

CORBA - A Closer Look

- Overview
- CORBA 3.0 Overview
- Closer Look at OMA
- Object location and object adapters
- Event and messaging services
- CORBA Interoperability

Amjad Umar

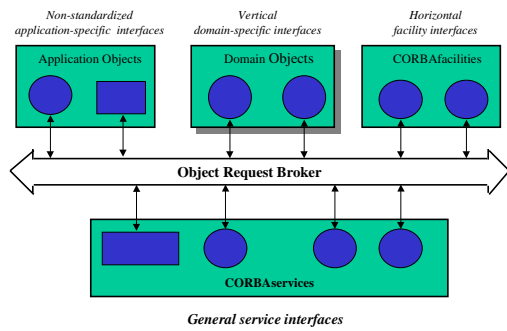
CORBA Basic Notions

- The rationale for CORBA: Middleware Interoperability
 - The Object Request Broker
 - CORBA for Internet applications
 - Services
 - Success stories, so far
- CORBA Core Reference Model and Architecture
 - CORBA IDL -- Interface Definition Language
 - Compile-time model: Stubs and Skeletons
 - Run-time model: remote object invocation
 - Object references
 - Basic CORBA Object Model

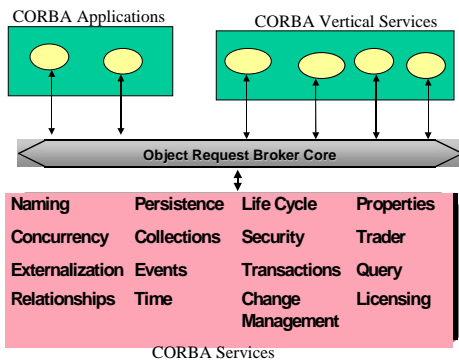
The Object Management Architecture

- Support for Analysis and Design
 - UML & MOF
- Basic Object-Oriented Computing Model
 - ORB, ISO/OMG IDL and language mappings
- Distribution
 - GIOP and its mapping (IIOP), messaging and async. invocation
- Component Model (under dev.)
- Specialized Models
 - Real-time, Fault-tolerant, Minimum CORBA
- CORBA Services
 - among which naming, trader, event and notification, transaction and security
- Horizontal facilities
 - Systems management, print spooling
- Vertical Domain facilities

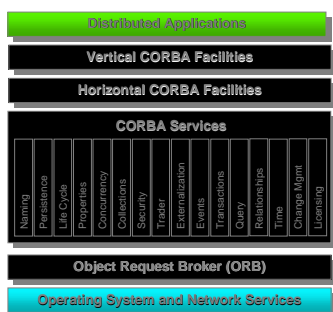
The OMA Architecture



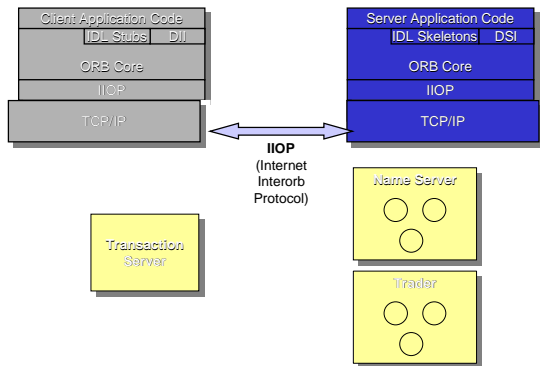
CORBA Services



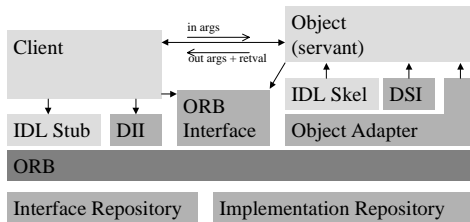
CORBA Architecture



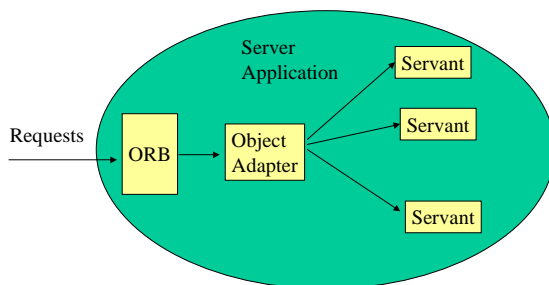
CORBA Physical View



Locating Objects and Object Adapter

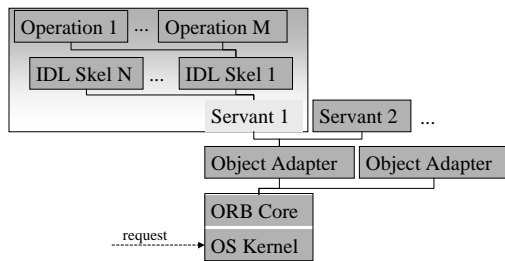


Object Adapter Conceptual View



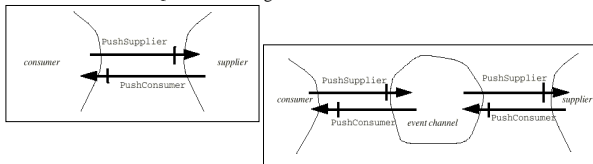
Portable Object Adapter

Object {
 state ↔ object state = attributes and specific info
 behavior ↔ object servant = code and resources
 identity ↔ object reference → object id



The Event Service

- Event Service
 - <[ftp://ftp.omg.org/pub/docs/formal/98-07-05.pdf](http://ftp.omg.org/pub/docs/formal/98-07-05.pdf)>
 - asynchronous interaction model
 - publish/subscribe model of event distribution
 - pull (consumer initiated) or push (producer initiated) model
 - channels
 - event content packaged into an Any
 - basic QoS and filtering



OMG IDL CosEventComm Module

```

module CosEventComm {
    exception Disconnected{};
    interface PushConsumer {
        void push (in any data) raises(Disconnected);
        void disconnect_push_consumer();
    };
    interface PushSupplier {
        void disconnect_push_supplier();
    };
    interface PullSupplier {
        any pull () raises(Disconnected);
        any try_pull (out boolean has_event);
        raises(Disconnected);
        void disconnect_pull_supplier();
    };
    interface PullConsumer {
        void disconnect_pull_consumer();
    };
};
  
```

Structured Events

Event filter example: `((($domain_type == "Telecom" and Sevt_type == "CommunicationsAlarm") or ($domain_type == "Transport" and Sevt_type == "RoadImpassable"))) and severity != 4"`

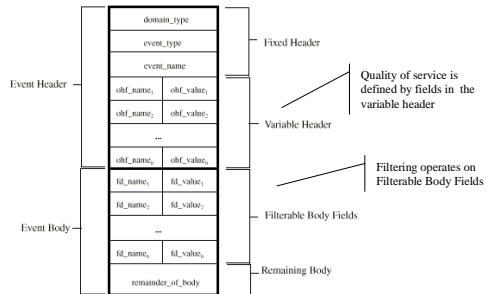


Figure 2-2 The structure of a Structured Event

CORBA IDL

General Inter-ORB Protocol (GIOP)

Environment Specific Inter-ORB Protocol (GIOP)

Internet Inter-ORB Protocol (IIOP)
TCP/IP

Others
For example (SPX, ATM, SS7)

DCE RPC over TCP/IP

Others
For example Wireless CORBA

Legend

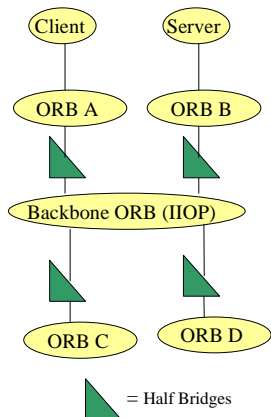


Mandatory for CORBA



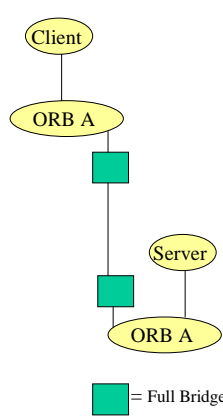
Optional for CORBA

A) Half Bridges



= Half Bridges

B) Full Bridges



= Full Bridges
