Adabas SOA Gateway

Installation and Operation

Version 2.6.1

November 2016

ADABAS & NATURAL

This document applies to Adabas SOA Gateway Version 2.6.1.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2006-2016 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at http://softwareag.com/licenses.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at http://softwareag.com/licenses/ and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at http://softwareag.com/licenses and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: APS-IMS-261-20161123

Table of Contents

Installation and Operation v
1 Installing the IMS/DC Client Environment in a Message Processing Region (MPR) 1
2 User documentation for SMARTS IMS/DC interface 5
Log Messages 6
Online Messages 7

Installation and Operation

This documentation provides information on how to install and operate the IMS/DC Client Environment in a Message Processing Region (MPR) for the use in conjunction with Software AG's Multiple Architecture Runtime System (SMARTS).

The information on installing and operating SMARTS is structured as follows:

Installing the IMS/DC Client Environment in a Message Processing Region (MPR)	Informations how to install the IMS/DC Client Environment in a Message Processing Region (MPR)
User documentation for SMARTS IMS/DC interface	The user documentation for SMARTS IMS/DC interface

1 Installing the IMS/DC Client Environment in a Message Processing Region (MPR)

Install the IMS/DC Client Environment in a Message Processing Region (MPR)

1 1. **Create a member which will hold all programs which need to be pre-loaded into the MPR.**

   The member must be named DFSMPLxx where xx is the suffix characters designated by the PRLD parameter in the MPR JCL.

2. **Include list of programs which invoke SMARTS facilities.**

   The pre-load program list is used to ensure that all programs that use SMARTS do not get deleted from the MPR storage during the life of their unit-of-work (transaction). Add all User-written application program names which use SoftwareAG software – e.g. Adabas SQL Server, XTS or any other required SoftwareAG optional programs.

2 1. **Create a member which will hold all programs which will need to run at MPR initialisation.**

   The member must be named DFSINTxx where xx is the suffix characters designated by the PREINIT parameter in the MPR JCL.

2. **Include POSIX Initialisation Table to the list.**

   The pre-initialisation program list is used to automatically initialise SMARTS POSIX Kernel at MPR start-up: Add PAINKERN to the list of programs to execute at MPR initialisation.

3 **Create the MPR Procedure**

Caution: All IMS applications which invoke SMARTS facilities must run in a dedicated IMS MPR.

1. **Create the JCL to invoke a MPR**

2. **Set a single dedicated Class parameter to run in this MPR – e.g. CL1=032.**

3. **Set a Pre-initialisation parameter to point to the dataset member which holds all modules to be run at MPR initialisation time – e.g. PREINIT=AQ.**

4. **Set a Preload parameter to point to the dataset member which holds all programs to be preloaded – e.g. PRLD=AP.**

5. **Include the IBM OS/390 Callable Service linkage-assist routine library Add SYS1.CSSLIB to the STEPLIB concatenation in the procedure.**

6. **Include the SMARTS load library**

   Add APSvvv.MVSLD00 dataset to the STEPLIB concatenation in the procedure- where vvv is the SMARTS version.

7. **Include other SoftwareAG libraries**

   Add other required SoftwareAG datasets as directed by the Installation Procedure of each product.

8. **Add the SYSPARM dataset**

   Add a "//SYSPARM DD" statement for the SMARTS POSIX configuration. This dataset is required to define the runtime characteristics of your POSIX environment. For more information, see SMARTS Configuration Sources.

9. **Add a dataset for environment variables**

   Add a "//CONFIG DD" statement for the dataset containing the environment variables required for your POSIX applications within IMS/DC. This DD name is specified by the ENVIRONMENT_VARIABLES parameter in SYSPARM, which defaults to "CONFIG". For more information, see the section SMARTS POSIX Miscellaneous Parameters, ENVIRONMENT_VARIABLES.

10. **Optional SoftwareAG supplied programs**

    PAINKERX – This program may be used to communicate with the SMARTS Kernel to initialize and terminate the Kernel, and to invoke Kernel commands. PAINUSNF – This performs a User sign-off of the SMARTS IMS/DC environment. This does not affect the normal running of the MPR if the associated transaction with this program is not used when a user ends an IMS session.

11. **IMS Sysgen requirements**

    Any optional SoftwareAG programs which are to be used will have to be included in the IMS Sysgen together with the transaction names associated with these programs. The transaction names may be named to installation standards.

12. **IMS/DC SMARTS interface module list**

    The following programs support the SMARTS IMS/DC interface:

    PAIAINT
    PAINENVF
    PAINITPT
    PAINKERN
    PAINKERX*
    PAINKTXT
    PAINMNIT
    PAINNMIT
    PAINPGMT
    PAINUSNF*

Where Next ?

You have now installed and configured the SMARTS IMS/DC software. All User-written IMS/DC application programs which interface with SoftwareAG software must link-edit the SMARTS Environment Dependent Initialisation module PAIAINT, and, if required, SMARTS main Environment Dependent wrapper module PAINMNIT, or SMARTS non-main Environment Dependent wrapper module PAINNMIT, into the application program load module. For more information, see the SMARTS documentation.

2 User documentation for SMARTS IMS/DC interface

- Log Messages 6
- Online Messages 7

This section describes:

Log Messages

APSPSX0089-nnnnnnn SMARTS/IMS: ffffffff of token failed - yyyy zzzzzzzz, return code n

Explanation This informs that the request ffffffff of MVS Token Services has returned an error code n for the token value zzzzzzzz

where :

nnnnnnnn
system id

ffffffff
function - Create, Retrieve or Delete

yyyy
IMS system id

zzzzzzzz
token value

n
return code from MVS Token Services

Action Contact SoftwareAG Customer Support Desk giving the exact error message for further diagnosis and support

**APSPSX0098-nnnnnnnn SMARTS/IMS: tttttt

Message text	Reason	Action
No IMS parameter list found (ppppppp) - where ppppppp is a SoftwareAG program id	IMS parameter cannot be found	Check that the IMS system is active and that the program is running in the correct MPR for SoftwareAG specific applications. Otherwise contact SoftwareAG Customer Support Desk.

Online Messages

Issued by PAINKERX and prefixed by

APSPSX0099

- No parameters/command passed with transaction id
- No IMS parameter list found (see above for further action)
- Kernel initialised successfully
- Kernel terminated successfully
- Kernel command issued successfully
- Kernel tttttttt error - return code: cc, reason code: rr

where

tttttttt
is the request type - Initialisation, Termination or Command

cc
is the return code

rr
is the reason code

Note: refer to SMARTS documentation for information on return and reason codes

Issued by PAINUSNF

None. See SMARTS log for all messages.

Issued by PAINMNIT and prefixed by -

APSPSX0099

Program *pppppppp* terminated with return code *rr*

where

pppppppp
is the application program name

rr
is the return code (if any) returned by the application program