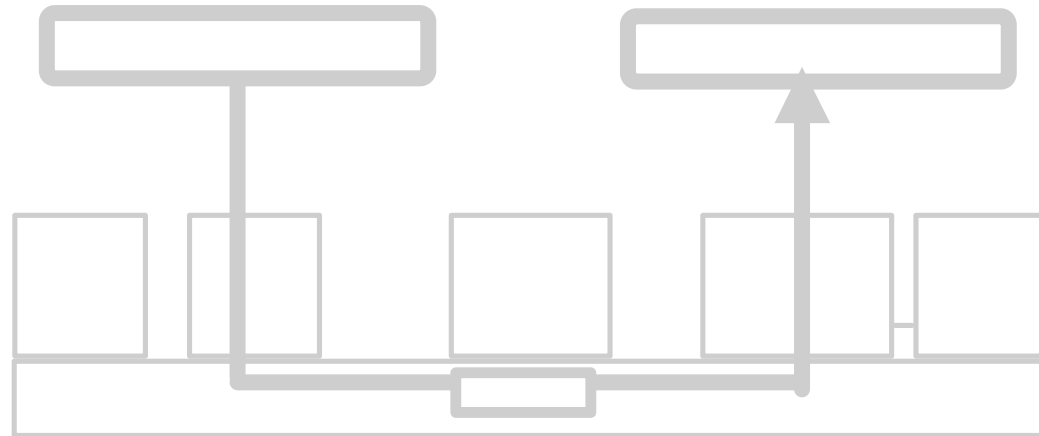


Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

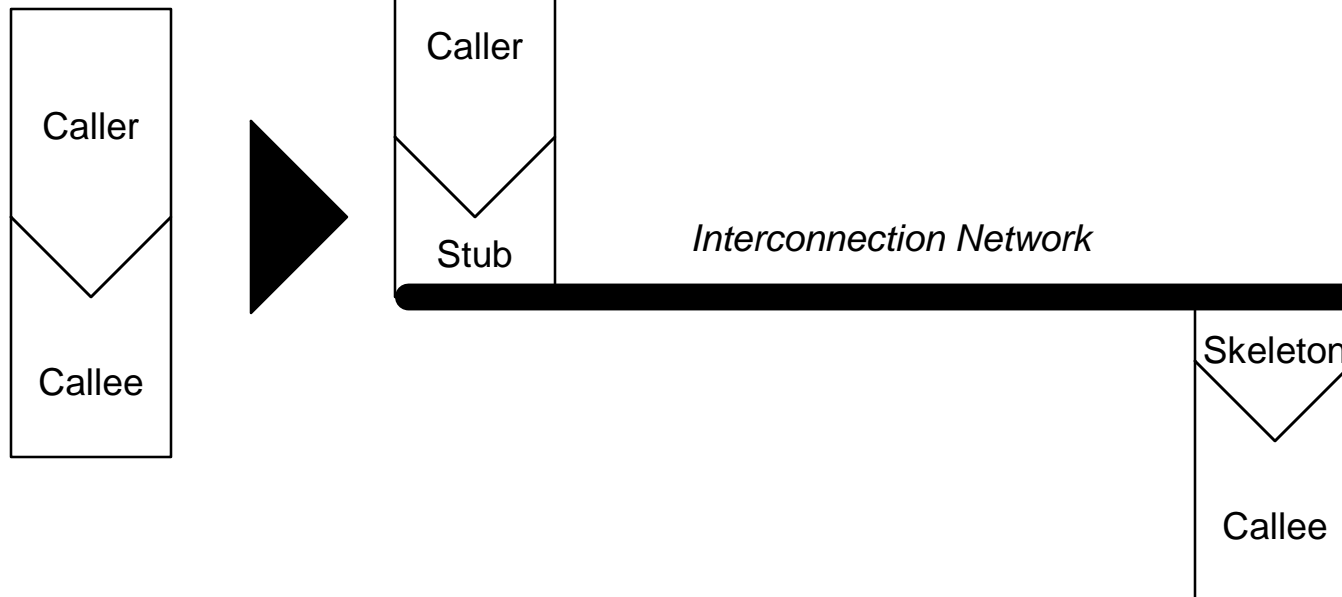
A Tour of CORBA

**A Brief Technical Tour through
the standard.....**



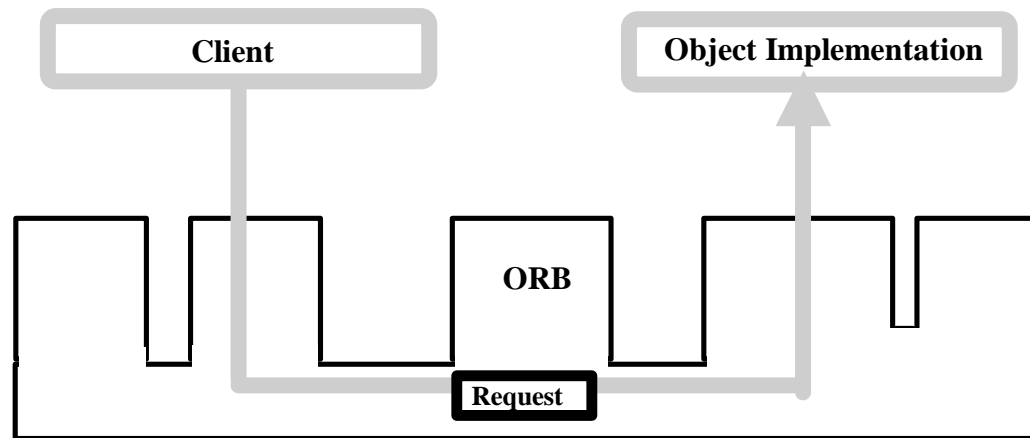
Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

A Quick History



Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

A Request



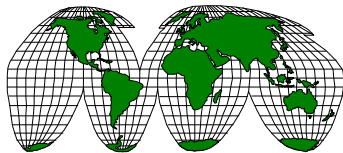
A request consists of:

- target object
- operation
- parameters
- optional request context

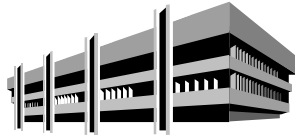
Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

A Scalable Architecture

Different Scales



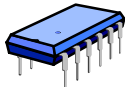
Widely distributed Objects



Network-Distributed Objects



Multi-process, local Objects



Lightweight, single-process Objects

Different Types

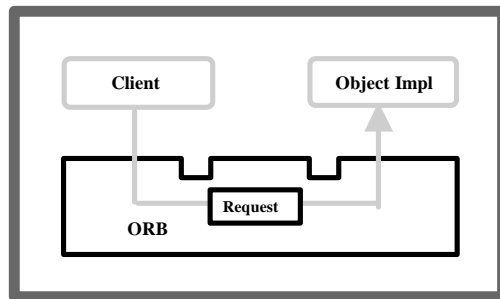
**OODB
Objects**

**OLTP
Objects**

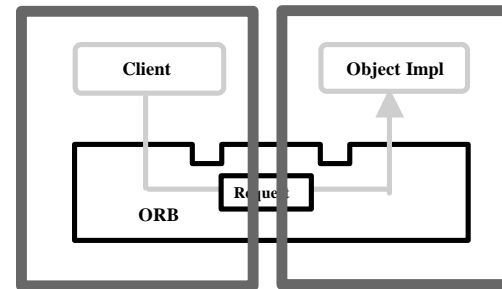
**C++
Objects**

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

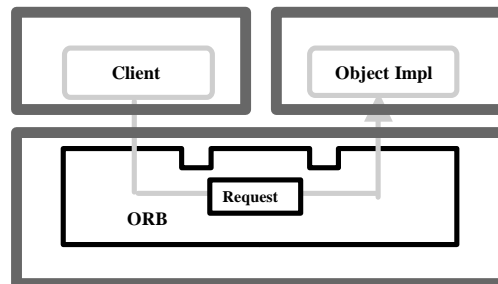
Different ORB Types



Single-Process
Library Resident



Client &
Implementation
Resident



Server or
Operating-System
Based

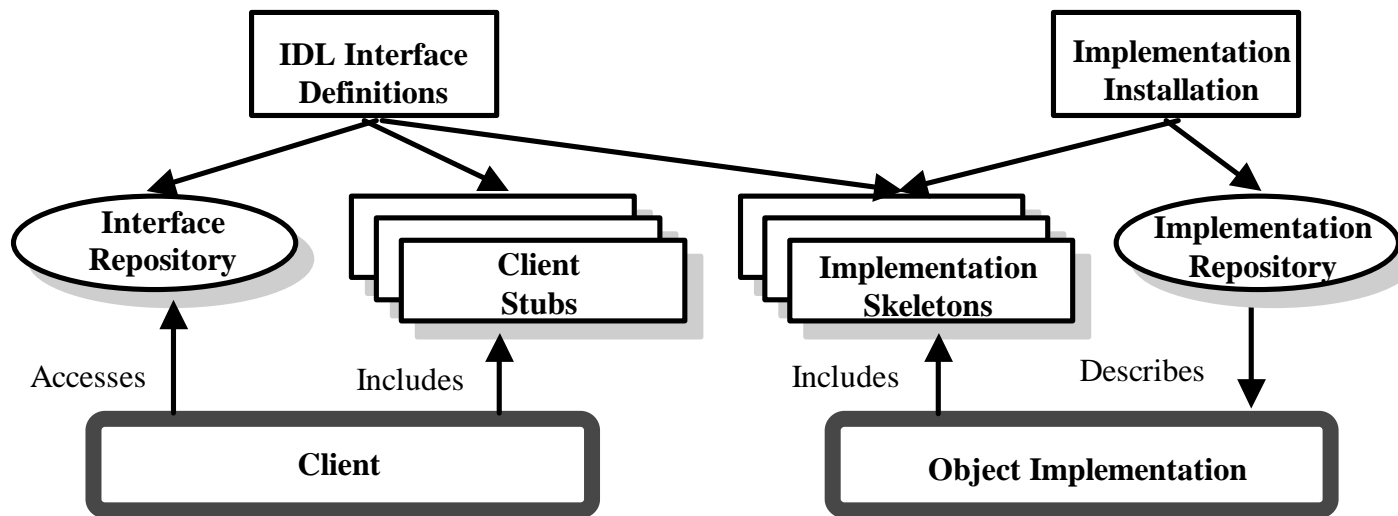
Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

OMG IDL

*Key component of the standard is **OMG Interface Definition Language (IDL)**:*

- mappings will be provided for many languages/compiler;
- independent of any particular language/compiler;
- multiple-inheritance, public interface-structured specification language;
- *not* for implementation.
- ***primary support for interoperability between static and dynamic requests mechanisms.***

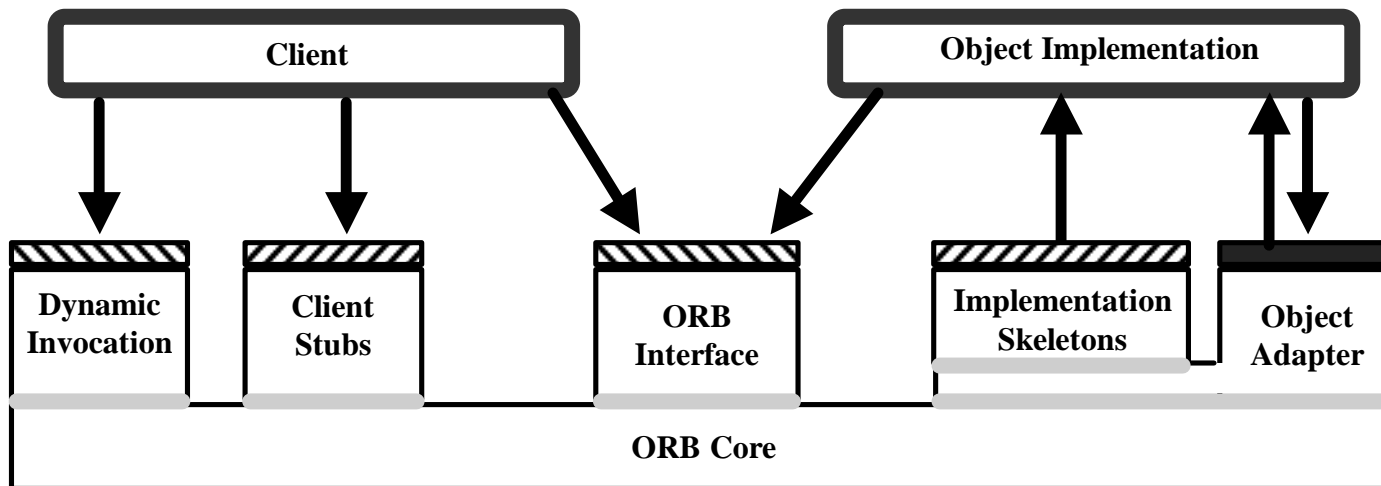
Interfaces



- *All objects are defined in IDL by specifying their interfaces.*
- *Object definitions (interfaces) are manifested as objects in the Interface Repository, as client stubs, and as implementation skeletons.*
- *Descriptions of object implementations are maintained as objects in the Implementation Repository.*

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

CORBA Components



One interface



One interface per object adaptor



One interface per object operation



Proprietary interface

Normal call interface

Up call interface

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

CORBA Components

Client stub

Each stub represents an object operation (a possible request) which a client invokes in a language-dependent manner (e.g., by calling a subroutine which represents the operation).

Dynamic Invocation

Alternatively, a client may dynamically construct and invoke request objects which can represent any object operation.

Implementation Skeleton

Each skeleton provides the interface through which a method receives a request.

Object Adapter

Each object adapter provides access to those services of an ORB (such as activation, deactivation, object creation, object reference management) used by a particular ilk of object implementation.

ORB Interface

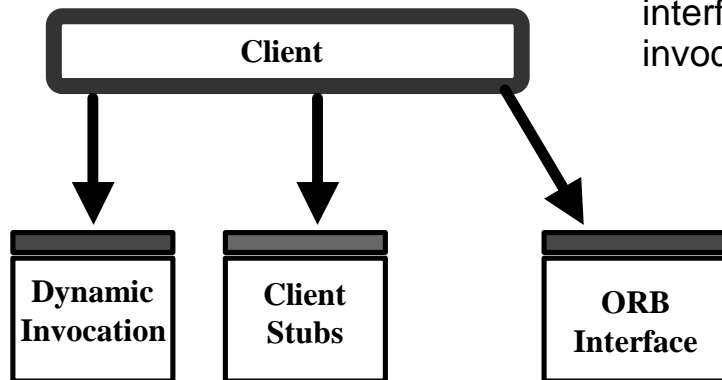
The interface to the small set of ORB operations common to all objects, e.g., the operation which returns an object's interface type.

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

Client Side

Clients perform requests using object references.

Clients may issue requests through object interface stubs (static) or dynamic invocation interface.



Clients may access general ORB services:

- Interface Repository.
- Context Management.
- List Management.
- Request Management.

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

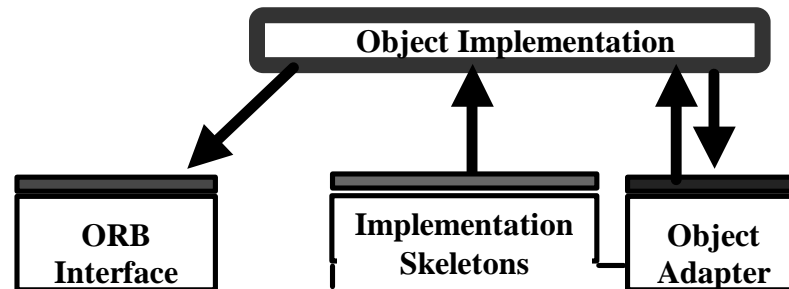
Implementation Side

Implementations receive requests through skeletons
(without knowledge of invocation approach).

The Object Adapter adapts to vagaries
of object implementation scheme.

The Basic Object Adapter provides for:

- management of references;
- method invocation;
- authentication;
- implementation registration;
- activation/deactivation.



Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

Examples ORB's

Client-and implementation-resident

ORB implemented as libraries (routines) resident in the clients and in the implementations.

Library-resident

ORB and implementations implemented as libraries (routines) resident in the client.

Server-based

ORB is implemented as a server (separate process) which brokers requests between client and implementation processes.

System-based

ORB is part of the operating system.

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

Example Adapters

Basic Object Adapter

Intended for implementations that are separate programs (processes) with no "ORB-like" services, the basic adapter provides for object reference generation and management, method invocation and request delivery, implementation registration, activation and deactivation.

OODB Adapter

As OODB's provide some "ORB-like" services (e.g., object reference generation and management), this adapter is tuned to integrate OODB's with ORB distribution and communication.

Library Adapter

Tuned for implementations resident in the client's process space, this adapter provides minimal implementation management and high-performance data transfer.

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

Dynamic Requests

The Dynamic Invocation Interface (DII) allows clients to dynamically:

- **discover objects;**
- **discover objects' interfaces;**
- **create requests;**
- **invoke requests;**
- **receive responses.**

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

DII Features

Major features of Dynamic Invocation Interface:

- **requests appear as objects themselves;**
- **requests are reusable;**
- **invocation may be synchronous or asynchronous; requests may be generated dynamically, statically or in combination approach.**

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

Request Components

Object reference	identifies the target object.
Operation	identifies which operation to perform.
Parameters	the input, output and in-out parameters defined for the operation; may be specified individually or as a list.
Context object	the context within which the request is to be performed.
Results	the result values returned by the operation.
Environment	the execution environment and exception information of the operation.
Request handle	the identifier for this instance of the request.

Title: OMG Logo.FH3
Creator: FreeHand 3.1
CreationDate: 3/22/92 9:59

Interface Repository

Integrated Interface Repository provides:

- **Dynamic client access to interface definitions to construct a request.**
- **Dynamic type-checking of request signatures.**
- **Traversal of inheritance graphs.**
- **ORB-to-ORB interoperability.**

Repository Structure

The interface repository service is defined as a set of objects specified in IDL:

