORB core back to the client's ORB core.
 - Red arrows from the text above point to the 'implementation repository' and 'interface repository' boxes.

19

CORBA

Interface repository

- provides information about registered IDL interfaces
 - ♣ for an interface it gives: method names and the names and types of the arguments and exceptions. (a short version of "Manifesto")
 - ♣ It is a facility for reflection in CORBA
 - Having a remote reference to a CORBA object, it is possible to ask at the interface repository about its methods and their parameter types
 - the client can use the dynamic invocation interface to dynamically construct an invocation with suitable arguments and send it to the server.
- the IDL compiler gives a Type ID to each IDL type, which is
 - ♣ included in remote object references
 - ♣ used also as a Type repository ID
- Applications that use static invocation with client proxies and IDL skeletons do not require an interface repository.
 - ♣ Not all ORBs provide an interface repository.



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 20

CORBA

Concetto di Manifesto

- Descrizione della classe/"component software"
- Potrebbe includere:
 - ♣ Nome, descrizione, creatore, produttore, data, versione, sistema operativo, etc.
 - ♣ Interfaccia di uso, o interfacce per l'uso, metodi e loro signature, etc.
 - ♣ Informazioni di trading: costo, DRM, location per il download di aggiornamenti, scadenze, etc.
 - ♣ Dipendenze da altri componenti
- CORBA e' limitato, per esempio:
 - ♣ ORB o chi per lui non e' in grado di prendere decisioni su quali diverse implementazioni scegliere, etc..
 - ♣ Le interfacce/implementazioni devono essere note agli ORB non possono arrivare dall'esterno del sistema



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 21



CORBA

Implementation repository

- The Implementation repository
 - ✦ activates registered servers on demand and locates running servers
 - ✦ uses the object adapter name to register and activate servers
 - ✦ stores a mapping from the names of object adapters to the pathnames of files containing object implementations.
 - when a server program is installed it can be registered with the implementation repository.
 - when an object implementation is activated in a server, the hostname and port number of the server are added to the mapping.
 - ✦ Implementation repository entry:

object adapter name	pathname of object implementation	hostname and port number of server
---------------------	-----------------------------------	------------------------------------

 - not all CORBA objects (e.g. call backs) need be activated on demand
 - access control information can be stored in an implementation repository



 

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 22

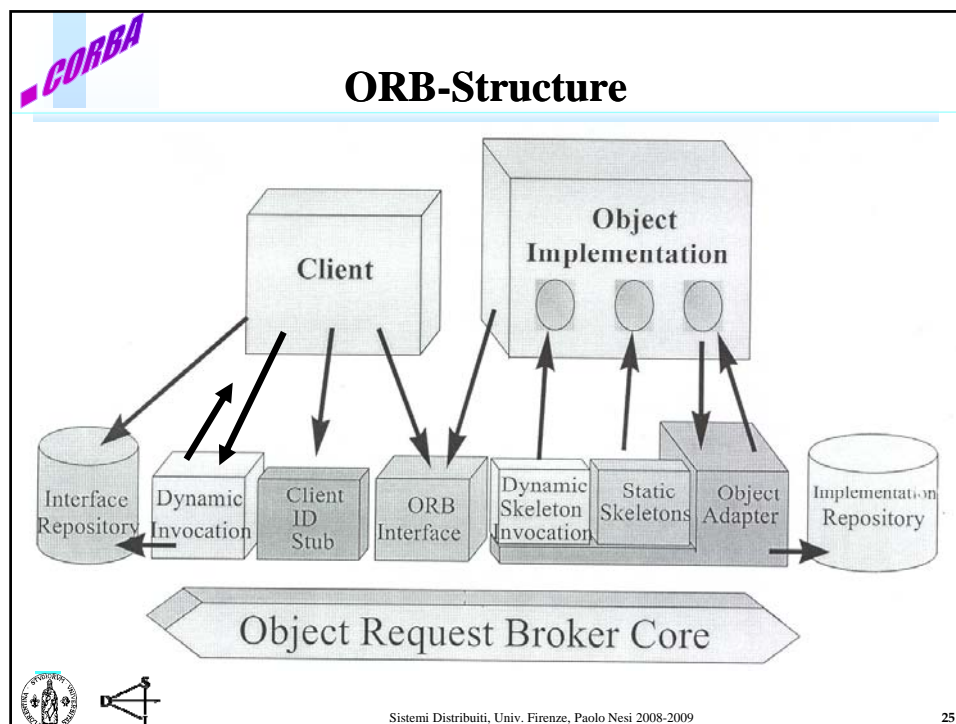
CORBA

ORB

- **Receives invocation** message to invoke specified method for registered object
- Finds object, unmarshals parameters, invokes method, marshals and returns results
- Requester needs not to be aware of location, language or OS of object

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 24





- CORBA**
- ## ORB-invocation, Client
- **ORB Interface**
 - ♣ Identification of the objects
 - ♣ String to objects and viceversa (Marshalling)
 - **Client IDL Stubs**
 - ♣ Static interface to object services, precompiled stubs
 - **Dynamic Invocation Interface, DII**
 - ♣ Permette di identificare i metodi che possono essere chiamati a run-time.
 - ♣ CORBA permette di definire i metadata, l'interfaccia, dei servizi
 - **Interface Repository**
 - ♣ Database con tutte le interfacce possibili e registrate in base agli oggetti disponibili
- Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 26

CORBA

ORB-provider part, Server

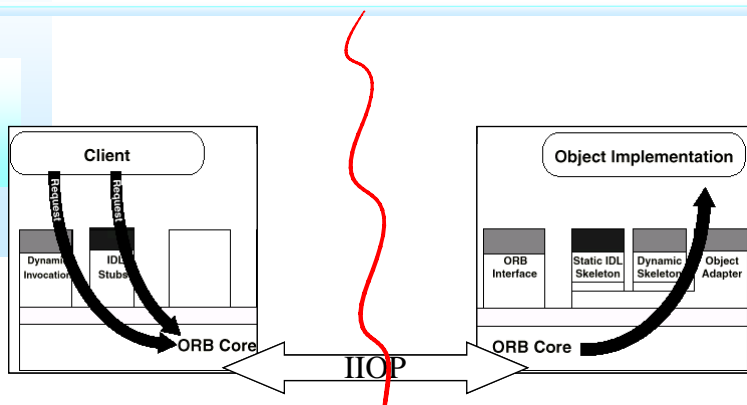
- **ORB Interface verso il Server**
 - ✦ Come quella da lato client
- **Static Skeleton equivalente al Server IDL Stubs**
 - ✦ Interfaccia statica dei servizi esportati dal server con IDL
- **Dynamic Skeleton Interface, DSI**
 - ✦ Interfaccia dinamica per la pubblicazione di servizi dinamici
 - ✦ Interface repository
- **Object Adapter / Dispatcher**
 - ✦ Accetta richieste di servizio per il Server
 - ✦ Istanza oggetti distribuiti e gli assegna richieste
 - ✦ Fa uso del Implementation Repository
- **IR, Implementation Repository**
 - ✦ Tabelle di classi e loro ID



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 27

CORBA

Data Flow/Communic in CORBA



The diagram illustrates the interaction between a Client and an Object Implementation. On the left, the Client contains Dynamic Invocation, IDL Stubs, and an ORB Core. On the right, the Object Implementation contains ORB Interface, Static IDL Skeleton, Dynamic Skeleton, Object Adapter, and an ORB Core. A red wavy line separates the two sides. A double-headed arrow labeled 'IIOP' connects the ORB Cores. Curved arrows show 'Request' flowing from the Client's Dynamic Invocation to its ORB Core, and from the Object Implementation's ORB Core to its Object Adapter. A legend at the bottom explains the shading: grey for interfaces identical for all ORB implementations, black for stubs and skeletons, and white for ORB-dependent interfaces.



 

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 28

CORBA

CORBA Communications Model

- **Inter-ORB protocols:**
 - ♣ IIOB (Internet InterORB Protocol),
 - IIOB = GIOP in TCP/IP
 - IIOB is a specialization of the GIOP.
 - IIOB = TCP/IP + GIOP
 - ♣ GIOP (General Inter-ORB protocol)
 - a high level standard protocol for communication between various CORBA ORBs and components.
 - GIOP can use other protocols such as TCP/IP and DCE.
 - Cannot be used alone, it has to be used with a protocol.





Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 29

CORBA

CORBA Characteristics

- Object-Oriented Programming
- Support multiple languages
 - ♣ Official: JAVA, C, C++, Smalltalk, COBOL
 - ♣ Also: eiffel, modula, perl, TCL, Python, etc.



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 30

CORBA

What are Objects in CORBA !!

- Objects are abstract: not realized by any particular technology
 - ♣ An object system is a collection of objects that isolates the requestor of services (clients) from the providers of services by a well-defined **encapsulating interface**
- Objects “talk” through requests: operation, target object, zero or more parameters, optional request context
- Objects are described with interfaces
 - ♣ operations (methods)
 - ♣ attributes (properties)
 - ♣ Standard data types are supported
 - object references
 - Any

```

graph LR
    Value --- OR[Object Reference]
    Value --- BV[Basic Value]
    Value --- CV[Constructed Value]
    CV --- S[Struct]
    CV --- Seq[Sequence]
    CV --- U[Union]
    CV --- A[Array]
    
```

Short
Long
LongLong
UShort
ULong
ULongLong
Float
Double
LongDouble
Fixed
Char
Wchar
String
Wstring
Boolean
Octet
Enum
Any

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 31

CORBA

CORBA and IDL

- **Interface Definition Language (IDL)**
 - ♣ The OMG-standard language for defining the interfaces for all CORBA objects.
 - ♣ An IDL interface declares a set of operations, exceptions, and attributes.
 - ♣ Each operation has a signature, which defines its name, parameters, result and exceptions.
 - ♣ Format of messages, external data representation in CDR
- **Below there is the**
 - ♣ **Internet InterORB Protocol (IIOP)**
The OMG-specified network protocol for communicating between ORBs fo different vendors. Based on TCP.

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 32

CORBA



Interface Definition Language

- Language neutral specification

```
interface Polynomial : MathObject {
    sequence<Monomial> monomials;
    int rank;
    Polynomial add(in Polynomial p);
};
```

- Mappings to several languages
- Tools (compilers) generate stubs and skeletons in various languages

Note. No way to know at run-time which interfaces an objects provides: IDL is compiled away
Dynamic taking of an interface, but it has to be created in advance



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 33

CORBA

Interface Definition Language (IDL)

- General Properties of IDL
 - ♣ Case sensitive
 - ♣ Definition syntax is the same as C++ definition syntax.
 - ♣ Assumes the existence of a C processor to process constructs such as macro definitions and conditional compilation
 - ♣ An example; The Module:

```
module Bank {
    interface Customer {
        ....
    };
    interface Account {
        ....
    };
};
```






Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 34

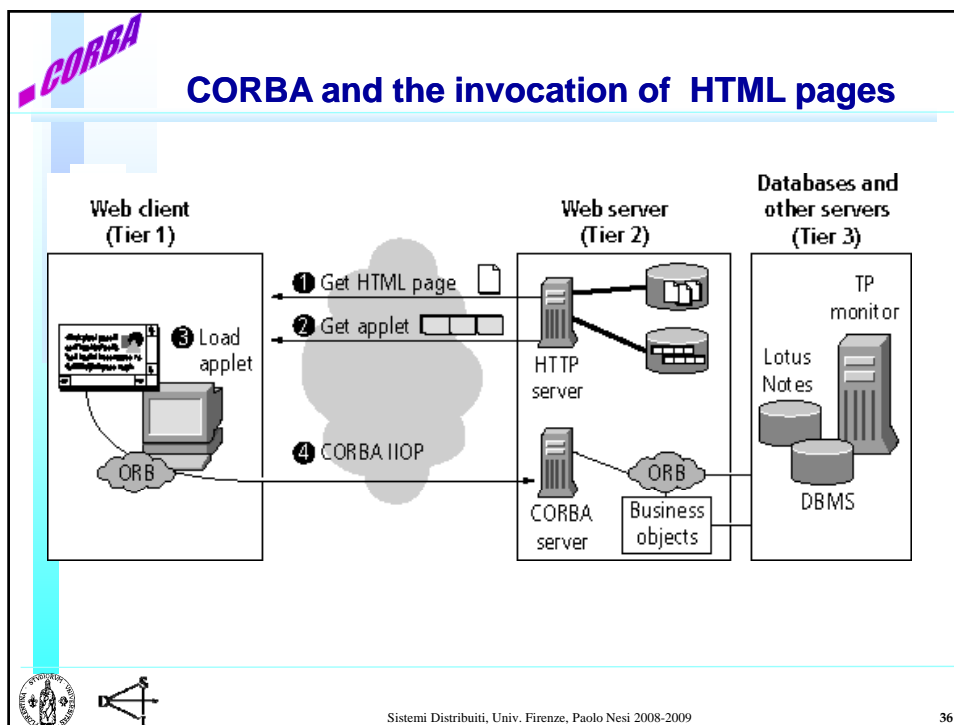
CORBA

Interface Definition Language (IDL)

- ♣ Primitive types: void, Boolean, char, wchar,
- ♣ Floating point types: float, double and long double
- ♣ Integer types: long, long long, unsigned long etc.
- ♣ Constructed types: enum, struct, union, etc.
- ♣ The interface type: in, out, inout
- ♣ Attributes: readonly..
- ♣ Other IDL constructs
 - typedef
 - forward declaration
- ♣ Container types: sequence, array.
- ♣ The Exception type
- ♣ The Any type
- ♣ The TypeCode Pseudotype



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 35



CORBA

Building a CORBA Application

- Step1: Write IDL interfaces for Server: Server.idl
- Step2: Compile IDL file and generate Server_c.cpp and Server_s.ccp
- Step3: Write server implementation in C++: ServerMain.ccp
- Step4: Compile the ServerMain.ccp with the files created by IDL
- Step5: Write IDL interface for Client: Client.idl
- Step6: Compile client.idl and generate associated java files such as ServerSymbolHelper.java and ServerSymbolListHelper.java etc.
- Step7: Write client implementation in Java
- Step8: Compile client implementation and helper files together
- Step9: Run server and client programs together



 

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 37

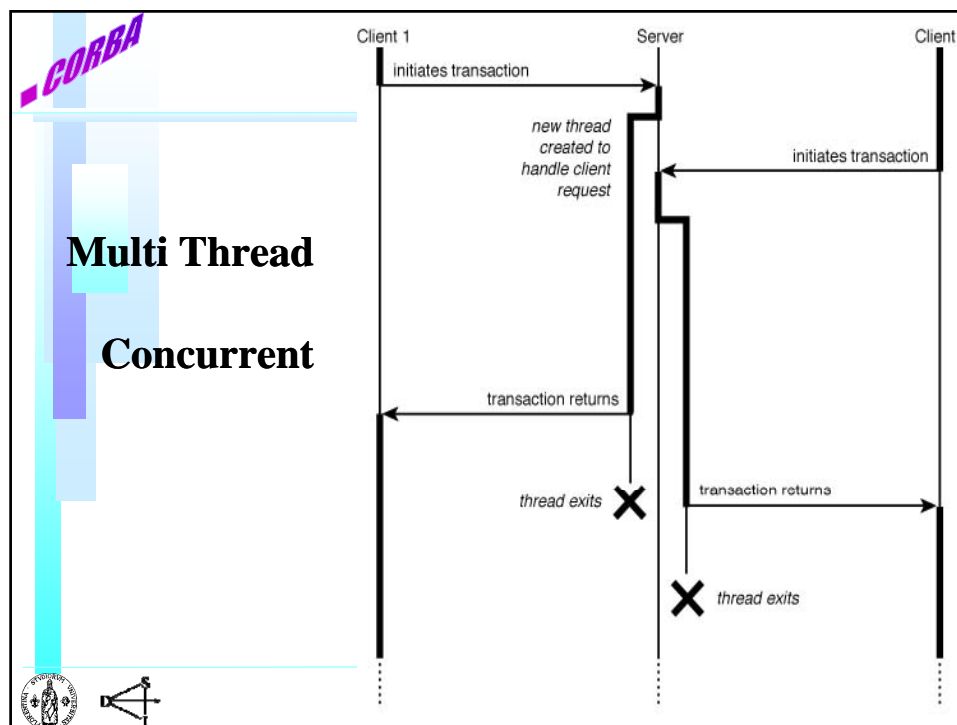
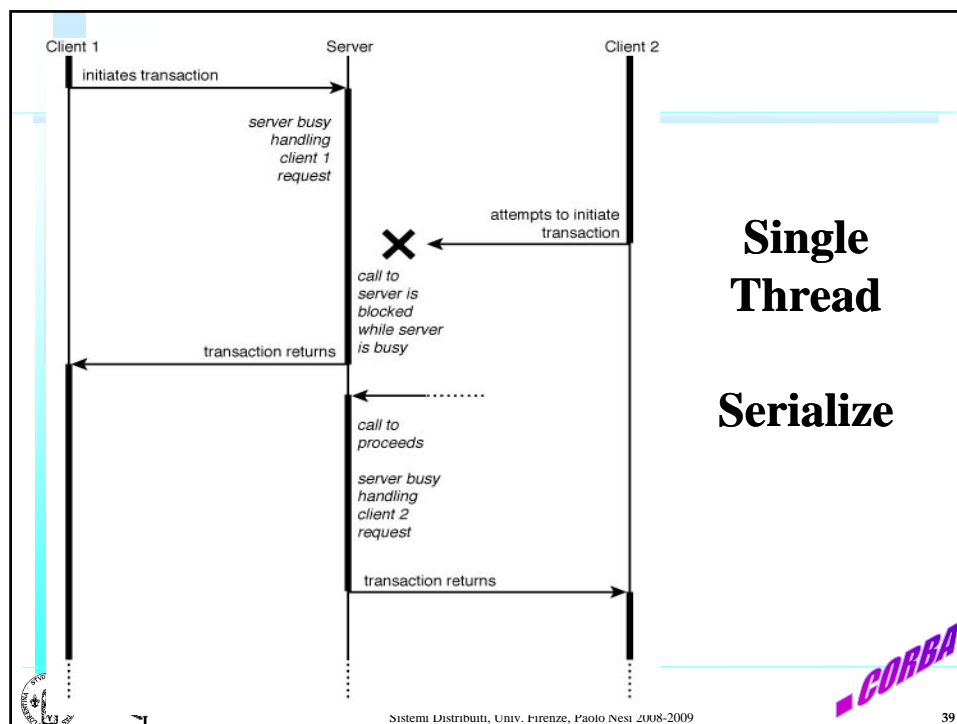
CORBA

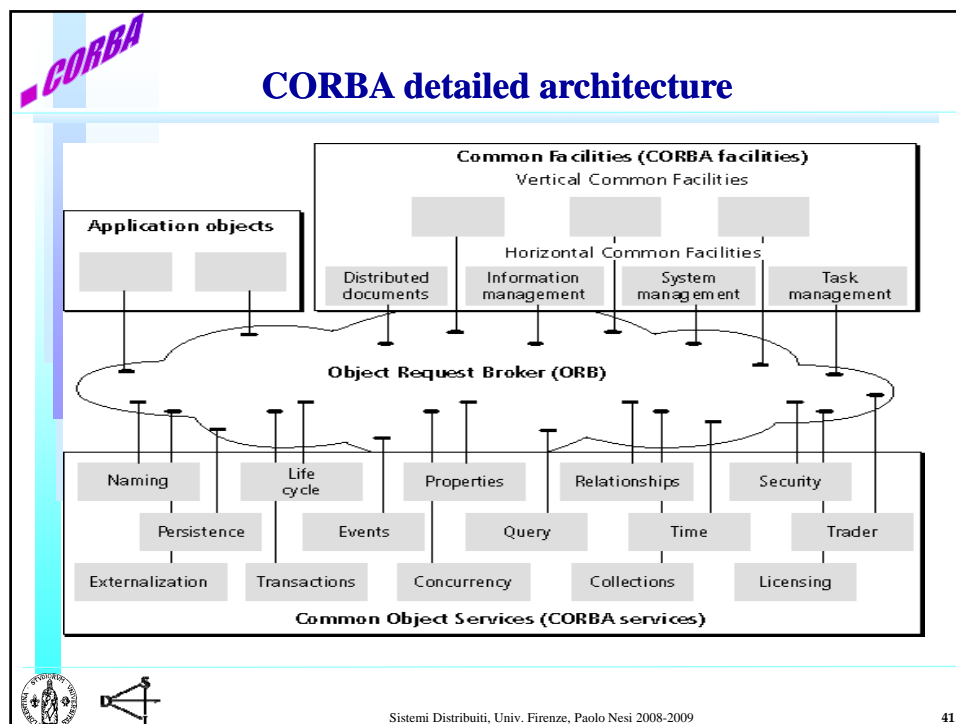
CORBA Design Issues

- **Single-Threaded Applications:**
 - ♣ Common and Easy.
- **Multi-Threaded Applications: Limited**
 - ♣ Not all the Operating systems supports it.
 - ♣ Not all the developers are using the later versions of OSs.
 - ♣ Introduces new issues: the need to manage concurrent access to objects.
- Server Applications
- Client Applications
- Mixed Server/Client Applications
- Object Life Time

Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 38







- CORBA**
- ## CORBA Services
- >20 defined services
 - check vendor for implementation and limitations !
 - Some interesting services
 - ♣ Naming Service
 - "directory-based"
 - single or federated
 - ♣ Event Service: suppliers and consumers communicate via an event channel
 - decouples suppliers from consumers
 - push or pull models
 - uses Any for event data
 - Notification Service ?
 - Messaging Service ?
- CORBA**
- Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 42

CORBA

CORBA services include the following

- **Notification Service:**
 - ♣ Event Service suppliers and consumers communicate via an event channel
 - ♣ NS extends Event Service to allow filtering and typed events
- **Security service:**
 - ♣ authentication of principals and access control of CORBA objects with policies
 - ♣ auditing by servers, facilities for non-repudiation
- **Trading service:**
 - ♣ allows CORBA objects to be located by attribute
- **Transaction service and concurrency control service**
 - ♣ TS provides flat or nested transactions
 - ♣ CCS provides locking of CORBA objects
- **Persistent object service:**
 - ♣ for storing the state of CORBA objects in a passive form and retrieving it





Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 43

CORBA

Summary

- **CORBA addresses heterogeneity:**
 - ♣ RMI between a client and a remote remote object in different languages.
 - ♣ GIOP
 - specifies an external data representation called CDR – clients and servers can have different hardware.
 - specifies OS independent operations for request-reply protocol
 - specifies a standard form for remote object references.
 - ♣ IIOP implements the request-reply protocol over TCP/IP.
- **Object adapter**
 - ♣ relates request messages to implementations of CORBA objects
- **Implementation repository**
 - ♣ enables CORBA objects to be activated on demand
- **Interface repository**
 - ♣ allows dynamic invocation of CORBA objects, some limitation on the manifest
- **IDL for defining interfaces**





Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 44

CORBA

More About CORBA

- Other features of CORBA
 - **vendor specific implementations - check performance you need !**
 - implementazioni diverse possono avere diversi servizi
 - Vendor diversi possono avere implementazioni anche in certa misura con semantica diversa,
 - anche se questo non dovrebbe accadere,
 - lo standard non descrive in modo preciso la semantica di certe operazioni
 - Gateways to DCOM and OLE automation exist !
 - CORBA Components (futureware, mai visto...)
 - Esistono versioni anche per sistemi mobili come PDA
 - Fortemente limitate
 - Esistono versioni che permettono di avere un maggiore controllo sui problemi di real time
- Alternatives to CORBA:
 - .net
 - J2EE
 - ...





Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 45

CORBA

Reference

- **Visibroker, from Visigenic**
 - <http://www.cse.cuhk.edu.hk/~csc5340/material/vbj40java-reference.pdf>
- **Orbix from Iona**
 - <http://www.cse.cuhk.edu.hk/~csc5340/material/OrbixProgrammersGuide.pdf>
- **CORBA FAQ**
 - <http://www.omg.org/gettingstarted/corbafaq.htm>



Sistemi Distribuiti, Univ. Firenze, Paolo Nesi 2008-2009 46