CORBA

- Object model
- Architecture
- IDL
- Services

- Writing CORBA code

- Reading:
  - Ch 17, “Distributed Systems – Concepts and Design”, 3rd ed, Colouris, Dollimore, Kindbert

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
- 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:

<table>
<thead>
<tr>
<th>Event Service</th>
<th>Security Service</th>
<th>Conc. Control Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transaction Service</td>
<td>Trading Service</td>
<td>Persist. Object Service</td>
</tr>
<tr>
<td>Life Cycle Service</td>
<td>Externalization Service</td>
<td>Query Service</td>
</tr>
<tr>
<td>Licensing Service</td>
<td>Time Service</td>
<td></td>
</tr>
<tr>
<td>Property Service</td>
<td>Relationship Service</td>
<td></td>
</tr>
</tbody>
</table>

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 asROID module (ROID creation, mapping betweenROID 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

CORBA Services

- Set of utilities that are useful for objects or distributed applications.
- Are optional.

- Distributed systems-related services:
  - Naming Service
  - Event Service
  - Security Service
  - Trading Service

- Database-related services:
  - Concurrency Service
  - Property Service
  - Transaction Service
  - Relationship Service

- General services:
  - Life Cycle Service
  - Licensing Service
  - Time Service
  - Query Service
  - Persistent Object Service
  - Externalization Service
Example: Event Service

- Send an event that can be received by any number of objects.
- Suppliers/Consumers
- Event Channel
- Events can be values of type `any`.
- Push model:
  - `push_supplier`
  - `push()`
  - `disconnect_push_supplier()`
  - `push_consumer`
  - `disconnect_push_consumer()`

Pull model:
- `pull_supplier`
- `pull()`
- `try_pull()`
- `disconnect_pull_supplier()`
- `pull_consumer`
- `disconnect_pull_consumer()`

Event Channels