Case Study: OMG/CORBA

Outline

- Who is the OMG
- Goals of CORBA
- CORBA Object Model
- CORBA Interface Definition Language
- CORBA Architecture
  - Presentation Layer Implementation
  - Session Layer Implementation

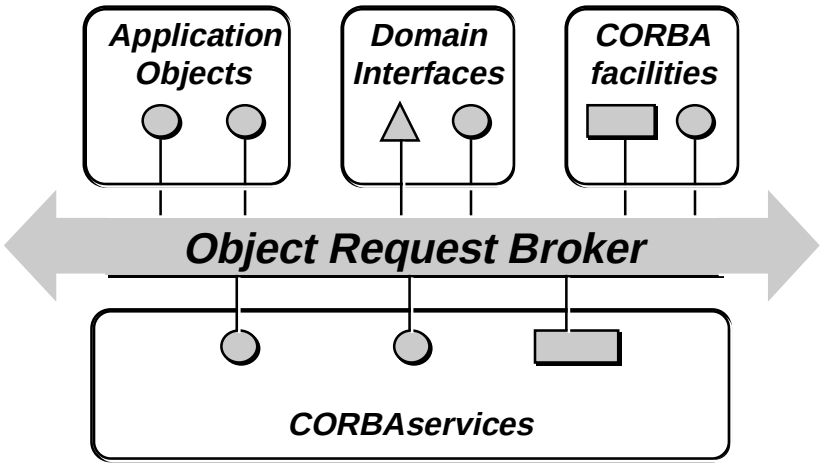
Who is the OMG?

- Non-profit organisation with HQ in the US, representatives in United Kingdom, Germany, Japan, India, and Australia.
- Founded April 1989.
- More than 800 members.
- Dedicated to creating and popularizing object-oriented industry standards for application integration, e.g.
  - CORBA
  - ODMG-93
  - UML

Goal of CORBA

- Support distributed and heterogeneous object request in a way transparent to users and application programmers
- Facilitate the integration of new components with legacy components
- Open standard that can be used free of charge
- Based on wide industry consensus

Object Management Architecture



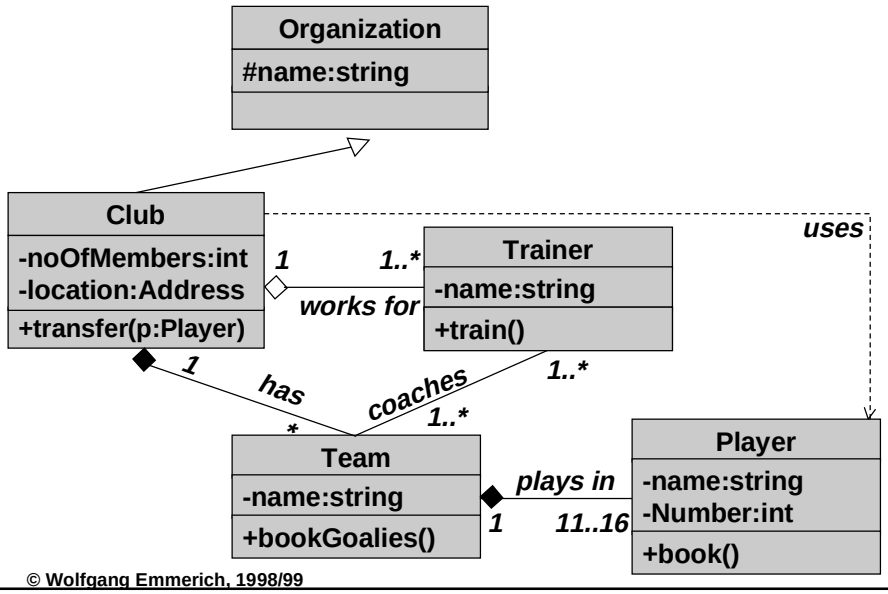
CORBA Object Model

- Objects
- Types
- Modules
- Attributes
- Operations
- Requests
- Exceptions
- Subtypes

OMG Interface Definition Language

- *Language for expressing all concepts of the CORBA object model*
- *OMG/IDL is*
  - *programming-language independent*
  - *orientated towards C++*
  - *not computationally complete*
- *Different programming language bindings are available*
- *Explanation of Model and Language by Example*

Running Example



CORBA Object Model: Objects

- *Each object has one identifier that is unique within an ORB*
- *Multiple references to objects*
- *References support location transparency*
- *Object references are persistent*

CORBA Object Model: Types

Object type

```
interface Organization {
  readonly attribute string name;
};
```

Constructed type

```
struct Address {
  string street;
  string postcode;
  string city;
};
```

Atomic types

CORBA Object Model: Attributes

Clients cannot change value

Attribute type

Attribute name

```
interface Team {
readonly attribute string name;
attribute sequence<Trainer> coached_by;
attribute Club belongs_to;
attribute sequence<Player> players;
 ...
};
```

changeable

Constructed attribute type

CORBA Object Model: Operations

Parameter kind

Parameter list

Return type

```
interface Team {
 ...
void bookGoalies(in Date d);
oneway void print();
};
```

Oneway Synchronization

Parameter type

Operation name used in requests

Parameter name

CORBA Object Model: Requests

- *Requests are defined by client objects*
- *Request consist of*
  - *Reference of server object*
  - *Name of requested operation*
  - *Actual request parameters*
  - *Context information*
- *Request is executed synchronously*
- *Requests can be defined*
  - *statically*
  - *dynamically*

CORBA Object Model: Exceptions

- *Generic Exceptions (e.g. network down, invalid object reference, out of memory)*
- *Type-specific Exceptions*

Exception name *Exception data*

```
exception PlayerBooked{sequence<Date> free;};
interface Team {
 ...
 void bookGoalies(in Date d) raises PlayerBooked;
};
```

Operations declare exceptions they raise

CORBA Object Model: Subtypes

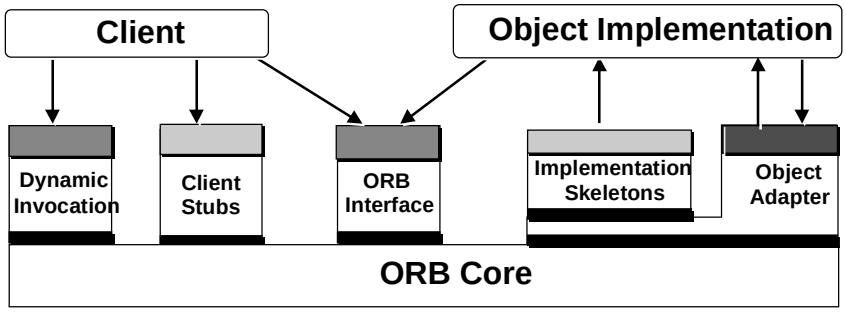
Implicit supertype: **Object**

Inherited by **Club**

```
interface Organization {
  readonly attribute string name;
 };
interface Club : Organization {
  exception NotInClub{};
  readonly attribute short noOfMembers;
  readonly attribute Address location;
  attribute sequence<Team> teams;
  attribute sequence<Trainer> trainers;
  void transfer(in Player p) raises NotInClub;
 };
```

Supertype

Components involved at run-time

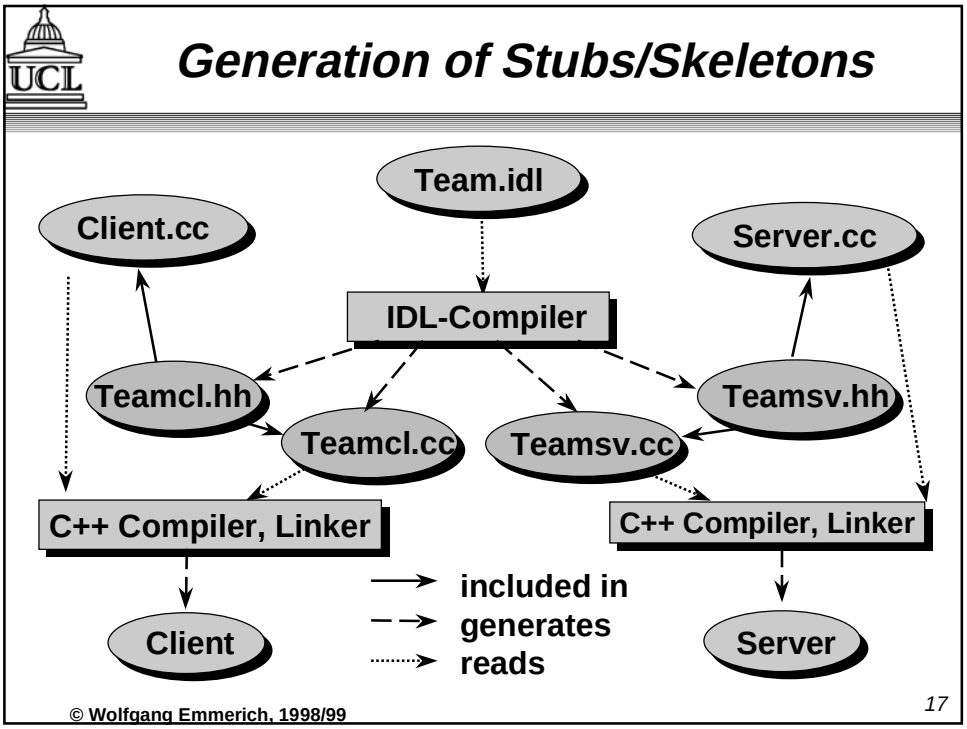


One standardised interface

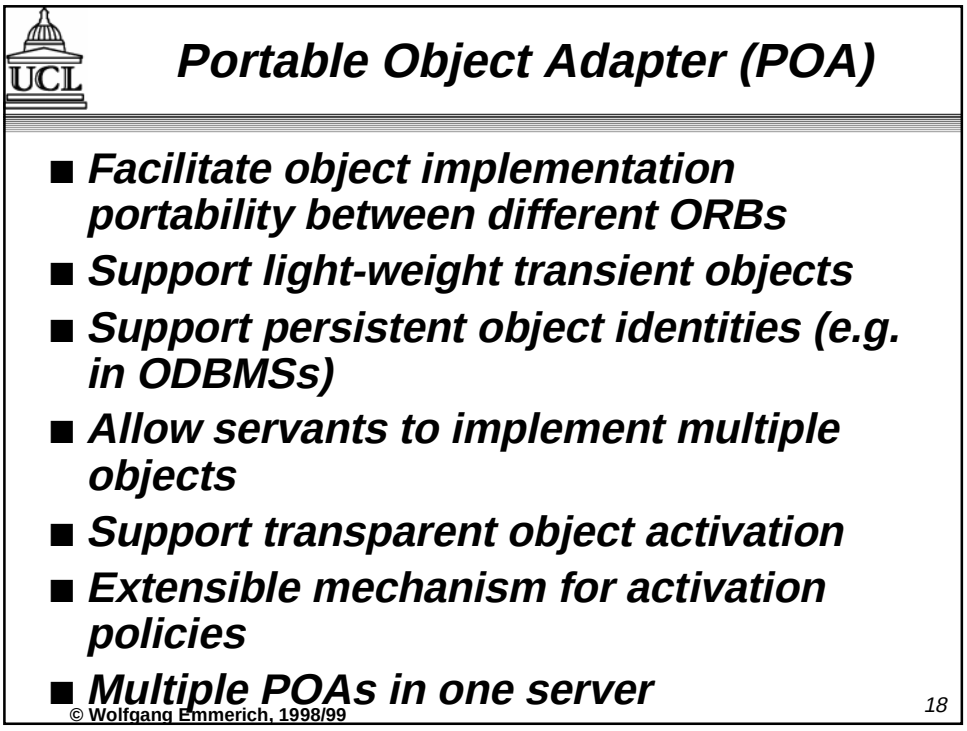
One interface per object operation

One interface per object adapter

ORB-dependent interface

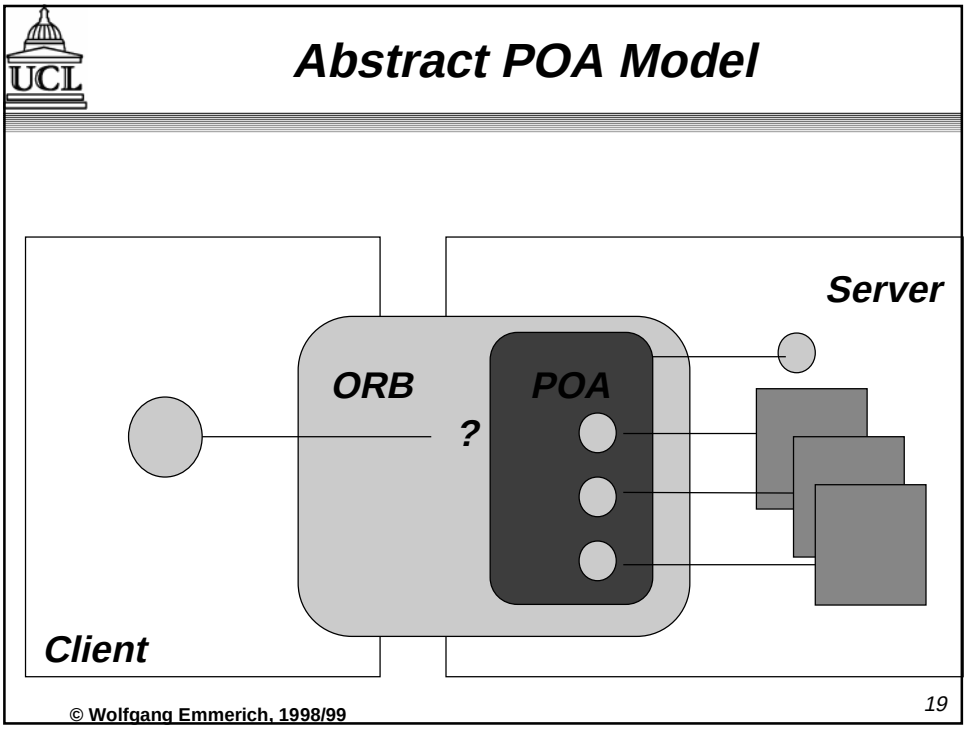


Generation of Stubs/Skeletons



Portable Object Adapter (POA)

- *Facilitate object implementation portability between different ORBs*
- *Support light-weight transient objects*
- *Support persistent object identities (e.g. in ODBMSs)*
- *Allow servants to implement multiple objects*
- *Support transparent object activation*
- *Extensible mechanism for activation policies*
- *Multiple POAs in one server*


Abstract POA Model
Server
ORB
POA
?
Client
© Wolfgang Emmerich, 1998/99
19