http://download.oracle.com/javase/tutorial/rmi/index.html

Java RMI

Java RMI …related technologies

RPC ("Remote Procedure Calls") Developed by Sun
Platform-specific

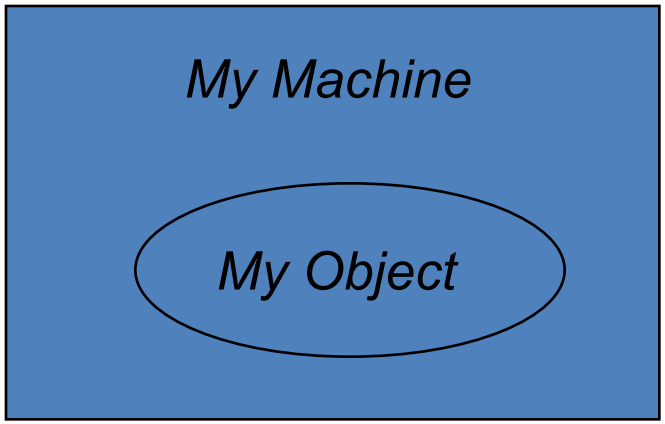
CORBA ("Common Object Request Broker Architecture")
Developed by OMG

Access to non-Java objects (as well as Java)

DCOM ("Distributed Common Object Model")
Developed by Microsoft
Access to Win32 objects
LDAP ("Lightweight Directory Access Protocol")
Finding resources on a network

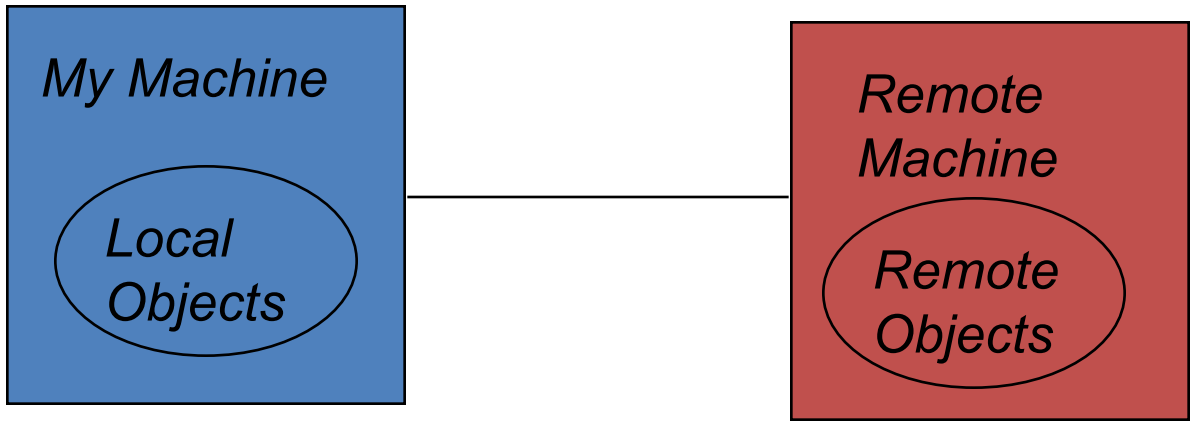
Earlier days

Only local objects existed



Today….

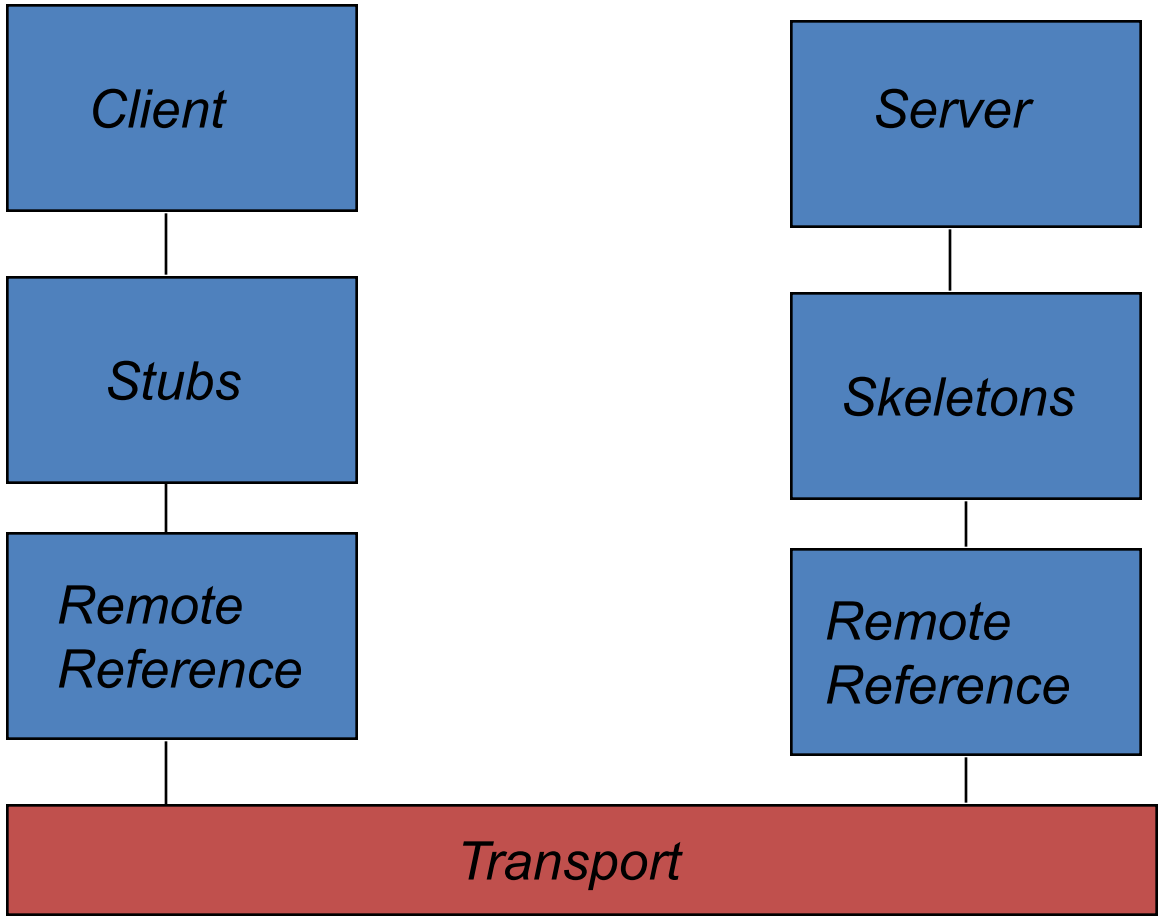
Network and Distributed Objects



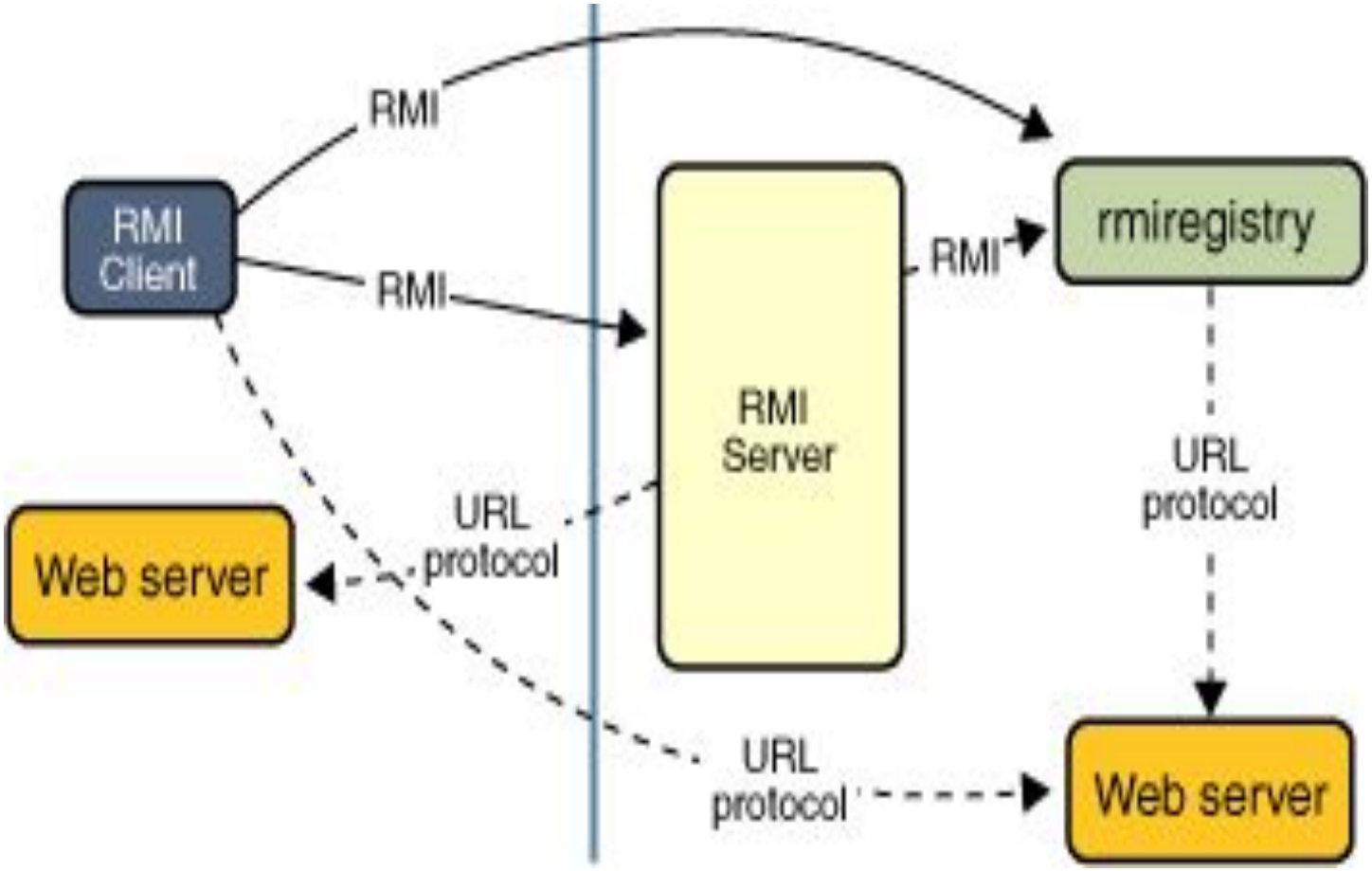
Java RMI allows...

- provide user with a “thin client “
  - allows good performance on low end workstations
- run server on high end hardware
  - maximize $ investment over many clients
  - server remote from client
- Distributed network object

Architecture



Architecture



The General Idea

- Instantiate an object on another machine
- Invoke methods on the remote object

The parts...

- Client - user interface
- Server - data source
- Stubs
  - marshals argument data (serialization)
  - unmarshals results data (deserialization)
- Skeletons (not reqd w/Java 2)
  - unmarshals argument data
  - marshals results data

The parts... (cont.)

- Remote Reference Layer
  - provides a RemoteRef object that represents the link to the remote service implementation object.
  - encodes and decodes the on-wire protocol
  - implements the remote object protocols
- Transport layer
  - The Transport Layer makes the connection between JVMs. All connections are stream-based network connections that use TCP/IP.
  - handles the underlying socket handling required for communications
    - sets up and maintains connections
    - communications related error handling

The steps...

- Create the Interface to the server
- Create the Server
- Create the Client
- Compile the Interface (javac)
- Compile the Server (javac)
- Compile the Client (javac)
- Generate Stubs and Skeletons (rmic)

To run

- Start the RMI registry
  - rmiregistry is in the JSDK bib directory
- Start the RMI Server
- Start the RMI Client

RMI Registry

- The RMI Registry is a naming service provided with the JDK as a teaching tool or for a small number of Remote Objects
- Uses port 1099 as its default port
- Can be considered to be a reference implementation
- runs out of steam above a 100 objects
- runs on same machine as the remote object

RMI Registry (more)

JNDI (Java Naming and Directory Interface)

- **J2EE (Jakarta EE) uses JNDI** as a more powerful alternative to the RMI Registry.
- **JNDI integrates with directory services** like **LDAP, CORBA Naming Service, DNS, and NIS**.
- Unlike the RMI Registry, JNDI provides:
  - **Persistent storage** of registered objects.
  - **Distributed lookup** of objects across different machines.
  - **Scalability** to support large enterprise applications.
-

Activatable Objects

- Added in Java 2 SDK
- Standard RMI objects exported as UnicastRemoteObject must run continuously
- instead by implementing java.rmi.activation.Activatable the object can be deactivated and reactivated remotely when a method call is made
- Must use the rmid server process to take advantage of this capability

Java Remote Method Protocol (JRMP)

- Proprietary, stream-based protocol that is only partially specified is now in two versions
  - JDK 1.1 version of RMI and required the use of Skeleton classes on the server
  - Java 2 SDK. It has been optimized for performance and does not require skeleton classes
- some implementations, such as BEA Weblogic and NinjaRMI don’t use JRMP instead use their own wire level protocol

Other JDK 1.1 and Java 2 Differences

- With the Java 2 SDK
  - Service interfaces are not required to extend from java.rmi.Remote
  - Service methods do not necessarily throw RemoteException.

Java 2 JSDK 1.3

- RMI-IIOP
  - instead of using JRMP RMI will use Internet Inter-Orb Protocol (IIOP)
  - IIOP is the wire protocol used for communication between Common Object Request Broker Architecture (CORBA) clients and servers
  - CORBA is a distributed object technology from the Object Management Group (OMG)
    - 800 member industry group
    - vendor neutral architecture
  - sets the direction for RMI/CORBA Integration
    - more on CORBA later