

*This is a draft specification which will be
released as a controlled document once
approved*

Resource Manager

A Remote Executive Protocol

DISTRIBUTION LIST

John Juer	
Dave Storey	
Ian Hughes	

By

M Fox

DOCUMENT TYPE

Software Design Specification

File: roc:

Contents

1	Scope	3
2	Related Documents	3
3	Introduction	3
4	Protocol	3
4.1	Execute	4
4.2	Query	5
4.3	TTY	5

REVISION HISTORY

Revision	Date	Changes
1	September 9, 1992	First formal issue

1 Scope

This document describes the protocol and programming interface of a remote executive protocol. *Audience*

2 Related Documents

[1] HP024105C301 Communications Messaging Services

3 Introduction

The protocol provides a mechanism for an executive to provide a remote interface to its services. The protocol is very simple and makes few assumptions about the executive. These assumptions are that it comprises a command line interpreter and so all commands are given as ASCII strings, and that each command results in a success code, and a possible error string.

In order to provide for a task to output its results to a remote consumer and for reading its input from a remote provider a simple teletype interface is included.

4 Protocol

The following services will be supported.

Execute Execute the command.

Query List the commands that can be executed.

TTY A teletype output message.

These services will constitute the protocol "REX".

4.1 Execute

This passes a command line to the remote executive for execution.

Its request will consist of:

Field	Type	Value(s)
Service	uint8	2 - RequestExecute
Length	uint16	Length of Command
Command	char[Length]	Command to execute

Its response will consist of:

Field	Type	Value(s)
Service	uint8	3 - ResponseExecute
Status	uint8	See below
ExitStatus	uint16	Only used if Status = 1
Handle	uint32	Execution handle returned by the executive
Length	uint16	Length of ErrorMessage
ErrorMessage	char[Length]	Message explaining error

The Values of the command status may be

Value	Description
0	Everything OK. Command successfully executed
1	Command completed
2	No such command
3	No resources to implement command at the moment

The executive may respond in one of 3 ways :

- ResponseExecute with Status > 1. The command has not been issued.
- ResponseExecute with Status = 0, and ResponseExecute with Status = 1 and the exit code returned from the command in ExitStatus. The order of these responses is not guaranteed. (This would be the case when the command is executed asynchronously with the executive).
- ResponseExecute with Status = 1 and the exit code returned from the command in ExitStatus. (This would only be the case if the command is executed synchronously with the executive).

4.2 Query

This command is used to determine what commands are supported on a node.

The query command may be used in 3 modes

- Query the existence of a command. This may be used to determine if a command is supported or what a command abbreviation resolves to. (eg “dir” or “DIR” or “DIRECTORY” = “directory”)
- List all supported commands.
- Obtain the remainder of a list of commands that could not complete due to lack of resources (eg the listing is too long to fit in one message or it would take too long to list in one go).

The request will consist of:

Field	Type	Value(s)
Service	uint8	4 - RequestQuery
Length	uint16	Length of Command
Command	char[Length]	Command to query
Continue	bool	FALSE - Implies command is the command to query TRUE - Implies continue listing after command

If no command to query is specified (ie Length = 0) and Continue = FALSE then all commands are to be listed in reply.

Its response will consist of:

Field	Type	Value(s)
Service	uint8	5 - ResponseQuery
NumCommands	uint16	Number of commands listed
Commands	Length - uint16 Command - char[Length]	Array of commands Length of Command Command name
MoreFollows	bool	TRUE - if further RequestQuery need to complete listing

4.3 TTY

The teletype service is simple request without response which transports a string without any meaning.

The request will consist of:

Field	Type	Value(s)
Service	uint8	6 - RequestTTY
Length	uint16	Length of String
String	byte[Length]	string as input

Conventions (Section)