DISTRIBUTED SYSTEMS

A2.2: RPC application using distributed objects

Acu Raul-Mihai
30442

Table of Contents

1. **Project requirements**

Functional requirements:

- Users introduce the information of their cars using a simple form (Web or Desktop):
  - int year – fabrication year
  - int engineSize – engine size
  - double price- purchasing price
- The application uses RPC to send the car information to the distributed object from the server that computes the following information depending on the client request:
- The result of the invoked operation, tax, respectively selling price, is displayed on the client GUI.

Implementation technologies:

- Use the following technologies: JAVA RMI or .NET Remoting.

2. **Implementation details**

Java RMI: RMI applications often comprise two separate programs, a server and a client. A typical server program creates some remote objects, makes references to these objects accessible, and waits for clients to invoke methods on these objects. A typical client program obtains a remote reference to one or more remote objects on a server and then invokes methods on them. RMI provides the mechanism by which the server and the client communicate and pass information back and forth. Such an application is sometimes referred to as a distributed object application.

Distributed object applications need to do the following:

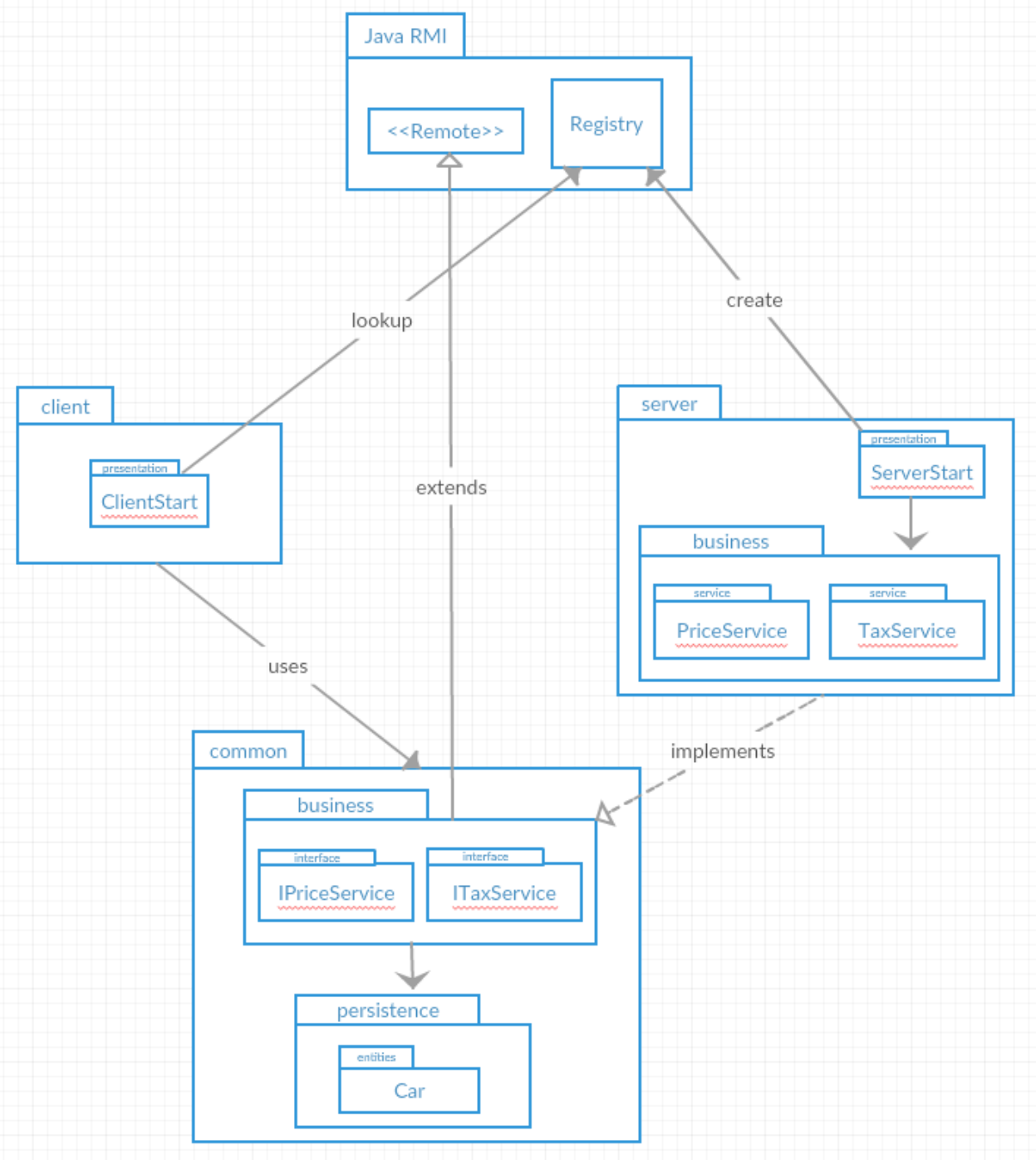
• Locate remote objects. Applications can use various mechanisms to obtain references to remote objects. For example, an application can register its remote objects with RMI's simple naming facility, the RMI registry. Alternatively, an application can pass and return remote object references as part of other remote invocations.

• Communicate with remote objects. Details of communication between remote objects are handled by RMI. To the programmer, remote communication looks similar to regular Java method invocations.

• Load class definitions for objects that are passed around. Because RMI enables objects to be passed back and forth, it provides mechanisms for loading an object's class definitions as well as for transmitting an object's data.

3. Conceptual architecture

Conceptual architecture is a form of architecture that utilizes conceptualism, characterized by an introduction of ideas or concepts from outside of architecture often as a means of expanding the discipline of architecture. This produces an essentially different kind of building than one produced by the widely held 'architect as a master-builder' model, in which craft and construction are the guiding principles



4. Deployment diagram