

Calculating Size Requirements

The command `YSRPC CSMASS` is used to calculate the buffer size stubless RPC calls require for sending data from the client to the server or vice versa. The indication of the size helps you configure the middleware layer used; for example, the Broker attribute file when EntireX Broker is used.

If desired, you can also perform size calculations for stub subprograms (interface objects), even though sizes are already calculated when generating the stub subprograms.

`YSRPC CSMASS` can be used in online or batch mode.

This section contains information on:

- Using the `YSRPC CSMASS` Command
 - Name Specification and Compression
-

Using the `YSRPC CSMASS` Command

You can enter the command `YSRPC CSMASS` at any `NEXT` or `MORE` command prompt for calculating size requirements online.

The section below contains information on:

Syntax of `YSRPC CSMASS`

The syntax that applies to the command `YSRPC CSMASS` is illustrated in the diagram below:

```
YSRPC CSMASS [name] [compression]
```

The syntactical items *name* and *compression* are explained in the section *Name Specification and Compression*.

`YSRPC CSMASS` Report

The command `YSRPC CSMASS` produces a report that indicates the send and receive length requirements of the subprograms (objects) specified with the command as shown in the following example:

```

Page          1                               2006-05-24 15:54:12

      SYSRPC - Calculation of Buffer Sizes for Stubless RPC in Library SAGTEST

Calculation Criteria:

      Object name or range: RPC*
      Compression: 1

Calculation Results:

      Number of objects found:          8
      Maximum send length:  200228
      Maximum receive length: 1024192

Object      Type      Send Length  Receive Length  Message
-----
RPCCALL1   N              209           202
RPCCALL2   N              219           240  Compression=2
RPCCALL3   N              204           193
MORE

```

The report is organized in three sections, which contain the following information:

- **Calculation Criteria:**

The criteria based on which the calculation was made: a single object name or a range of names (here: RPC*) and the compression (here: 1).

- **Calculation Results:**

The number of objects selected for the size calculation.

The maximum buffer sizes all selected objects require for sending and receiving data from the client.

- **Object List:**

The name and type (here: N for type subprogram) of each object selected for the calculation. The buffer sizes each object requires for sending (**Send Length**) and receiving (**Receive Length**) data from the client. A possible comment on each object calculation in the **Message** column. In the example above, **Compression=2** indicates that object RPCCALL2 was not calculated with Compression 1 as requested in the command. The object list is sorted in alphabetical order of object names.

If the MORE prompt appears, choose ENTER to scroll to the end of the report.

If the size calculation fails for single or multiple objects, the report shows the number of objects affected and appropriate error messages.

Name Specification and Compression

You can specify the objects (subprograms) to be selected for size calculation and the type of compression to be used:

- Name
- Compression

Name

You can specify an object name or a range of names. If you do not specify a name or a range of names, the size of all subprograms contained in the current library will be calculated.

Valid name specifications are described below where *value* is any combination of one or more alphanumeric characters:

Input	Objects Selected
*	All subprograms. This is the default setting.
<i>value</i>	A subprogram with a name equal to <i>value</i> .
<i>value</i> *	All subprograms with names that start with <i>value</i> .
<i>value</i> <	All subprograms with names less than or equal to <i>value</i>
<i>value</i> >	All subprograms with names greater than or equal to <i>value</i> .

Compression

You can specify any of the following compression types: 0, 1, 2 . The specification of compression is optional. The default type used for stub generation is 1.

See also *Using Compression* described in *Operating a Natural RPC Environment* in the *Natural Remote Procedure Call (RPC)* documentation.