



CORBA

Common Object Request
Broker Architecture

CORBA

What is it? [1]

■ **Object Request Broker**

- A common communication bus for transparently message passing between objects

■ **Object Services**

- a collection of services (interfaces and objects) that support basic functions for using and implementing objects.

CORBA

What is it? [2]

■ **Common Facilities**

- a collection of services that many applications may share, but which are not as fundamental as the Object Services.

■ **Application Objects**

- products of a single vendor or in-house development group that controls their interfaces.

CORBA

History – 1.0 (Oct. 1991~Dec.1993)

- CORBA Object model (Core 92)
- Interface Definition Language (IDL)
- Mapping from IDL to the C language.
- the core set of APIs for dynamic request management and invocation and Interface Repository.
- interfaces for the Basic Object Adapter and memory management

CORBA

History – 2.0~2.3 (Aug.96~ Jun.99)

- General Inter-ORB Protocol / Internet Inter-ORB Protocol (GIOP/IIOP)
- Portable Object Adapter (POA)
- Collaboration with OLE2/COM
- Mapping from IDL to Java, Cobol, Ada, Smalltalk, C++

CORBA

History – 2.4~3.0 (Feb. 2001~ ??)

- Java and Internet Integration
 - Objects Passable by Value
 - Java-to-IDL Mapping
 - Interoperable Name Service
- Asynchronous Messaging and Quality of Service Control
- Minimum, Fault-Tolerant, and Real-Time CORBA
- CORBAcomponents and CORBAscripting

CORBA

CORBA versus RPC

- RPC: When a specific function is called, the data type of parameters are fixed
- CORBA: The polymorphism of object is more flexible.
- RPC: Language dependent
- CORBA: Language independent
- RPC: Get operation only
- CORBA: Support push operation

CORBA

Advantages

- Static and dynamic method invocations
- High-level language bindings
- Self-describing system
- Location transparency
- Built-in security and transactions
- Coexistence with existing systems

CORBA Application Object Web Server

