

---

# SPICE

Span Integrated Checkpoint/Restart Environment

---

---

## SPICE SQL™ & SPICE DL/I™ Getting Started

Release 3.1

SPI 14 12

---

### Span Software Consultants Limited

Little Moss, Peacock Lane  
High Legh  
Knutsford  
Cheshire  
WA16 6PL  
UNITED KINGDOM

Telephone: +44 (0) 1565 832999  
Fax: +44 (0) 1565 830653  
email [spice@spansoftware.com](mailto:spice@spansoftware.com)  
website <http://www.spansoftware.com>

---

© Copyright 2005 Span Software Consultants Limited.  
All rights reserved. No part of this publication may be re-produced,  
stored in a retrieval system or transmitted, in any form or by any  
means, electronic, mechanical, or otherwise, without the prior written  
consent of the publisher.

---

24 March 2006



---

# Preface

## Abstract

SPICE is an acronym for Span Integrated Checkpoint/Restart Environment.

SPICE SQL™ and SPICE DL/I™ are software products that simplify the design, implementation and operation of restartable batch application programs in the DB2 and IMS environment, respectively.

This manual documents how the SPICE software products should be installed. The reader should be familiar with z/OS SMP/E.

## Other SPICE Manuals

### ***SPI 08 SPICE SQL™ Product Reference Manual***

This manual is the principal reference for SPICE SQL, the SPICE product for the DB2 environment.

### ***SPI 09 SPICE DL/I™ Product Reference Manual***

This manual is the principal reference for SPICE DL/I, the SPICE product for the IMS environment.

### ***SPI 10 SPICE SQL™ & SPICE DL/I™ Diagnostics Reference Manual***

This manual is the reference for the messages and other diagnostic information issued by the SPICE products.

## Note:

SPICE, SPICE SQL, SPICE DL/I and In-Flight Restart are trademarks of Span Software Consultants Limited.

IBM, DB2, IMS, CICS, z/OS and OS/390 are trademarks of the International Business Machines Corporation.



---

# Contents

<b>Chapter 1. Introduction</b> .....	<b><u>1</u></b>
<b>Using This Manual</b> .....	<b><u>1</u></b>
<b>System Requirements</b> .....	<b><u>1</u></b>
<b>Distribution</b> .....	<b><u>2</u></b>
<b>Chapter 2. SPICE Installation</b> .....	<b><u>3</u></b>
<b>Transfer Product to z/OS</b> .....	<b><u>3</u></b>
Unload Distributed Material .....	<b><u>3</u></b>
Unpack Installation Material .....	<b><u>3</u></b>
<b>Customize Installation Libraries</b> .....	<b><u>4</u></b>
Installation Job Stream Customization .....	<b><u>4</u></b>
Customize the Installation Library .....	<b><u>4</u></b>
Unpack the Installation Libraries .....	<b><u>5</u></b>
<b>SMP/E Installation of SPICE</b> .....	<b><u>6</u></b>
Install SPICE SMP/E FMIDs .....	<b><u>6</u></b>
Incorporate DB2 HLI into SPICE Run-time Libraries .....	<b><u>7</u></b>
Install SPICE SMP/E Service .....	<b><u>7</u></b>
<b>Installation Considerations</b> .....	<b><u>8</u></b>
z/OS Language Environment (LE) Considerations .....	<b><u>8</u></b>
SPICE Security .....	<b><u>8</u></b>
<b>Chapter 3. SPICE System Verification</b> .....	<b><u>9</u></b>
<b>SPICE SQL Database Installation and Verification</b> .....	<b><u>9</u></b>
SPICE SQL Restart Database Creation .....	<b><u>9</u></b>
<b>SPICE DL/I Database Installation and Verification</b> .....	<b><u>10</u></b>
IMS Preparation .....	<b><u>10</u></b>
SPICE DL/I Restart Database Creation .....	<b><u>11</u></b>
<b>Chapter 4. SPICE Operator Subsystem Administration</b> .....	<b><u>13</u></b>
<b>SPICE Operator Subsystem Installation</b> .....	<b><u>13</u></b>
ISPF Installation .....	<b><u>13</u></b>
Verify Installation of the SPICE Operator Subsystem .....	<b><u>14</u></b>
DB2 & IMS System Definition .....	<b><u>15</u></b>
SPICE Operator Subsystem DB2 System Definition .....	<b><u>16</u></b>
SPICE Operator Subsystem IMS System Definition .....	<b><u>17</u></b>
Select SPICE Subsystem Table Entry .....	<b><u>18</u></b>
Deploy the SPICE Operator Subsystem .....	<b><u>19</u></b>
SPICE Operator Subsystem ISPF Environment Requirements .....	<b><u>19</u></b>
SPICE Operator Subsystem ISPF Panel Requirements .....	<b><u>20</u></b>

SPICE Operator Subsystem Security .....	<a href="#">20</a>
<b>Chapter 5. SPICE Sample Applications .....</b>	<b><a href="#">21</a></b>
<b>SPICE SQL Sample Applications .....</b>	<b><a href="#">21</a></b>
SPICE SQL Sample Application Installation .....	<a href="#">22</a>
Installation Verification (SQL statements) .....	<a href="#">22</a>
Installation Verification (SPICE Restart API) .....	<a href="#">23</a>
<b>SPICE DL/I Sample Application .....</b>	<b><a href="#">24</a></b>
IMS System Preparation .....	<a href="#">24</a>
SPICE DL/I Sample Application Installation .....	<a href="#">25</a>
Installation Verification .....	<a href="#">25</a>
<b>Chapter 6. SPICE Utility Security Exit .....</b>	<b><a href="#">27</a></b>
Introduction .....	<a href="#">27</a>
Processing Overview .....	<a href="#">27</a>
Installation .....	<a href="#">28</a>
Sample Exit .....	<a href="#">28</a>
<b>Appendix A. SPICE Data Sets .....</b>	<b><a href="#">29</a></b>
Distributed Material .....	<a href="#">29</a>
SMP/E Distribution Libraries .....	<a href="#">31</a>
SPICE Run-Time Libraries .....	<a href="#">32</a>

---

# Chapter 1. Introduction

---

## Using This Manual

This manual describes how the SPICE products are installed.

This chapter, [Chapter 1. Introduction](#) on page [1](#), documents the environments required by the SPICE products, and how the products are distributed.

[Chapter 2. SPICE Installation](#) on page [3](#) describes how you should install the SPICE products libraries onto the target system.

[Chapter 3. SPICE System Verification](#) on page [?](#) describes how you should verify that the SPICE products have been installed correctly.

[Chapter 4. SPICE Operator Subsystem Administration](#) on page [13](#) describes how you configure the SPICE Operator Subsystem.

[Chapter 5. SPICE Sample Applications](#) on page [21](#) describes how you install and execute the SPICE sample applications.

[Chapter 6. SPICE Utility Security Exit](#) on page [27](#) describes the optional SPICE Security Exit.

[Appendix A. SPICE Data Sets](#) on page [29](#) details the data sets created on your system by SPICE installation.

---

## System Requirements

All SPICE products will execute under any release of z/OS and OS/390 supported by IBM.

SPICE SQL will execute under any release of DB2 supported by IBM.

SPICE DL/I will execute under any release of IMS supported by IBM.

For application environments that execute in z/OS Language Environment (LE) enclaves, the reader's attention is drawn to item [z/OS Language Environment \(LE\) Considerations](#) on page [8](#).

---

# Distribution

The SPICE products are supplied on CD-Rom. Full instructions on how to unload the product(s) are included on the disk.

---

# Chapter 2. SPICE Installation

This chapter shows how to build the libraries used to install the SPICE products.

- [Transfer Product to z/OS](#) on page [3](#)
- [Customize Installation Libraries](#) on page [4](#)
- [SMP/E Installation of SPICE](#) on page [6](#)
- [Installation Considerations](#) on page [8](#)

---

## Transfer Product to z/OS

---

### Unload Distributed Material

Follow the instructions supplied with the SPICE product package to transfer the distributed materials onto the target z/OS system.

---

### Unpack Installation Material

The distributed material includes a job, UNPAK. It unpacks the data sets required for the installation of the SPICE product(s) from the distributed materials. As supplied, the job creates the installation libraries with the following names:

<i>Library</i>	<i>Notes</i>
SPICE.SPI31.INSTALL	Installation JCL
SPICE.SPI31.INSTALL	Installation REXX Execs
SPICE.SPI31.INSTALL	Installation Verification material

Table 2.1: Installation Libraries

To run it, perform the following:

<i>Installation Library Member Name</i>	<i>Compo- nent</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Customize job 'UNPAK' to local requirements <ul style="list-style-type: none"> <li>• The supplied job creates data set names beginning 'SPICE.SPI31' on 3390 volume SPIVOL</li> <li>• The appropriate statements for customization are flagged '/* LOCAL */'.</li> </ul>
			Submit 'UNPAK'
UNPAK	all	none	Unpack installation libraries

Table 2.2: Unpack Installation Libraries

---

# Customize Installation Libraries

---

## Installation Job Stream Customization

The supplied installation jobs are built to a naming convention that may not conform to your local standards. SPICE installation caters for this with its customization process.

### Customize the Installation Library

SPICE includes an installation ISPF Edit Macro REXX Exec 'CUSTOMIZ'. Job 'I02#CUST' applies it against all the members of the SPICE installation library. By editing the REXX Exec appropriately before submitting the job, you can customize the Installation Library to your requirements. Read the comments within 'CUSTOMIZ' for details on how you can extend it to further customize the installation library.

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Edit REXX Exec library 'SPICE.SPI31.INSTLIB' member 'CUSTOMIZ', to produce JCL that conforms to your installation <ul style="list-style-type: none"><li>Use the guidelines embedded as comments</li></ul>
			Submit the following installation library members:
I02#CUST	all	none	<ul style="list-style-type: none"><li>Customize the installation library</li></ul>

Table 2.3: Customize the Installation Library

Check that the customization process has generated JCL as expected. If you find that it has not, you should re-run the 'UNPAK' job (see [Unpack Installation Material](#) on page 3) before re-running 'I02#CUST'. Note, however, that this job will overwrite all the changes that you have made to 'CUSTOMIZ', so you may wish to backup the REXX Exec first.

### Unpack the Installation Libraries

Submit one of the following jobs, according to which product(s) you are installing:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Submit one of the following installation library members:
I03DUNPS	DL/I only	none	<ul style="list-style-type: none"><li>Unpack SPICE Base and SPICE DL/I libraries</li></ul>
I03RUNPS	SQL only	none	<ul style="list-style-type: none"><li>Unpack SPICE Base and SPICE SQL libraries</li></ul>
I03XUNPS	both DL/I & SQL	none	<ul style="list-style-type: none"><li>Unpack SPICE Base, SPICE DL/I and SPICE SQL libraries</li></ul>

Table 2.4: Unpack the Installation Libraries

# SMP/E Installation of SPICE

## Install SPICE SMP/E FMIDs

The SPICE products are installed with SMP/E. The SPICE function modification identifiers are of the form VSPxy00, where p indicates the SPICE component and xy the SPICE release. For instance, VSN3100 is the FMID for the SPICE base product and VSR3100 is the FMID for SPICE SQL, both for release 03.1. SPICE PTF sysmod identifiers are prefixed VP0, and APARs are prefixed VA0, VF0 and VZ0.

Submit one of the following jobs, according to which product(s) you are installing:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
Submit one of the following installation library members:			
I11DSMPI	DL/I only	4 on step SPIUCL	• For SPICE DL/I, prepare the SMP/E CSI's (SPICE SQL <b>not</b> being installed)
I11RSMPI	SQL only	4 on step SPIUCL	• For SPICE SQL, prepare the SMP/E CSI's (SPICE DL/I <b>not</b> being installed)
I11XSMPI	both SQL & DL/I	4 on step SPIUCL	• For SPICE SQL <b>and</b> SPICE DL/I, prepare the SMP/E CSI's.

Table 2.5: Setup SMP/E environment

Submit one of the following jobs, according to which product(s) you are installing:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
Submit one of the following installation library members:			
I12DSMPR	DL/I only	4 on any step	• For SPICE DL/I, receive, apply and accept the SMP/E FMIDs (SPICE SQL <b>not</b> being installed)
I12RSMRP	SQL only	4 on any step	• For SPICE SQL, receive, apply and accept the SMP/E FMIDs (SPICE DL/I <b>not</b> being installed)
I12XSMRP	both SQL & DL/I	4 on any step	• For SPICE SQL <b>and</b> SPICE DL/I, receive, apply and accept the SMP/E FMIDs

Table 2.6: Install SPICE SMP/E FMIDs

---

## Incorporate DB2 HLI into SPICE Run-time Libraries

SPICE SQL requires that the DB2 High Level Interface is included in certain of its load modules. For SPICE SQL installation, submit the following members of the installation library:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
Submit the following installation library members:			
I13RLELI	SQL or SQL & DL/I	none	• Re-link SPICE to include the DB2 HLI
I14RSMPJ	SQL or SQL & DL/I	none	• Perform SMP/E JCL, to include the DB2 HLI in SMP/E link edits of SPICE SQL modules

Table 2.7: Incorporate DB2 HLI into SPICE Run-time Libraries

---

## Install SPICE SMP/E Service

The 'readme' file provided with SPICE will contain details of the service that should be applied to the products. Obtain up to date service information from our web site (<http://www.spansoftware.com>). This will provide details of further SMP/E Sysmods that we recommend you install on your system.

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
Download recommended service			
Edit installation job 'I15#SMPS', to receive and apply the recommended service			
Submit the following installation library members:			
I15#SMPS	all	none	• Receive and apply SMP/E Sysmods

Table 2.8: Install SPICE SMP/E Service

---

# Installation Considerations

---

## **z/OS Language Environment (LE) Considerations**

The successful execution of SPICE applications requires that SPICE be informed of application termination, normal or abnormal. This enables SPICE to indicate the application's successful completion in the SPICE Restart Database. Failure to so do can result in application failure.

In z/OS Language Environment Enclaves, this is performed via the SPICE implementation of the z/OS Language Environment Assembler exit, CEEBXITA. The exit resides in the SPICE load library, 'SPICE.SPI31.SPILOAD'. For SPICE applications that execute in LE enclaves, this data set should be the first load library containing CEEBXITA in the STEPLIB DD concatenation.

---

## **SPICE Security**

By default, SPICE relies for security on customer control of access to the SPICE libraries and the SPICE Restart Databases. Should further security be required, the SPICE utility security exit, detailed in [Chapter 6. SPICE Utility Security Exit](#) on page [27](#) of this manual, can be used to control access to the commands of the SPICE Operator Subsystem and the SPICE Utility program.

---

## Chapter 3. SPICE System Verification

This chapter details the tasks required to verify that the installation of the SPICE was successful. For each product, we create a SPICE Restart Database and run the SPICE Utility program against it.

[\*SPICE SQL Database Installation and Verification\*](#) on page [9](#)

[\*SPICE DL/I Database Installation and Verification\*](#) on page [10](#)

---

### SPICE SQL Database Installation and Verification

---

#### SPICE SQL Restart Database Creation

The following installation library jobs create a SPICE SQL Installation Verification Restart DB2 Database, and verify the SPICE SQL software installation by executing the SPICE utility program.

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Prepare installation job I31RSDBI <ul style="list-style-type: none"> <li>The steps flagged &lt;CLEANUP&gt;, &lt;STOGROUP&gt; and &lt;DATABASE&gt; administer DB2 storage groups and databases               <ul style="list-style-type: none"> <li>Your installation may require that your DB2 administrator perform them</li> </ul> </li> <li>You can suppress such steps by changing their program names from <i>IKJEFT01</i> to <i>IEFBR14</i></li> </ul>
			Submit the following installation library members:
I31RSDBI	SQL	8 on steps DROPSPi and CLEANUP, 4 on step GRANT	<ul style="list-style-type: none"> <li>For the installation verification SPICE DL/I Restart Database:               <ul style="list-style-type: none"> <li>Create DB2 database and storage groups</li> <li>Create DB2 tables</li> <li>Bind SPICE SQL DBRM into the SPICE DB2 Package</li> <li>Bind the SPICE Utility program DB2 Plan</li> <li>Grant DB2 access rights and initialise the database</li> </ul> </li> </ul>
I32RSDBV	SQL	4 on any step	<ul style="list-style-type: none"> <li>Execute the SPICE Utility program in the DB2 environment.</li> </ul>

Table 3.1: Verify SPICE SQL System Installation

---

## SPICE DL/I Database Installation and Verification

---

### IMS Preparation

The SPICE DL/I Installation Verification Restart Database must be defined to the IMS system. Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Include the following database definition statements in your IMS system definition: <pre style="text-align: center;">DATABASE DBD=SPICHSAM</pre>
			Perform an IMS MODBLKS generation.
			Re-start your IMS system, or perform IMS On-Line Change.

Table 3.2: IMS System Preparation

## SPICE DL/I Restart Database Creation

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Submit the following installation library members:
I41DSDBI	DL/I	4 on step ACBGEN and 12 on DBRCDEL	<ul style="list-style-type: none"> <li>For the installation verification SPICE DL/I Restart Database:               <ul style="list-style-type: none"> <li>Generate DBD, PSB's and ACB's</li> <li>Create and initialise</li> <li>Define IMS Dynamic Allocation control blocks</li> <li>Register it with DBRC</li> <li>Perform IMS On-Line Change</li> </ul> </li> </ul>
			Issue the following commands to IMS, to implement the prepared IMS On-Line Change:  /MODIFY PREPARE ACBLIB /MODIFY COMMIT
			Submit the following installation library members:
I42DSDBV	DL/I	4 on any step	<ul style="list-style-type: none"> <li>Execute the SPICE Utility program in the IMS batch environment</li> </ul>
I43DSDBV	DL/I	4 on any step	<ul style="list-style-type: none"> <li>Execute the SPICE Utility program in the IMS BMP environment</li> </ul>

Table 3.3: Create SPICE DL/I Verification Restart Database



---

# Chapter 4. SPICE Operator Subsystem Administration

---

## SPICE Operator Subsystem Installation

The user can administer SPICE either with z/OS batch jobs, or with a set of ISPF panels, the SPICE Operator Subsystem.

---

### ISPF Installation

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
Submit one of the following jobs:			
I21DSOSX	DL/I only	none	• Install SPICE Operator Subsystem startup REXX Execs
I21RSOSX	SQL only	none	• Install SPICE Operator Subsystem startup REXX Execs
I21XSOSX	both DL/I and SQL	none	• Install SPICE Operator Subsystem startup REXX Execs

Table 4.1: Install SPICE Operator Subsystem REXX Execs

The REXX Exec library 'SPICE.SPI31SYS.EXEC' will now contain the following members, appropriate to the SPICE products installed:

<i>Library</i>	<i>Notes</i>
SPIISPFD	ISPF wrapper for the SPICE Operator Subsystem <ul style="list-style-type: none"><li>• This allocates the libraries required by IMS using the TSO 'TSOLIB ACTIVATE' command before invoking ISPF</li></ul>
SPIISPFR	ISPF wrapper for the SPICE Operator Subsystem <ul style="list-style-type: none"><li>• This allocates the libraries required by DB2 using the TSO 'TSOLIB ACTIVATE' command before invoking ISPF</li></ul>
SPIISPFX	ISPF wrapper for the SPICE Operator Subsystem <ul style="list-style-type: none"><li>• This allocates the libraries required by IMS and DB2 using the TSO 'TSOLIB ACTIVATE' command before invoking ISPF</li></ul>
SPIOPS	ISPF REXX Exec to initiate the SPICE Operator Subsystem

Table 4.2: SPICE Operator Subsystem REXX Execs

---

## Verify Installation of the SPICE Operator Subsystem

To verify the correct installation of the SPICE Operator Subsystem, perform the following actions:

- 1) From TSO (i.e. *not* from ISPF), issue the one of following commands:

```
READY
exec 'SPICE.SPI31SYS.EXEC(SPIISPF)'
```

Figure 4.1: Start ISPF (SPICE DL/I)

```
READY
exec 'SPICE.SPI31SYS.EXEC(SPIISPF)'
```

Figure 4.2: Start ISPF (SPICE SQL)

```
READY
exec 'SPICE.SPI31SYS.EXEC(SPIISPF)'
```

Figure 4.3: Start ISPF (SPICE SQL & DL/I)

- 2) From the Dialog Test Functions panel of ISPF (option 7.1), invoke REXX Exec SPIOPS, as illustrated by the following:

```
Menu Save Utilities Help
-----
Invoke Dialog Function/Selection Panel
Command ==>
Invoke selection panel:
PANEL . .
OPT . . .
More: +
Invoke command:
CMD . . . exec 'SPICE.SPI31SYS.EXEC(SPIOPS)'
LANG . . . (APL, CREX, or blank)
MODE . . . (LINE, FSCR, or blank)
Invoke program:
PGM . . .
PARM . . .
```

Figure 4.4: SPICE Operator Subsystem Menu



To add a new entry to the table, select line command I (Insert) against any entry in the table. The following panel will be displayed:

```

===== SPICE Operator Subsystem =====
OPTION ==> _
----- SPICE System Entry Edit -----

System name      ==>                Type ==>                (IMS/DB2)
Description      ==>

Enter END to exit

```

Figure 4.8: SPICE Operator Subsystem System Definition

Enter a suitable name and description for the system you are defining. Enter its type, DB2 or IMS, and press Enter.

### SPICE Operator Subsystem DB2 System Definition

The following panel is displayed for system table entry type “DB2”:

```

===== SPICE Operator Subsystem =====
OPTION ==> _
----- SPICE System Entry Edit -----

System name      ==> DSN1          Type ==> DB2          (IMS/DB2)
Description      ==> DB2 test system
DB2 command      ==> CAF SYSTEM(<DB2 IDENTIFIER>)
RUN subcommand   ==> RUN PROGRAM(SPIUT100) PLAN(<PLAN NAME>)

Enter END to exit

```

Figure 4.9: SPICE Operator Subsystem DB2 System Entry Definition

Edit the screen to specify the DB2 identifier and plan name values appropriate to the system you are defining. The SPICE Installation jobs use plan name SPIUTI. Press Enter to define the DB2 system to the SPICE operator subsystem.

## SPICE Operator Subsystem IMS System Definition

The following panel is displayed for system table entry type “IMS”:

```
===== SPICE Operator Subsystem =====
OPTION ==> _
----- SPICE System Entry Edit -----

System name      ==> IMSA          Type   ==> IMS      (IMS/DB2)
Description      ==> IMS test system
IMS Parameter String ==> BMP,SPIUTI00,<PSBNAME>,,,,,,,,,<IMS IDENTIFIER>

Enter END to exit
```

Figure 4.10: SPICE Operator Subsystem IMS System Entry Definition

Edit the screen to replace the PSB name and IMS identifier values with those appropriate to the system you are defining. The parameter string contents are in the same format as that used in the IMS BMP procedure `IMSBATCH`. They are passed to IMS when the SPICE operator subsystem is connected to the IMS system. Include within it the parameters appropriate to your IMS system. The string can be continued onto the entry area on the next line of the screen.

The following example is suitable for connecting to IMS system `IVP1`, using PSB name `SPIDCPSB` (the name used in the Installation jobs) and application group name `IVP`:

```
BMP,SPIUTI00,SPIDCPSB,,,,,,,,,IVP1,IVP,,
```

The following example is suitable for connecting the SPICE operator subsystem to IMS system `IMSA`, using PSB name `SPIFPPSB` and application group name `IMSAAGN`, with fast path buffer normal allocation of 6 and overflow allocation of 3:

```
BMP,SPIUTI00,SPIFPPSB,,,,,,,,,6,3,IMSA,IMSAAGN,,
```

Press Enter to define the IMS system to the SPICE operator subsystem.



---

## Deploy the SPICE Operator Subsystem

Whilst the technique employed in the previous topic [Verify Installation of the SPICE Operator Subsystem](#) on page 14 could be used regularly to open the SPICE Operator System, most users would prefer that it be available from the normal ISPF panels.

### SPICE Operator Subsystem ISPF Environment Requirements

The following options are suggested, to ensure that the correct IMS and/or DB2 libraries are available from ISPF.

- 1) Make the SPICE ISPF REXX Execs available in users' TSO sessions by copying members 'SPIISPFDD/R/X' and 'SPIOPS' of 'SPICE.SPI31SYS.EXEC' to a REXX Exec library allocated to 'SYSPROC' or 'SYSEXEC' in their TSO Logon procedure.
- 2) Include the libraries required by the SPICE Operator Subsystem available in users' ISPF sessions by performing one of the following alternatives:
  - a. In the Command option of the TSO/E Logon panel, select the appropriate 'SPIISPFDD/R/X' SPICE ISPF REXX Exec, i.e.

```
----- TSO/E LOGON -----  
  
Enter LOGON parameters below:                                RACF LOGON parameters:  
Userid   ==> MYID                                           Seclabel   ==>  
Password ==>                                               New Password ==>  
Procedure ==> ISPFPROC                                       Group Ident ==>  
Acct Nbr ==>  
Size     ==>  
Perform  ==>  
Command  ==> %SPIISPFXR  
  
Enter an 'S' before each option desired below:  
          -Nomail          -Nonotice          -Reconnect          -OIDcard  
PF1/PF13 ==> Help      PF3/PF15 ==> Logoff      PA1 ==> Attention      PA2 ==> Reshow  
You may request specific help information by entering a '?' in any entry field
```

Figure 4.14: TSO/E Logon Panel Settings

- b. Incorporate into your ISPF REXX Exec (or CLIST) logic from the appropriate 'SPIISPFDD/R/X' SPICE REXX Exec.

- c. Modify your TSO Logon Procedure, as follows:

<i>Activity</i>	<i>Component</i>	<i>Notes</i>
User edit	DL/I	<pre> ... //STEPLIB DD DISP=SHR,DSN=IMS.SDFSRESL //          DD DISP=SHR,DSN=SPICE.SPI31SYS.SPILOAD //          DD other concatenated libraries ... //DFSRESLB DD DISP=SHR,DSN=IMS.SDFSRESL ... //DFSESLL DD DISP=SHR,DSN=IMS.SDFSRESL ... //PROCLIB DD DISP=SHR,DSN=IMS.PROCLIB ... </pre>
User edit	SQL	<pre> ... //STEPLIB DD DISP=SHR,DSN=DB2.SDSNLOAD //          DD DISP=SHR,DSN=SPICE.SPI31SYS.SPILOAD //          DD other concatenated libraries ... </pre>
User edit	SQL & DL/I	<pre> ... //STEPLIB DD DISP=SHR,DSN=DB2.SDSNLOAD //          DD DISP=SHR,DSN=IMS.SDFSRESL //          DD DISP=SHR,DSN=SPICE.SPI31SYS.SPILOAD //          DD other concatenated libraries ... //DFSRESLB DD DISP=SHR,DSN=IMS.SDFSRESL ... //DFSESLL DD DISP=SHR,DSN=IMS.SDFSRESL //          DD DISP=SHR,DSN=DB2.SDSNLOAD ... //PROCLIB DD DISP=SHR,DSN=IMS.PROCLIB ... //SPIOFDB2 DD DUMMY ... </pre>

Table 4.3: TSO Logon Procedure Modifications

## SPICE Operator Subsystem ISPF Panel Requirements

The following ISPF panel changes are suggested to add a SPICE Operator Subsystem option to an appropriate ISPF panel.

<i>Activity</i>	<i>Component</i>	<i>Notes</i>
User edit	all	<pre> ... + %S+ SPICE - SPICE Operator Subsystem ... S,'CMD(%SPIOPS) NEWAPPL(SPIOPS)' ... </pre>

Table 4.4: SPICE Operator Subsystem ISPF Panel Requirements

## SPICE Operator Subsystem Security

By default, SPICE relies for security on customer control of access to the SPICE libraries and the SPICE Restart Databases. Should further security be required, the SPICE utility security exit, detailed in [Chapter 6. SPICE Utility Security Exit](#) on page 27 of this manual, can be used to control access to the commands of the SPICE Operator Subsystem and the SPICE Utility program.

---

# Chapter 5. SPICE Sample Applications

This chapter completes the verification of the installation of the SPICE products. It introduces sample DB2 and IMS SPICE applications. These example programs exercise SPICE functionality, and serve as examples of the coding required in SPICE applications.

[SPICE SQL Sample Applications](#) on page [21](#)

[SPICE DL/I Sample Application](#) on page [24](#)

---

## SPICE SQL Sample Applications

SPICE SQL Installation includes two sample applications, one requesting SPICE facilities using SQL statements, and the other the SPICE SQL Restart Application Programming Interface (SPICE Restart API). The SPICE Application Testing Facility is used to verify SPICE Restart.

The sample applications use the DB2 sample application databases and programs. The SPICE sample applications are both adaptations of the batch COBOL phone application, program DSN8BC3. Before installing the SPICE sample applications, you must set up the DB2 sample application, so that program DSN8BC3 can execute. The DB2 documentation describes their installation. This is documented in chapter [Verifying with the sample applications](#) of the manual [DB2 Universal Database for z/OS Installation Guide](#).

## SPICE SQL Sample Application Installation

Submit the following members of the installation library.

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
DB2 sample JCL			Set up DB2 sample application, such that program DSN8BC3 can be executed <ul style="list-style-type: none"> <li>Consult DB2 documentation for full details</li> </ul>
Submit the following installation library members:			
I51RIVPP	SQL	none	<ul style="list-style-type: none"> <li>Allocate SPICE SAM files</li> <li>Create sample programs (by applying updates to DB2 sample program DSN8BC3)</li> <li>Pre-compile, compile and link-edit sample programs               <ul style="list-style-type: none"> <li>A link editor unresolved external reference to module DSN8MCG indicates incomplete DB2 sample application setup</li> </ul> </li> </ul>
I51RIVPD	SQL	4 on step GRANT	<ul style="list-style-type: none"> <li>DB2 bind sample programs               <ul style="list-style-type: none"> <li>Bind failure for tables VPHONE and/or VEMPLP indicates incomplete DB2 sample application setup</li> </ul> </li> </ul>

Table 5.1: Install SPICE SQL sample programs

## Installation Verification (SQL statements)

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Test the sample program, by submitting the installation library member I53RIVPR in the following variations:
I53RIVPR	SQL	none	<ul style="list-style-type: none"> <li>Unchanged (normal execution)               <ul style="list-style-type: none"> <li>No SPICE messages should be issued</li> </ul> </li> </ul>
		none	<ul style="list-style-type: none"> <li>Introduce JCL statement '//SPIOTERR DD DUMMY'               <ul style="list-style-type: none"> <li>SPICE messages SPI0261I and SPI0122I should be issued</li> </ul> </li> </ul>
		ABEND U4090	<ul style="list-style-type: none"> <li>Replace JCL statement '//SPIOTERR ...' with '//SPIOTDIE DD DUMMY'               <ul style="list-style-type: none"> <li>SPICE messages SPI0112I, SPI0260I and SPI0208I should be issued</li> </ul> </li> </ul>
		none	<ul style="list-style-type: none"> <li>Remove JCL statement '//SPIOTDIE ...'               <ul style="list-style-type: none"> <li>SPICE messages SPI0112I should be issued.</li> </ul> </li> </ul>
		none	<ul style="list-style-type: none"> <li>Unchanged               <ul style="list-style-type: none"> <li>No SPICE messages should be issued</li> </ul> </li> </ul>

Table 5.2: Verify SPICE SQL Sample Application (SQL statements)

## Installation Verification (SPICE Restart API)

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Test the sample program, by submitting the installation library member I54RIVPR in the following variations:
I54RIVPR	SQL	none	<ul style="list-style-type: none"> <li>Unchanged (normal execution)</li> <li>No SPICE messages should be issued</li> </ul>
		none	<ul style="list-style-type: none"> <li>Introduce JCL statement '//SPIOTERR DD DUMMY'</li> <li>SPICE messages SPI0261I and SPI0122I should be issued</li> </ul>
		ABEND U4090	<ul style="list-style-type: none"> <li>Replace JCL statement '//SPIOTERR ...' with '//SPIOTDIE DD DUMMY'</li> <li>SPICE messages SPI0112I, SPI0260I and SPI0208I should be issued</li> </ul>
		none	<ul style="list-style-type: none"> <li>Remove JCL statement '//SPIOTDIE ...'</li> <li>SPICE messages SPI0112I should be issued.</li> </ul>
		none	<ul style="list-style-type: none"> <li>Unchanged</li> <li>No SPICE messages should be issued</li> </ul>

Table 5.3: Verify SPICE SQL Sample Application (SPICE Restart API statements)

# SPICE DL/I Sample Application

SPICE DL/I Installation includes a sample application, and jobs to test it in the IMS Batch and BMP environments. The SPICE Application Testing Facility is used to verify SPICE Restart.

## IMS System Preparation

The database and program resources used in the SPICE DL/I sample application must be defined to the IMS system.

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Include the following statements in your IMS system definition: <ul style="list-style-type: none"> <li>Database definition statements               <pre>                 DATABASE   DBD=SPICHSAM                 DATABASE   DBD=SPIVB2CS                 DATABASE   DBD=SPIVB3OR                 DATABASE   DBD=SPIVB3OX                 DATABASE   DBD=SPIVB3PR                 DATABASE   DBD=SPIVB3PT               </pre> </li> <li>PSB definition statements               <pre>                 APPLCTN   PSB=SPIDCPSB,PGMTYPE=BATCH,      X                         SCHDTYP=PARALLEL                 APPLCTN   PSB=SPIVP3PR,PGMTYPE=BATCH                 APPLCTN   PSB=SPIVP3OR,PGMTYPE=BATCH                 APPLCTN   PSB=SPIVP4CP,PGMTYPE=BATCH                 APPLCTN   PSB=SPIVPSUA,PGMTYPE=BATCH               </pre> </li> </ul>
			Perform an IMS MODBLKS generation
			Re-start your IMS system, or perform IMS On-Line Change

Table 5.4: Prepare IMS System

## SPICE DL/I Sample Application Installation

Perform the following:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Submit the following installation library members:
I61DIVPP	DL/I	4 on steps SPIVCPUR and ACBGEN	<ul style="list-style-type: none"> <li>• Compile SPICE DL/I IVP programs</li> <li>• Prepare SPICE DL/I IVP DBDs, PSBs and ACBs</li> </ul>
I62DIVPD	DL/I	4 on step PRERESOL and 12 on DBRCDEL	<ul style="list-style-type: none"> <li>• Create SPICE DL/I IVP generation data groups.</li> <li>• Prepare SPICE DL/I IVP dynamic allocation blocks.</li> <li>• Create and load SPICE DL/I IVP application data bases</li> <li>• Allocate SPICE SAM files</li> <li>• Register SPICE DL/I IVP databases with DBRC</li> <li>• Execute IMS On-Line Change Utility</li> </ul>
			Issue the following commands to IMS, to implement the prepared IMS On-Line Change:  /MODIFY PREPARE ACBLIB /MODIFY COMMIT

Table 5.5: Install SPICE DL/I sample programs and databases

## Installation Verification

The SPICE Application Testing Facility is used to verify that SPICE Restart is functioning correctly.

Perform the following to verify the SPICE DL/I sample program in IMS Batch:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Test the sample program, by submitting the installation library member I63DIVBA in the following variations:
I63DIVBA	DL/I	none	<ul style="list-style-type: none"> <li>Unchanged (normal execution)</li> <li>No SPICE messages should be issued</li> </ul>
I63DIVBA	DL/I	ABEND U3400	<ul style="list-style-type: none"> <li>Introduce JCL statement '//SPIVTERR DD DUMMY'</li> <li>SPICE messages SPI0261I and SPI0208I should be issued</li> </ul>
I64DIVBO	DL/I	none	Submit this job to perform IMS Batch Backout
I63DIVBA	DL/I	ABEND U4090	<ul style="list-style-type: none"> <li>Replace '//SPIVTERR' by '//SPIVTDIE DD DUMMY'</li> <li>SPICE messages SPI112I and SPI0261I should be issued</li> </ul>
I64DIVBO	DL/I	none	Submit this job to perform IMS Batch Backout.
I63DIVBA	DL/I	none	<ul style="list-style-type: none"> <li>Remove JCL statement '//SPIVTDIE DD DUMMY'</li> <li>SPICE message SPI112I should be issued</li> </ul>
I63DIVBA	DL/I	none	<ul style="list-style-type: none"> <li>Unchanged</li> <li>No SPICE messages should be issued</li> </ul>

Table 5.6: Verify SPICE DL/I Sample Program (IMS batch)

Perform the following to verify the SPICE DL/I sample program as an IMS BMP:

<i>Installation Library Member Name</i>	<i>Component</i>	<i>Allowable non-zero Return Codes</i>	<i>Notes</i>
			Test the sample program, by submitting the installation library member I65DIVBM in the following variations:
I65DIVBM	DL/I	none	<ul style="list-style-type: none"> <li>Unchanged (normal execution)</li> <li>No SPICE messages should be issued</li> </ul>
I65DIVBM	DL/I	ABEND U3400	<ul style="list-style-type: none"> <li>Introduce JCL statement '//SPIVTERR DD DUMMY'</li> <li>SPICE messages SPI0261I and SPI0208I should be issued</li> </ul>
			Restart the program with an IMS 'START PROGRAM(SPIVP4CP)' command
I63DIVBA	DL/I	ABEND U4090	<ul style="list-style-type: none"> <li>Replace '//SPIVTERR' by '//SPIVTDIE DD DUMMY'</li> <li>SPICE messages SPI112I and SPI0261I should be issued</li> </ul>
			Restart the program with an IMS 'START PROGRAM(SPIVP4CP)' command
I65DIVBM	DL/I	none	<ul style="list-style-type: none"> <li>Remove JCL statement '//SPIVTDIE DD DUMMY'</li> <li>SPICE message SPI112I should be issued</li> </ul>
I65DIVBM	DL/I	none	<ul style="list-style-type: none"> <li>Unchanged</li> <li>No SPICE messages should be issued</li> </ul>

Table 5.7: Verify SPICE DL/I Sample Program (IMS BMP)

---

# Chapter 6. SPICE Utility Security Exit

---

## Introduction

SPICE has an optional Utility Security Exit. With the standard installation, a user's access to the SPICE Utility program and Operator Subsystem is controlled by the installation's access policies to the SPICE libraries and the SPICE Restart Database. The SPICE Utility Security Exit routine may be used to further control access to the different SPICE Utility commands, and their Operator Subsystem counterparts.

This chapter describes how the SPICE security user exit routine may be used to control access security to the SPICE operator facilities.

---

## Processing Overview

SPICE calls the SPICE security user exit from the SPICE operator subsystem and the SPICE utility program. It passes the exit a standard parameter list containing the address of the SXCB control block. This control block is mapped by assembler DSECT macro SPIUSXCB, which can be found in library SPICE.SPI31DLB.SPINMAC. The exit responds by returning a code in general register 15.

Upon initialization of the SPICE operator subsystem or SPICE utility program, SPICE will determine if the SPICE user exit is installed, and, if so, call it, requesting its initialization. The exit should return one of the following codes in register 15:

- Zero* SPICE will continue execution.
- Non-zero* SPICE will refuse access to the operator subsystem or utility, by issuing failure message SPI04511 before abnormally terminating.

Upon each option or command capable of changing the contents of a SPICE restart database, SPICE will call the SPICE security exit, detailing the request in the SXCB control block. The exit should return one of the following codes in register 15:

- Zero* SPICE will continue execution and process the request.
- Non-zero* SPICE will deny access to the option or command, by issuing error message SPI04501, and not executing the request.

Upon termination of the SPICE operator subsystem or SPICE utility program, SPICE will call the SPICE user exit, requesting its termination.

---

## Installation

The SPICE security exit is installed by compiling and link-editing the exit routine source code into load module SPIUEX01 in the SPICE resident library. SPICE will automatically use the exit routine, if present.

---

## Sample Exit

A sample exit is supplied in member SPIUEX01 of library SPICE.SPI31SAMPLIB. It contains simple lists of user identifiers and job names that are allowed to use certain SPICE operator subsystem and utility facilities. The user may want to modify it to use the installation's resource access control facility.

---

## Appendix A. SPICE Data Sets

This appendix documents the data sets employed in the installation of the SPICE products.

---

### Distributed Material

The SPICE products are distributed as the following data sets.

<i>Library</i>	<i>Contents</i>
SPICE.SPI31.XMITPDSN.XMIT	SPICE Base data set (A TSO TRANSMIT packaged PDS of TSO TRANSMIT packaged PDS data sets).
SPICE.SPI31.XMITPDSR.XMIT	SPICE SQL data set (A TSO TRANSMIT packaged PDS of TSO TRANSMIT packaged PDS data sets).
SPICE.SPI31.XMITPDS.D.XMIT	SPICE DL/I data set (A TSO TRANSMIT packaged PDS of TSO TRANSMIT packaged PDS data sets).
UNPAK.JCL	z/OS batch job to unpack the installation libraries.

Table A.1: SPICE Distributed Material

These data sets are record format FB, record length 80, blocksize 6160. Data set UNPAK.JCL is an ASCII file, the others are binary.

---

### Unpacked Distributed Material

**The SPICE distribution material is first unpacked into the following libraries.**

<i>Library</i>	<i>Contents</i>
SPICE.SPI31.XMITPDSN	SPICE Base data set (A PDS library of TSO TRANSMIT packaged PDS data sets).
SPICE.SPI31.XMITPDSR	SPICE SQL data set (A PDS library of TSO TRANSMIT packaged PDS data sets).
SPICE.SPI31.XMITPDS.D	SPICE DL/I data set (A PDS library of TSO TRANSMIT packaged PDS data sets).

Table A.2: SPICE unpacked distribution material

**These libraries unpack into the following installation libraries.**

<i>Library</i>	<i>Contents</i>
SPICE.SPI31.INSTALL	SPICE installation jobs.
SPICE.SPI31.INSTLIB	SPICE installation REXX Execs.
SPICE.SPI31.SAMPLIB	SPICE installation verification jobs.
SPICE.SPI31.VSN3100A	SPICE Base Assembler source library.
SPICE.SPI31.VSN3100G	SPICE Base ISPF message library.
SPICE.SPI31.VSN3100J	SPICE Base SMP/E JCLIN library.
SPICE.SPI31.VSN3100L	SPICE Base load module library.
SPICE.SPI31.VSN3100M	SPICE Base Assembler macro library.
SPICE.SPI31.VSN3100P	SPICE Base fix library.
SPICE.SPI31.VSN3100V	SPICE Base ISPF panel library.
SPICE.SPI31.VSN3100X	SPICE Base REXX Exec library.

Table A.3: SPICE Base installation libraries

<i>Library</i>	<i>Contents</i>
SPICE.SPI31.VSD3100A	SPICE DL/I Assembler source library.
SPICE.SPI31.VSD3100J	SPICE DL/I SMP/E JCLIN library.
SPICE.SPI31.VSD3100L	SPICE DL/I load module library.
SPICE.SPI31.VSD3100M	SPICE DL/I Assembler macro library.
SPICE.SPI31.VSD3100P	SPICE DL/I fix library.

Table A.4: SPICE DL/I installation libraries

<i>Library</i>	<i>Contents</i>
SPICE.SPI31.VSR3100A	SPICE SQL Assembler source library.
SPICE.SPI31.VSR3100B	SPICE SQL DB2 DBRM library.
SPICE.SPI31.VSR3100J	SPICE SQL SMP/E JCLIN library.
SPICE.SPI31.VSR3100L	SPICE SQL load module library.
SPICE.SPI31.VSR3100M	SPICE SQL Assembler macro library.
SPICE.SPI31.VSR3100P	SPICE SQL fix library.
SPICE.SPI31.VSR3100R	SPICE SQL pre-compiled source library.

Table A.5: SPICE SQL installation libraries

---

## SMP/E Distribution Libraries

The SMP/E installation of SPICE creates the following SMP/E distribution libraries.

<i>Library</i>	<i>Contents</i>
SPICE.SPI31DLB.SPIINSTL	SPICE installation jobs.
SPICE.SPI31DLB.SPIIEXEC	SPICE installation REXX Execs.
SPICE.SPI31DLB.SPIISAMP	SPICE installation verification jobs.
SPICE.SPI31DLB.SPINASM	SPICE Base assembler source library.
SPICE.SPI31DLB.SPINCNTL	SPICE Base installation JCL library.
SPICE.SPI31DLB.SPINEXEC	SPICE Base REXX procedure library.
SPICE.SPI31DLB.SPINLOAD	SPICE Base load module library.
SPICE.SPI31DLB.SPINASM	SPICE Base assembler source library.
SPICE.SPI31DLB.SPINMAC	SPICE Base assembler macro library.
SPICE.SPI31DLB.SPINMLIB	SPICE Base ISPF message library.
SPICE.SPI31DLB.SPINPLIB	SPICE Base ISPF panel library.
SPICE.SPI31DLB.SPINSLIB	SPICE Base ISPF skeleton library.
SPICE.SPI31DLB.SPINTLIB	SPICE Base ISPF table library.

Table A.6: SPICE Base SMP/E distribution libraries

<i>Library</i>	<i>Contents</i>
SPICE.SPI31DLB.SPIRASM	SPICE SQL assembler source library.
SPICE.SPI31DLB.SPIRDBRM	SPICE SQL DBRM library.
SPICE.SPI31DLB.SPIRLOAD	SPICE SQL load module library.
SPICE.SPI31DLB.SPIRMAC	SPICE SQL assembler macro library.
SPICE.SPI31DLB.SPIRSASM	SPICE SQL precompiled assembler source library.

Table A.7: SPICE SQL SMP/E distribution libraries

**SPICE DL/I also installs into the following distribution libraries.**

<i>Library</i>	<i>Contents</i>
SPICE.SPI31DLB.SPIDASM	SPICE DL/I assembler source library.
SPICE.SPI31DLB.SPIDLOAD	SPICE DL/I load module library.
SPICE.SPI31DLB.SPIDMAC	SPICE DL/I assembler macro library.

Table A.8: SPICE DL/I SMP/E distribution libraries

---

## SPICE Run-Time Libraries

All SPICE products install into the following system run-time libraries.

<i>Library</i>	<i>Contents</i>
SPICE.SPI31SYS.EXEC	SPICE REXX Exec library.
SPICE.SPI31SYS.ISPMLIB	SPICE ISPF message library.
SPICE.SPI31SYS.ISPPLIB	SPICE ISPF panel library.
SPICE.SPI31SYS.ISPSLIB	SPICE ISPF skeleton library.
SPICE.SPI31SYS.ISPTLIB	SPICE ISPF table library.
SPICE.SPI31SYS.SPILOAD	SPICE load library.
SPICE.SPI31SYS.SPINLLIB	SPICE ISPF load library.
SPICE.SPI31SYS.DBRMLIB	SPICE SQL DBRM library.

Table A.9: SPICE SQL & SPICE DL/I run-time libraries