Unit-III

Remote Procedure Calls

Introduction

As IPC protocol is designed for one distribute application and does not provide a foundation on which to built a variety of distributed applications. Therefore, a need was felt for a general IPC protocol that can be used for designing several distributed applications. The Remote Procedure Call (RPC) facility emerged out of this need.IPC gained popularity because of following reasons;

1. Simple Call Syntax.
2. Familiar Semantics.
3. Its specification of a well-defined interface.
4. Its ease of use.
5. Its generality.
6. Its efficiency.
7. It can be used as an IPC mechanism to communicate between processes on different machines as well as between different processes on the same machine.

RPC Model

The RPC model is similar to the well-known and well-understood procedure call model used for the transfer of control and data within a program in the following manner;

1. To make a procedure call
2. Control transfer
3. Procedure body execution
4. Returning control

The RPC mechanism is an extension of the procedure call mechanism in the sense that it enables a call to be made to a procedure that does not reside in the address space of the calling process. The called procedure(commonly called remote procedure) may be on the same computer or on the different computer.

RPC Model...[Contd..]

Therefore the mecahnism of RPC is;

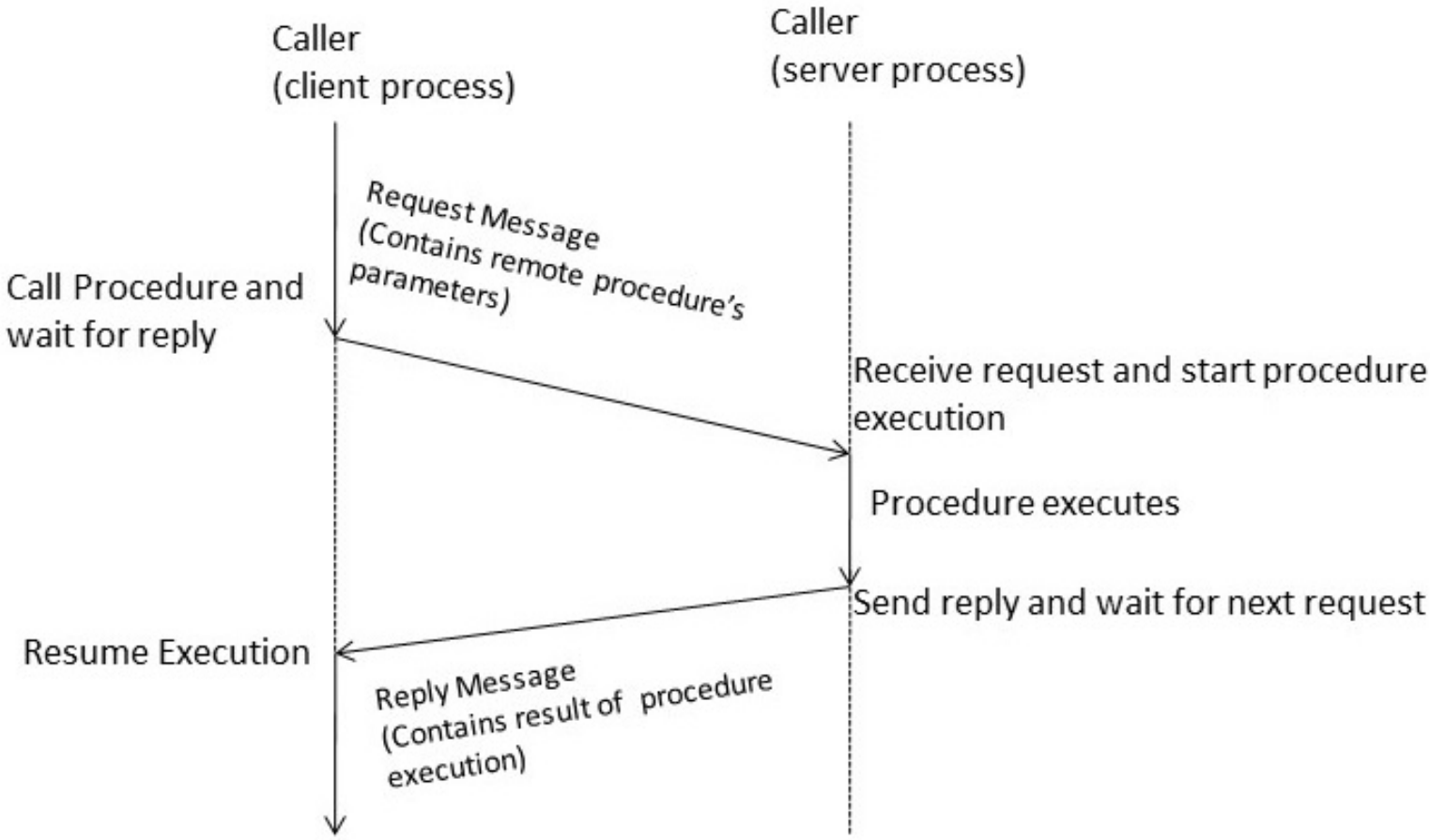

Figure: RPC Model

Transparency of RPC

A transparent RPC mechanism is one in which local procedures and remote procedures indistinguishable to the programmers. This requires the following two types of transparencies

- Syntactic Transparency
- Semantic Transparency

- Syntactic transparency are easy
- Semantic Transparency are not easy

Implementation of RPC

Implementation of RPC mechanism usually involves the following five elements of program.

1. Client
2. Client Stub
3. RPC Run time
4. Server Stub
5. Server

Implementation of RPC...[Cntd..]

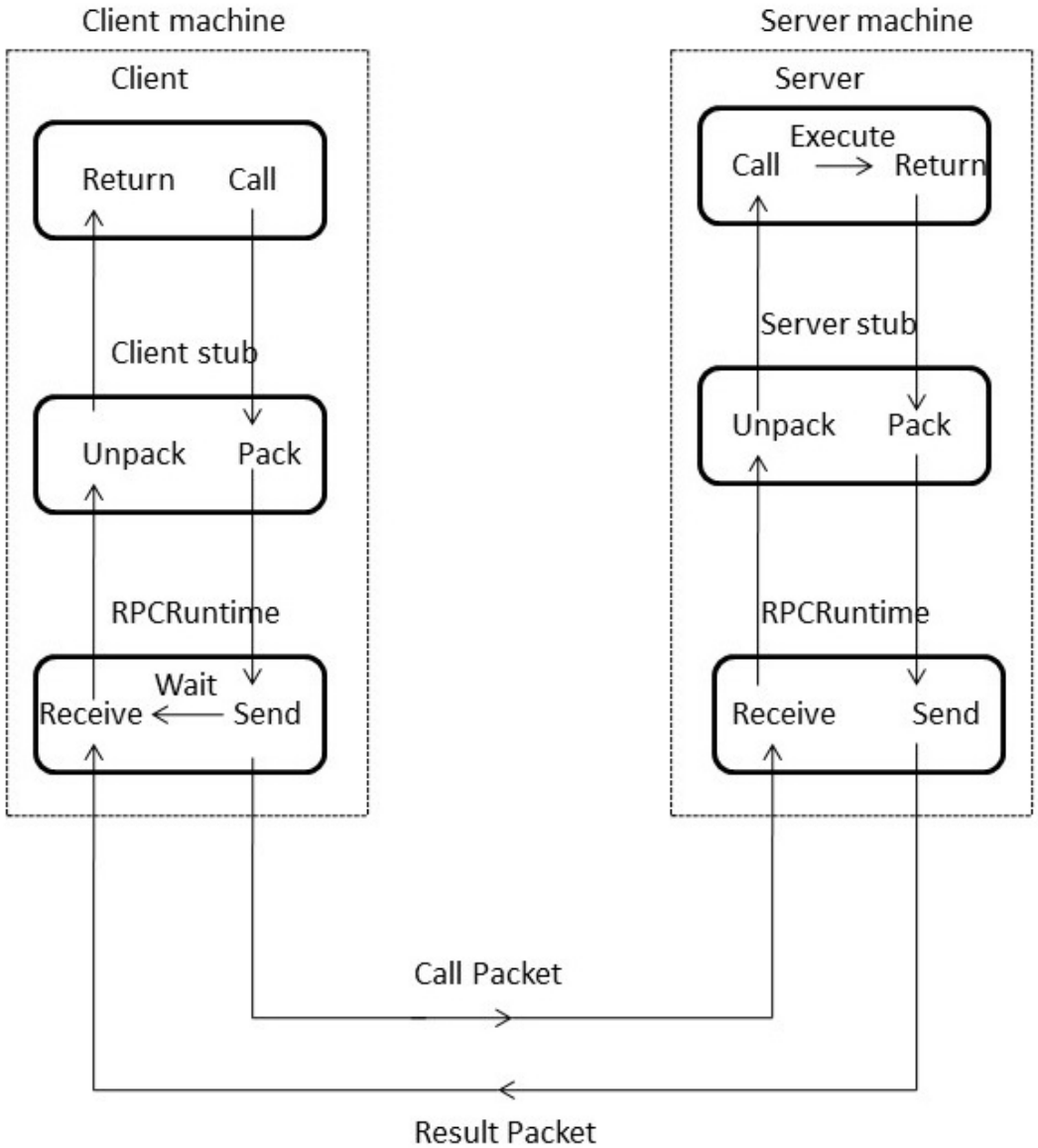

Figure: Implementation of RPC mechanism

Stub Generation

Stub can be generated in one of the following ways.

- Manually
- Automatically

RPC Messages

There are two types of messages involved in RPC implementation.

1. Call Messages
2. Reply Messages

1. Call Message

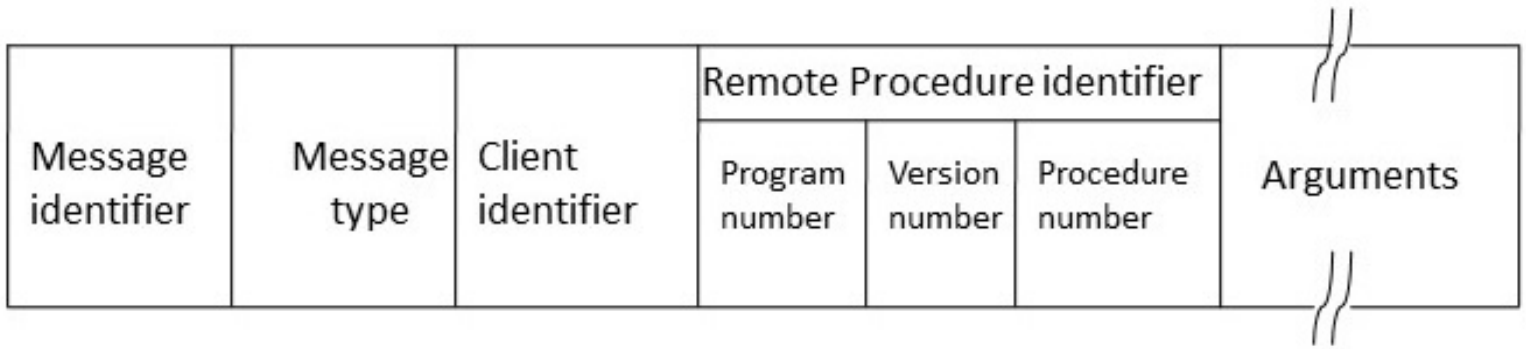

Figure: RPC Call Message Format

RPC Messages...[Cntd...]

2. Reply Message

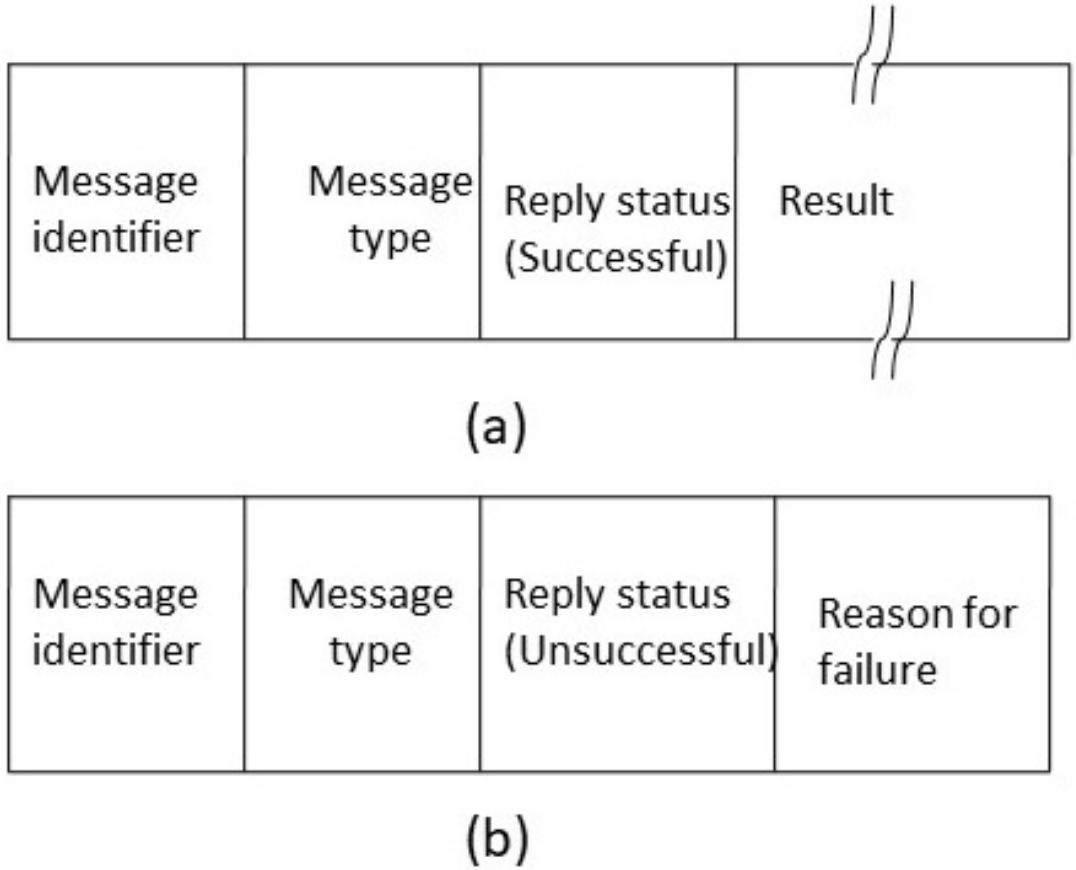

Figure: (a)A Successfule Reply Message Format (b)An Unsuccessful Message Format