



Sistemas Distribuídos e Concorrentes
Escola Politécnica – PUCPR
Luiz A. de P. Lima Jr. ▪ luiz.lima@pucpr.br

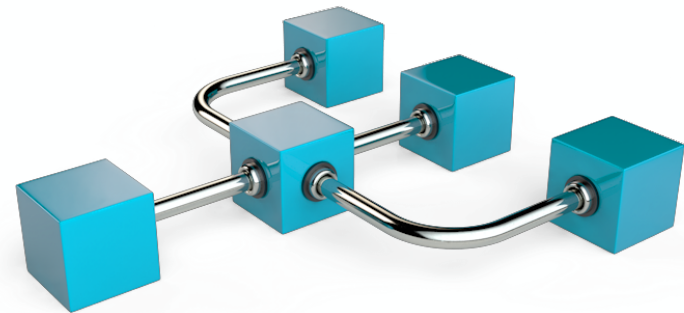
Objetos Distribuídos

Estudo de Caso: CORBA



Tópicos

- ◉ Introdução
- ◉ Transparências
- ◉ O Modelo de Programação
- ◉ IDL
- ◉ Arquitetura CORBA
- ◉ Invocações Estáticas e Dinâmicas
- ◉ O Modelo de Interoperabilidade CORBA





Introdução

Objetos Distribuídos – CORBA



Object Management Group (OMG)

- ◉ OMG: <http://www.omg.org>
- ◉ Consórcio:
 - Desenvolvedores
 - Organizações governamentais
 - Grupos de usuários
- ◉ Objetivos:
 - Promover tecnologias OO
 - Criar especificações
 - Definir padrões de interoperabilidade





A Especificação CORBA

- ◉ CORBA:
 - Suporte para **programação distribuída aberta**
 - Padrão aberto
 - **Interoperabilidade** + **serviços** adicionais
- ◉ RFP + RFC:
 - Muitas implementações comerciais



Implementações da especificação CORBA

- ◉ **ACE/TAO**
- ◉ ORBIX (Micro Focus)
- ◉ VisiBroker (Micro Focus)
- ◉ omniORB (+ omniORBpy)
- ◉ JacORB
- ◉ Orbacus
- ◉ Component Broker – WebSphere (IBM)
- ◉ ObjectBroker (BEA)
- ◉ JavaIDL (Sun)
- ◉ CORBAplus (Expersofts)
- ◉ COOL-ORB (Chorus Systems)
- ◉ Orblite (HP)
- ◉ DSOM (IBM)
- ◉ DOE (Sun)
- ◉ ISIS, Orblite, etc.



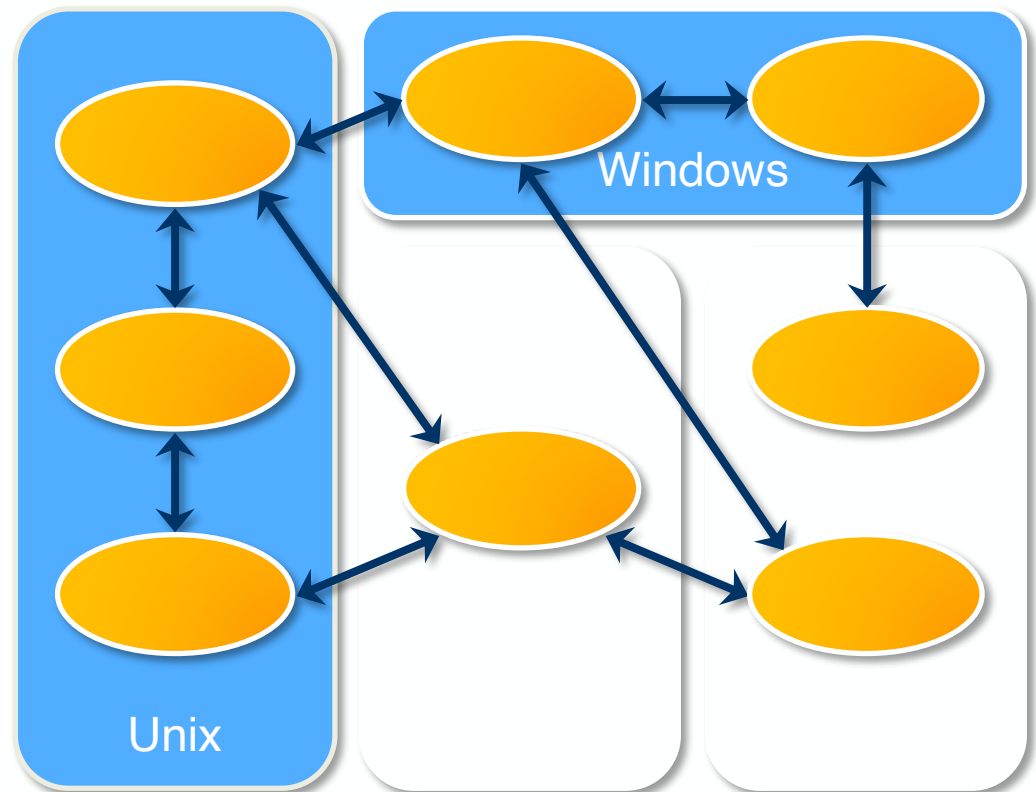
Implementações da especificação CORBA

- Produtos são:
 - Implementações das especificações CORBA mais ou menos conformes às normas e mais ou menos estendidas por serviços adicionais.
- Free CORBA:
 - ACE/TAO (C++)
 - JacORB (Java)
 - OmniCORBA, outros...



Objetivos de CORBA

Fornecer **interoperabilidade** entre objetos rodando em sistemas distribuídos (*heterogêneos*) e possibilitar a sua composição em aplicações.





Como garantir interoperabilidade?

1. Padronizar a “aparência externa” dos objetos (**interfaces**).
2. Propor um mecanismo genérico de ligações entre objetos (**bindings**).
3. Esconder as dificuldades da distribuição dos programadores (**transparências**).
4. Oferecer um conjunto padronizado de **serviços adicionais** aos programadores.



CORBA fornece...

- ◉ Uma base para computação OO **distribuída**:
 - Suporte para **invocação de métodos remotos**
 - **Referências transparentes** para objetos remotos
- ◉ Uma plataforma OO **aberta**:
 - **Independência de linguagens** de implementação
 - Transparência de HW e SO
- ◉ Um conjunto de **serviços** distribuídos
 - ***Naming, events, trader...***
 - Acessados como objetos CORBA