

**NAME**

rpc – A simple RPC protocol for Tcl applications.

**DESCRIPTION****RPC COMMAND****rpc server** *port*

Set up an RPC server listening on port *port*. This command returns a rpc handle that be used to register Tcl procs that will be exported by a trivial RPC protocol on top of TCP. This RPC mechanism allows us to write cooperating scripts without worrying much about the communication protocol.

**rpc client** *host port*

Connect to the RPC server running on *host* with port *port*. The returned client handle can be used to invoke an exported proc on the server host.

**rpc delete** *handle*

Delete a server or a client RPC handle. All connections associated with the handle will be closed.

**rpc register** *rpchandle proc*

Export *proc* as an RPC callable proc using the server handle *rpchandle*. Only registered procs can be called by RPC clients.

**rpc unregister** *rpchandle proc*

Remove *proc* from the list of exported procs on server handle *rpchandle*.

**rpc info** [*rpchandle*]

Calling rpc info without any arguments returns a list of all rpc handles that have been created using the rpc server and rpc client command. Info about a specific rpc file handle *rpchandle* is returned by using the optional argument. The result is a list containing the tcp file handle used for the communication and the list of registered commands for a rpc handle and a list of all files that represent connections to an rpc server. This list is empty if *rpchandle* belongs to a rpc client.

**rpc#** [-async] *proc arg1 arg2 ...*

To invoke a remote procedure *proc*, use the same syntax as used with a local invocation preceeded by a client *rpchandle*. The result of the remote procedure call will be returned or an error if the underlying communication channel signals errors. The -async options can be used to invoke *proc* without waiting for it to complete. The async option will never return a result.

**SEE ALSO**

scotty(1), Tcl(n)

**AUTHORS**

Juergen Schoenwaelder <schoenw@cs.utwente.nl>