

Using Conversational RPC

This chapter explains how clients built with the COBOL Wrapper use conversational RPC.

RPC conversations are supported when communicating with an RPC server. It is further assumed that you are familiar with the concepts of conversational RPC and non-conversational RPC. To use conversational RPC, you need the following components:

- the *Generic RPC Services Modules* are provided to open, close or abort conversations;
- the *The RPC Communication Area (Reference)*

➤ To use conversational RPC

1. Open a conversation with the function Open Conversation OC provided by the generic RPC services module.

In the scenarios *Micro Focus*, *Batch CICS* and *IMS* with the *Call Interface*:

```
MOVE "2000" TO COMM-VERSION.
MOVE "OC" TO COMM-FUNCTION.
CALL "COBSRVI" USING ERX-COMMUNICATION-AREA
ON EXCEPTION
. . .
NOT ON EXCEPTION
. . .
END-CALL.
```

Or:

In the scenario *Using the COBOL Wrapper for CICS with DFHCOMMAREA Calling Convention (z/OS and z/VSE)* with the *EXEC CICS LINK Interface*:

```
MOVE "2000" TO COMM-VERSION.
MOVE "OC" TO COMM-FUNCTION.
EXEC CICS LINK PROGRAM ("COBSRVI")
                RESP (CICS-RESP1)
                RESP2 (CICS-RESP2)
                COMMAREA (ERX-COMMUNICATION-AREA)
                LENGTH (LENGTH OF ERX-COMMUNICATION-AREA)
END-EXEC.
IF WORKRESP = DFHRESP(NORMAL)
    IF (COMM-RETURN-CODE = 0) THEN
*       Perform success-handling
    ELSE
*       Perform error-handling
    END-IF
ELSE
*       Perform error-handling
END-IF.
```

2. Issue your RPC requests as within non-conversational mode using the generated client interface objects. Different client interface objects can participate in the same RPC conversation.

➤ To abort conversational RPC communication

- Abort an unsuccessful RPC conversation with the function Close Conversation CB provided by the generic RPC services module

In the scenarios *Micro Focus, Batch, CICS and IMS with the Call Interface*:

```
MOVE "2000" TO COMM-VERSION.
MOVE "CB" TO COMM-FUNCTION.
CALL "COBSRVI" USING ERX-COMMUNICATION-AREA
ON EXCEPTION
. . .
NOT ON EXCEPTION
. . .
END-CALL.
```

Or:

In the scenario *Using the COBOL Wrapper for CICS with DFHCOMMAREA Calling Convention (z/OS and z/VSE) with the EXEC CICS LINK Interface*:

```
MOVE "2000" TO COMM-VERSION.
MOVE "CB" TO COMM-FUNCTION.
EXEC CICS LINK PROGRAM ("COBSRVI")
                RESP    (CICS-RESP1)
                RESP2   (CICS-RESP2)
                COMMAREA (ERX-COMMUNICATION-AREA)
                LENGTH   (LENGTH OF ERX-COMMUNICATION-AREA)

END-EXEC.
IF WORKRESP = DFHRESP(NORMAL)
  IF (COMM-RETURN-CODE = 0) THEN
*   Perform success-handling
  ELSE
*   Perform error-handling
  END-IF
ELSE
*   Perform error-handling
  END-IF.
```

➤ To close and commit a conversational RPC communication

- Close the RPC conversation successfully with the function Close Conversation and Commit CE provided by the generic RPC services module

In the scenarios *Micro Focus, Batch, CICS and IMS with the Call Interface*:

```
MOVE "2000" TO COMM-VERSION.
MOVE "CE" TO COMM-FUNCTION.
CALL "COBSRVI" USING ERX-COMMUNICATION-AREA
ON EXCEPTION
. . .
NOT ON EXCEPTION
. . .
END-CALL.
```

Or:

In the scenario *Using the COBOL Wrapper for CICS with DFHCOMMAREA Calling Convention (z/OS and z/VSE) with the EXEC CICS LINK Interface*:

```
MOVE "2000" TO COMM-VERSION.
MOVE "CE" TO COMM-FUNCTION.
EXEC CICS LINK PROGRAM ("COBSRVI")
      RESP (CICS-RESP1)
      RESP2 (CICS-RESP2)
      COMMAREA (ERX-COMMUNICATION-AREA)
      LENGTH (LENGTH OF ERX-COMMUNICATION-AREA)
END-EXEC.
IF WORKRESP = DFHRESP(NORMAL)
  IF (COMM-RETURN-CODE = 0) THEN
*   Perform success-handling
  ELSE
*   Perform error-handling
  END-IF
ELSE
*   Perform error-handling
END-IF.
```