Object Model

• Object References
  • Objects can be accessed via object references

• Interfaces
  • Definition of signatures of a set of methods

• Actions
  • The state of the receiver may be changed
  • Further invocations on methods in other objects may take place

• Exceptions

• Garbage Collection
Distributed Objects

• RMI
  • Client Invocation is sent to the server managing the remote object.
  • An object may be accessed concurrently.
Distributed Object Model  1/2

• Remote Method Invocation
  • Method invocations between Object in different processes

• Remote Object References
  • Objects can be accessed via remote object references, from different processes.
  • Identifier of a unique remote object.
  • Used to invocation of remote methods
  • Can be passed as arguments and results of remote methods invocations
Distributed Object Model  2/2

• Remote Interfaces
  • Definition of signatures of a set of methods
  • Some systems (CORBA) provides Interface Definition Languages (IDL) for defining remote interfaces.
    • Classes can be implemented using different languages.
  • In Java RMI, remote interfaces are defined as any other Java Interface, but they need to extend an interface named Remote.

• Actions
  • Nested remote invocations

• Exceptions (network)

• Garbage Collection
  • Using remote references
RMI Invocation semantics 1/2

• RMI based on request-reply protocol

• Techniques that provide different delivery guaranties are:
  • Retry request message
  • Duplicate filtering
  • Retransmission of results

• Definition of **Idempotent operations**
  • An operation is *idempotent* if it can be performed repeatedly with the same effect as if it had been performed exactly once.
### RMI Invocation semantics 2/2

<table>
<thead>
<tr>
<th>Fault</th>
<th>tolerance</th>
<th>measures</th>
<th>Definition of Invocation Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retransmit request message</td>
<td>Duplicate filtering</td>
<td>Re-execute procedure or Retransmit reply</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td><strong>Maybe</strong></td>
</tr>
<tr>
<td>Yes</td>
<td>No</td>
<td>Re-execute procedure (*)</td>
<td><strong>At-least once</strong></td>
</tr>
<tr>
<td>Yes</td>
<td>Yes</td>
<td>Retransmit reply</td>
<td><strong>At-most once (</strong>)**</td>
</tr>
</tbody>
</table>

(*) unacceptable if the procedures on server are not idempotent.

(**) it is the semantic of Java RMI.