CORBA

- Object model
- Architecture
- IDL
- Services

- Writing CORBA code

- Reading:
  - Draft material for 3rd edition of Distributed Systems - Concepts and Design, Jean Dollimore 1997. `http://www.dcs.qmw.ac.uk/research/distrib/dsbook`
  - S. Baker "CORBA Distributed Objects (Using Orbix)", ISBN 0-201-92475-7, Addison-Wesley/ACM Press 1997.

CORBA OMG

- "Specification for object-oriented architecture for applications"
- 1989/1990: Object Management Group
  - DEC, HP, Hyperlink, NCR, Object Design, SunSoft, ….
  - `http://www.omg.org`
- 1991: *"The Common Object Request Broker: Architecture and Specification"*, V.1.1 : How to develop a CORBA implementation.
- Later updated to Version 1.2 and 2.0.

CORBA

- Metaphor: Object Request Broker (ORB)
- helps clients invoke method on an object
- locates
- activates
- communicates

- Object interfaces defined in CORBA Interface Definition Language (IDL)

- Corba *vs.* RPC:
  - interface to objects *vs* interface to servers
  - pass ROIDs as arguments or results

CORBA Object Model

- Clients send request messages to objects.
- Objects carry out methods.
- Objects are encapsulated; hidden data representation / code.

- Request message: recipient ROID, method, parameters

- Reply message: results, exceptions

- CORBA does not state how to implement remote objects (legacy code!)
  - handled by Object Adaptor

Limitations of CORBA Object Model

- CORBA does not directly support:
  - transactions
  - concurrency control
  - recovery
  - replication
  - object copying
  - caching?
- Some of this is managed in separate CORBA Services:

Event Service	Security Service	Conc. Control Service
Transaction Service	Trading Service	Persistent Object Service
Life Cycle Service	Externalization Service	Query Service
Licensing Service	Time Service	
Property Service	Relationship Service	

CORBA Architecture

- Server: process executing implementation of one or more remote objects.
- Client Stubs, Server Stubs (IDL Skeletons)
- Object Adaptor deals with everything that a client needs at run time in order to invoke a method in a remote object.
  - registers implementation in repository
  - activates object implementation in server
  - registers servers with activated objects
  - functions as ROID module (ROID creation, mapping between ROID and OID)
  - functions as dispatcher
  - Realization of Object Adaptor may be distributed.

CORBA Architecture (II)

- Object invocation:
  - e.g. server in C++:
  - skeleton is instance of a class in C++ with method for each method in IDL interface.
  - server in C?
  - what is the OID?
  - how is a method of an “object” called?
- Implementation Repository