Software AG

Adabas for Linux, UNIX and Windows

Adabas REST Administration

Version 7.0

October 2022

ADABAS & NATURAL

Table of Contents

Preface

Adabas RESTful administration is a server that provides the REST interface for Adabas monitoring and Adabas administration tasks. Any RESTful client can connect to the server and use standard web techniques to retrieve Adabas information.

The Adabas REST Administration document is organized as follows:

General Information	Describes in short the concepts and components of the Adabas RESTful administration.
Installation and First Steps	Contains information on how to install and configure Adabas RESTful administration.
RESTful Security	Describes the concepts and administration of Adabas RESTful server security topics such as SSL-encrypted connections and access restrictions to Adabas data.
Adding Support for Multiple Adabas Versions	Contains information on how to enable support for multiple Adabas versions in the Adabas RESTful administration.
System Service	Describes the system service provided with the Adabas RESTful administration.
Messages	Contains information about the response codes that are returned if errors occur while processing.

1 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *`folder.subfolder.service`*, APIs, Java classes, methods, properties.
`Italic`	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Product Documentation

You can find the product documentation on our documentation website at **https://documenta-tion.softwareag.com**.

In addition, you can also access the cloud product documentation via **https://www.software-ag.cloud**. Navigate to the desired product and then, depending on your solution, go to "Developer Center", "User Center" or "Documentation".

Product Training

You can find helpful product training material on our Learning Portal at **https://knowledge.soft-wareag.com**.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at **https://tech-community.softwareag.com**. From here you can, for example:

- Browse through our vast knowledge base.

- Ask questions and find answers in our discussion forums.

- Get the latest Software AG news and announcements.

- Explore our communities.

- Go to our public GitHub and Docker repositories at **https://github.com/softwareag** and **https://hub.docker.com/publishers/softwareag** and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at **https://empower.softwareag.com**. Many services on this portal require that you have an account. If you do not yet have one, you can request it at **https://empower.softwareag.com/register**. Once you have an account, you can, for example:

- Download products, updates and fixes.

- Search the Knowledge Center for technical information and tips.

- Subscribe to early warnings and critical alerts.

- Open and update support incidents.

- Add product feature requests.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 General Information

Adabas RESTful administration consists of three main access parts:

Adabas Administration and Monitor Handler

The Adabas administration and monitor handler supports:

Resource	Description
Adabas Database	An Adabas database can be created or deleted on the server.
Adabas Runtime	Adabas can be started or stopped using the possibilities that Adabas provides. User queue and UCB entries can be managed. Provides Adabas Nucleus log.
Monitor	Display Adabas user queue, command queue and hold queue. Adabas high water marks and buffer pool statistics are available. All possible monitor features can be found in the Swagger definition.
Adabas Files	Provide the creation and deletion of new Adabas files (online) .
Adabas Fields	Inside the Adabas file, Adabas fields can be modified in the frame of the Adabas functionality.

Job Control

Inside the RESTful server it is possible to start scripts for Adabas utilities. The scripts can be triggered asynchronously. After the job is finished, the corresponding log output can be retrieved using a RESTful access.

Detailed information about the RESTful interface is described in the Swagger definition.

File Transfer Handler

The RESTful server provides an interface to upload or download files to or from the server. The server only provides access to restricted, defined directories. The directories have to be configured in the configuration file.

Each configuration entry can be accessed via a given name. All subdirectories under the configured one can be accessed and files can be transferred.

If a job creates output, the created file can be transferred to the HTTP client (browser or RESTful client).

Technical Insight

Adabas RESTful administration includes standard RESTful HTTP access methods. It supports two types of response formats:

- JSON is the standard supported response format you can use.
- XML can be used for read requests.

Adabas RESTful administration also supports SSL-secured HTTPS connections. HTTPS connections are recommended. For further information about configuring secure REST connections and secure data resource accesses, refer to the section *RESTful Security*.

Adabas RESTful administration supports the following HTTP/S request types:

- GET: Retrieve administration or monitor information
- PUT: Change an administration value or parameter in the Adabas database configuration
- PUSH: Create new objects such as Adabas database files or an Adabas database
- DELETE: Delete a resource inside the Adabas database

The Adabas REST administration server is open to be used by any RESTful client. There are no restrictions by Software AG components. The RESTful definition is provided in the Swagger file located in the product installation. You can use the Swagger/OpenAPI infrastructure to use the Swagger file.

Adabas Batch Administration

On the Software AG GITHUB page, you can download a batch administration client. The client application can access all flavors of Adabas administration tasks that the Adabas RESTful administration server provides.

The GITHUB page can be found at <**https://github.com/SoftwareAG/adabas-admin-restful-client**>.

The following example shows the batch command to list all available databases on the remote server:

```
client -url <host>:<port> list
```

The output of this example out would look something like this:

```
2018/10/10 12:40:54 Adabas Administration RESTful client started

2018/10/10 12:40:54 Server: linhost:8390
2018/10/10 12:40:54 User:   admin

Enter Password:
 Dbid   Name                   Active    Version

  001 [TestDatabase    ]       false    Adabas v6.6 (20)
  015 [SAMPLE_DB       ]       false    Adabas v6.6 (20)
  050 [GENERAL_DATABASE]       false    Adabas v6.5 (19)
  075 [GENERAL_DATABASE]       false    Adabas v6.7 (21)
  102 [GENERAL_DATABASE]       false    Adabas v6.6 (20)
  155 [SAMPLE_DB       ]       false    Adabas v6.7 (21)
  195 [DEMODB          ]       false    Adabas v6.7 (21)


2018/10/10 12:40:57 Adabas Administration RESTful client took 61.262267ms terminated
```

In batch mode, the ADABAS_ADMIN_URL environment points to the Adabas RESTful administration server, and the ADABAS_ADMIN_PASSWORD environment can contain the user password.

The Adabas batch client supports the following commands/functionality:

Command	Functionality
env	List Adabas environment version
list	List all Adabas databases
start	Start Adabas database
shutdown	Shutdown Adabas database
cancel	Cancel Adabas database
abort	Abort Adabas database
info	Retrieve Adabas database information
userqueue	Display current user queue
cmdqueue	Display current command queue
holdqueue	Display current hold queue
highwater	Display high water mark
commandstats	Display Adabas command statistics
bp	Display Adabas buffer pool statistics
activity	Display Adabas activity
threadtable	Display Adabas thread table
createdatabase	Create new Adabas database
deletedatabase	Delete an Adabas database
renamedatabase	Rename an Adabas database

Command	Functionality
parameter	List database parameter information
parameterinfo	List database parameter information with minimum and maximum ranges
setparameter	Set database parameter
nucleuslog	Display Adabas nucleus log
files	Display Adabas file list
file	Display Adabas file
deletefile	Delete Adabas file
fields	Display Adabas file definition table
information	Display Adabas database information
container	Display Adabas database container
renamefile	Rename database file
createfile	Create database file
checkpoints	Display database checkpoints. Without parameter it shows one day. The following is a parameter example with from and to parameter: `2018-05-15_01:00:00,2018-05-20_00:00:00`
joblist	Job control list
jobstart	Start a specific job
deletejob	Delete a specific job and the execution log
deletejobexec	Delete the execution log of a job
createjob	Create a new specific job
joblog	Job entry log
listucb	List Adabas UCB entries
deleteucb	Delete an Adabas UCB entry
addfields	Add Adabas fields
status	Adabas database online state
filelocations	List all available file locations
listfiles	List files in file location
downloadfile	Download file from the file location
uploadfile	Upload file to file location

Merge Adabas RESTful Server

The Adabas client for Java contains a module to access Adabas database data content. It is possible to enable the data access in the Adabas RESTful administration server. Please contact Software AG support for the steps required to enable this data access.

3 Installation and First Steps

Installation is done using the Software AG Installer. The Adabas REST administration is part of Adabas but does not have to be installed with Adabas. If another Adabas installation is on the installation host, this can be used, if the Adabas version is greater than or equal to Adabas Version 6.5.

Access to Adabas RESTful resources is protected using authentication. There are two possibilities to use passwords. The first possibility is to manage the configuration file *realm.properties*, which contains a list of users and passwords. The second possibility is the configuration of the JAAS infrastructure to provide access to LDAP, for example. The configuration of JAAS is managed in *configuration/security.conf*. It is recommended to use the LDAP approach, or to use any other plugin alternatives of JAAS.

This chapter is organized as follows:

Password Management

Setting the Initial Admin Password

During installation, you will be prompted by a dialog box to set the initial admin password. This password-setting feature is available from version 7.0.1 onwards.

If for any reason you need to reset the initial password at a later stage, you can do so by deleting the *realm.properties* configuration file in the *configuration* directory, and then recreating the file by running the service script as follows:

```
./service.bat init
```

Managing New Users and Passwords

You can add new user and password entries to *configuration/realm.properties*. The service script provides you with the functionality to add an additional user as follows:

```
./service.bat add_user
```

Adabas Installation Management

The Adabas RESTful server is able to manage different version of Adabas. Up to two minor version are support downwards. In this case, the Adabas Version 6.7 based Adabas RESTful server supports managing Adabas Version 6.6 and Adabas Version 6.5 databases. A prerequisite for this is to provide a corresponding Adabas version installation. The Adabas installation location has to be added to the Adabas RESTful server configuration.

The service script provides you with the functionality to add additional Adabas installations. The current status is displayed by using the following command:

```
./service.bat add_env
```

The output of this example out would look something like this:

```
2019-02-14 10:54:02 - Load default configuration from file ↵
/opt/softwareag/AdabasRestAdministration/configuration/config.xml
Current defined configurations:
---------------------------
ADADATADIR      : /data/SAG/Adabas
--------------------------------------------
Location        : /opt/softwareag
 ADAPROGDIR      : /opt/softwareag/Adabas
 Version         : V67002
 Structure level : 21
--------------------------------------------
```

Use the following command to add a new Software AG installation:

```
./service.bat add_env <Installation location>
```

This will add the Adabas installation to the Adabas RESTful configuration.

Starting the Server

Inside the installation directory of the Adabas REST administration, a *bin* directory can be found. The *bin* directory contains all of the start scripts required to start the Adabas REST administration. All scripts will provide reference information about valid parameters in case no argument is entered.

For example, to start the REST server on Windows just start:

```
./service.bat run
```

Ensuring that the Server is Working

To check if the server works correctly, you can either use your favorite web browser or use curl. By default, the username is *admin* and the password is *manage*.

Use the following command to check whether the server is running:

```
http://<host>:<port>/adabas/database
```

where `<host>` and `<port>` correspond to your environment. The following are some example REST calls:

URL	Function	Result
http://localhost:8120/adabas/database	List databases	List all databases known to the REST server
http://localhost:8120/adabas/database/24	Database Info	Returns the database information for database 24
http://localhost:8120/adabas/database/24/file	List database files	List all files defined in the database 24

Using curl, the following command would correspond to the first example:

```
curl -vv --user admin:manage http://localhost:8120/adabas/database
```

4 RESTful Security

This section provides information about Adabas RESTful server security topics such as SSL-encrypted connections and access restrictions to Adabas data.

Secure connection with SSL

The Adabas RESTful server can handle simultaneous HTTP and HTTPS connections to Adabas Manager or any other RESTful client. To prevent the RESTful server from providing an unsecured HTTP protocol link, the port number must be set to -1.

If the secure HTTPS protocol port is to be used, a keystore file (.JKS) containing the server certificate and key must be added to the RESTful server configuration. An example certificate to be used for testing purposes is delivered as part of the RESTful server release. This certificate can be found in *<SAG installation>/AdabasRestAdministratrion/keys*.

⚠ **Important:** After testing, the example certificate must be exchanged with the corporate certificates of the customer environment!

If a self-signed certificate is needed, it can be generated by a script that is delivered with the Adabas RESTful server release (available with version 7.0.1 and above). For detailed instructions on how to generate the self-signed certificate, keys and keystore file, refer to the section *Creating a Keystore file (.JKS)*.

Adabas User Authentication

The Adabas RESTful server supports two types of user authentication:

▪ User and password file called Realm.

The password is stored in an MD5 and SHA-hash in a text file.

▪ Java JAAS support.

The authentication can be done using the standard Java authentication module with Java Authentication and Authorization Service (JAAS).

Realm text file usage

Realm-based password authentication is handled by the configuration file *AdabasRestAdministration/configuration/realm.properties*. The file contains usernames and passwords. The password is stored as SHA or MD5 hashsum.

The *service.sh* script can add users to the file. The output of the command will appear on screen as shown below:

```
AdabasRestAdministration/bin> ./service.sh add_user
   JAVA_HOME: /opt/softwareag/jvm/jvm
 SERVER_HOME: /opt/softwareag/AdabasRestAdministration
 This program generates a password entry for an Adabas administration user.
 The authorization uses BasicAuth access. You may use other authorization
 methods by changing the JAAS configuration of the administration server.
 File /opt/softwareag/AdabasRestAdministration/configuration/realm.properties ↵
already available
 Add entry into realm property file: ↵
/opt/softwareag/AdabasRestAdministration/configuration/realm.properties
 Enter Adminstration user:
 sag2
 Enter password:
 Successfully initialize password entry in administration server.
```

If no *realm.properties* file is available, a new file can be generated by using the `service.sh init` command.

JAVA JAAS usage

With Adabas RESTful server version 7.0 and above, a number of additional platform specific JAAS modules are delivered to be used in authenticating users with the local system:

- On Unix:

  The Adabas RESTful server delivers a Software AG internal JAAS module using the internal Software AG Security eXtensions (SSX). An example configuration is provided in the installation in the directory *configuration/security.conf*.

- On Windows:

  The Adabas RESTful server is delivered with a JAAS module called Waffle. For more information see **https://waffle.github.io/waffle/**

Adabas User Authorization

The Adabas RESTful server provides Adabas users direct access to Adabas database administration tasks, records and data resources like Adabas Maps and Database IDs.

This user access to Adabas data resources can be controlled and limited using security definitions and Role-based Access Control (RBAC). RBAC definitions are configured on Adabas.

However, if RBAC definitions cannot be used for any reason, an alternative advanced security feature is provided on the RESTful server. This security feature allows you to configure and control access authorizations to Adabas data resources through the use of several configuration files in the *<SAG>/AdabasRestAdministration/configuration* directory. The main configuration file is *config.xml*.

The following user authorizations can be assigned using the configuration files:

- Enable access for an Adabas Database ID

- User-based access control
    - Provide Administration Permission
    - Provide Data Resources Permission

Enable access for an Adabas database ID

In the Adabas RESTful configuration (*config.xml*), direct access to Adabas databases using Database IDs is restricted with the `DatabaseAccess` tag. The `DatabaseAccess` attribute *global* defines whether all databases are accessible by default. If the attribute *global* is set to false, then any Adabas database ID used for direct access needs to be listed in the `Database` tag. In the example below, direct access is enabled only for the database with ID 123:

```
<DatabaseAccess global="false">
    <Database dbid="123" />
</DatabaseAccess>
```

User-based access control

Any user can be added to the Administrator role. In addition, the user can be restricted to access only specific Adabas Maps or Database IDs.

Within the `LoginService` tag of *config.xml* there are two tags that are used to differentiate Administrators from plain Users:

```
<LoginService module="" webTokenExpires="24">
    <Administrators ↵
file="${CURDIR}/configuration/administrator.xml"></Administrators>
    <Users file="${CURDIR}/configuration/users.xml"></Users>
</LoginService>
```

The *administrator.xml* file that is specified in the `Administrator` tag must list all users with Administrator rights. Similarly, the *users.xml* file specified in the `Users` tag must list all users that should not have administrator rights. The *users.xml* also allows you to set permissions to specific data resources.

Provide Administration permission

To provide Administrator permissions to a user, specify that user in the file *<SAG>/AdabasRestAdministration/configuration/administrator.xml*. In the example below, the users `admin` and `sag` have Administration permissions:

```
<Users>
    <User name="admin" />
    <User name="sag" />
</Users>
```

The rights to work with Adabas as Administrator covers all accesses and modifications that relate to administration and monitoring of the database

Provide Data resources permission

To restrict user access to specific Adabas data resources, you can define permissions with the `Users` tag in the file *<SAG>/AdabasRestAdministration/configuration/users.xml*. The tag allows users' read and write permissions to be set for following Adabas record-based resources:

- Adabas Map
- Database ID

An example is shown below:

```
<Users>
    <Default read="*" write="" />
    <User name="sag" read="*,#*" write="*" />
    <User name="sag2" read="Employees,Vehicles" write="Employees" />
</Users>
```

In this example:

- All users not listed under `Users` can read all Adabas Maps (*) and have no write access ("") to any Adabas Maps.

- The user `sag` has read permission to all Adabas Maps (*) and all database IDs (#*). Note that all database ID definitions need to have the prefix #. The user `sag` also has write permissions to all Adabas Maps (*).

- The user `sag2` only has read permission to the Adabas Maps named `Employees` and `Vehicles`. `sag2` only has write permission to the Adabas Map `Employees`.

If RBAC or LDAP-based authentication is used, you can disable all user restrictions by setting the `Default` tag to prove full access for all users.

JSON Web Token

Adabas RESTful server supports the use of signed JSON Web Tokens (JWT). After logging in using the `/login` web page, a JWT token is returned. The JWT token is valid for a limited time range and may be provided alternatively as a credential.

The token issuer and time limit can be configured in the configuration file *<SAG>/AdabasRestAdministration/configuration/config.xml*. In the example below, the token is configured to expire after 120 minutes:

```
<Server>
  <JsonWebToken issuer="https://softwareag.com" expire="120" />
</Server>
```

The JSON Web Token can be sent in an HTTP call as `Authorization: Bearer <jwt token>` to the Adabas RESTful server.

Creating a Keystore File (.JKS)

Follow the steps below to create a self-signed certificate for SSL and JWT, or to create keys, or to generate a keystore file for the certificate and keys. These scripts are only available from version 7.0.1 and above:

1. Navigate to the directory *<SAG installation>/AdabasRestAdministratrion/keys/scripts*. This directory contains the sample config file (*csr_config.cnf*) and the script (*generate_jks.sh*) used to generate the self-signed certificate, keys and keystore file.

2. The config file *csr_config.cnf* is shown below. Adapt this file for the domain and infrastructure in your destination environment. For example, `example.com` must be replaced with your actual DNS names.

```
[ req ]
default_bits = 2048
default_keyfile = test_privatekey.pem
distinguished_name = req_distinguished_name
encrypt_key = no
prompt = no
string_mask = nombstr
req_extensions = v3_req

[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = digitalSignature, keyEncipherment, dataEncipherment
extendedKeyUsage = serverAuth, clientAuth
subjectAltName = DNS:test.example.com, DNS:*.test.example.com

[ req_distinguished_name ]
countryName = DE
stateOrProvinceName = Hessen
localityName = Locale Test
O.organizationName = Test Certificate
organizationalUnitName = Evaluation
commonName = test.example.com
```

3. Create the *keyfile.jks* keystore file by running the command:

```
> generate_jks.sh
```

The contents of the script *generate_jks.sh* are shown below and they can be adapted to suit your needs:

```
# Generate self signed certificate and keys dependent on the csr_config.conf ↵
input file
openssl req -new -x509 -nodes -days 365 -sha256 -newkey rsa:4096 -keyout key.pem ↵
-out certificate.pem -config csr_config.cnf
openssl x509 -text -noout -in certificate.pem
openssl pkcs12 -password pass:test123 -inkey key.pem -in certificate.pem -export ↵
-out certificate.p12
openssl pkcs12 -password pass:test123 -in certificate.p12 -noout -info

# Generate keystore file based on the p12 certificate
keytool -importkeystore -keypass test123 -srcstorepass test123 -srckeystore ↵
certificate.p12 -srcstoretype jks -destkeystore keystore.jks -deststoretype pkcs12 ↵
-deststorepass test123
```

4. Copy the generated keystore file *keyfile.jks* to the location defined in the configuration. The default location is the *keys* directory of the installation.

5 Adding Support for Multiple Adabas Versions

This version of Adabas RESTful server supports administration of multiple Adabas versions. Currently, Adabas Version 6.5 with internal structure level 19 and Adabas Version 6.6 with internal structure level 20 are supported. To support additional Adabas versions the relevant Adabas version installations need to be registered in the configuration of the RESTful server.

In order to register the Adabas version installation in *config.xml*, the *service.sh* script (or *service.bat* respectively) has an additional feature. With the following command, you may add the current Adabas installation, if you are in an Adabas command prompt shell:

```
service.bat add_env
```

To register an Adabas version installation outside of the current environment, you can add the corresponding directory:

```
service.bat add_env <Adabas version installation directory>
```

6 System Service

The Adabas RESTful administration provides a system service for starting the daemon during startup. The service is started with a specific user, and all database containers, configuration files and log files are created with this user.

The system service scripts provide the possibility to start, stop, remove and install the system service. The script is called *system_service.sh* (on Unix) and *system_service.bat* (on Windows).

Please be aware that the user who starts the Adabas RESTful server will be the user that works on the offline methods on the database container. Online activities are performed using the login id that is used to connect to the Adabas RESTful server. This user id is displayed in the user queue of the database.

Unix System Service

On Unix platforms, the Adabas RESTful service is added to the Unix system startup during installation. The service is started with the installation user.

Windows System Service

On Windows platforms, the security system needs to be ignored. The user who starts the Adabas RESTful administration does not have to have administration rights. Because of Windows security, the container might be created without permission for other Software AG administration users. It is recommended to use a normal user instead of a user with Administration rights.

During the installation the system service is not installed automatically. You need to start the *system_service.bat* script to create the service using a specific user.

```
C:\SoftwareAG\AdabasRestAdministration\bin>system_service.bat install
wrapperm | Prompting for account (DOMAIN\ACCOUNT)...
Please input the domain name [DMN]:
Please input the account name [sag]:
Please input the password for account 'DMN\sag': ************
wrapperm | Software AG Adabas REST Server(16) service installed.
wrapperm | Starting the Software AG Adabas REST Server(16) service...
wrapperm | Software AG Adabas REST Server(16) service started.
```

In order to use the Adabas RESTful administration, the Adabas client global environment configuration needs to be set.

7   Messages

The following response codes are returned if errors occur while processing.

Administration Error Codes

General Error Messages

PAIF00001	**AIF command execution error, got response code {0}**
Action	Check the error log and the platform. Contact your nearest support centre for further information.

PAIF00008	**AIF environment incorrect**
Action	The Adabas interface is not supported, Upgrade to the most recent version.

PAIF00009	**Buffer transfer error**
Action	Provide a correct buffer.

PAIF00010	**AIF environment not registered**
Action	The requested Adabas installation path is not correct. Provide the correct installation path.

PAIF00012	**Software AG environment null given**
Action	An incorrect path name was given. Provide the correct path name.

PAIF00020	**Execution exception: {0}**
Action	Contact your nearest support centre for further information.

PAIF00021	**Unsupported request**
Action	Unsupport RESTful API request. Check the request.

PAIF00022	Monitor file {0} invalid
Action	No correct monitor file found. Check the file number of the monitor file.

Error Messages from the REST Server

EAIF00002	Database name too long
Action	The given database name is too long. Provide a correct database name.

EAIF00003	Database file name too long
Action	The given database file name is too long. Provide a correct database file name.

EAIF00006	No Adabas environment found
Action	No Adabas environment is found. Define Adabas environments using the configuration file.

EAIF00007	Invalid Adabas database operation given: {0} ({1})
Action	An unknown error during database operations occurred. Check the output, and contact your nearest support centre for further information.

EAIF00009	Adabas database id {0} is not in range of [{1}:{2}]
Action	The given database id is not in the valid range of databases between 1 and 255.

EAIF00010	Adabas field reference missing
Action	Invalid parameter REST API reference for the file operation.

EAIF00011	Adabas resource reference {0} is not valid
Action	The given Adabas resource request is unknown. Check the URL.

EAIF00012	Structure level {1} for environment in {0} not supported
Action	The given structure level is not supported by Adabas REST API.

EAIF00013	**Adabas environment path error for {0}**
Action	The path of the Adabas environment does not exist. Check the suite path or the Adabas program directory.

EAIF00014	**Adabas environment path {0} already registered**
Action	The given Adabas path is already registered.

EAIF00017	**DELETE http method not valid for resource {0}**
Action	It is not allowed to delete the given REST API resources.

EAIF00018	**Invalid DELETE http method request**
Action	The given REST API resource is not known.

EAIF00019	**POST http method not valid for resource {0}**
Action	It is not allowed tp post the given REST API resource.

EAIF00020	**Invalid POST http method request**
Action	The given REST API resource is not known.

EAIF00021	**System file {0} on database id {1} cannot be renamed**
Action	It is not allowed to rename Adabas system files.

EAIF00022	**Adabas offline tool evaluation for Adabas installation at {0} failed**
Action	Problems occurred during the initialization of the offline Adabas interface. Contact your nearest support centre for further information.

EAIF00023	**Invalid Adabas parameter provided: {0}**
Action	Unknown Adabas parameter given.

EAIF00024	**Adabas parameter request need type of modification (dynamic or static)**
Action	The Adabas parameter setting REST API needs type information. Provide the type parameter for the REST API.

EAIF00025   **Adabas parameter only dynamic or static type of modification allowed**

Action   The incorrect type value is given for the Adabas REST API setting Adabas parameter.

EAIF00026   **Adabas options invalid: {0}**

Action   The given option value is incorrect.

EAIF00027   **Either UTILITIES_ONLY or LOCAL_UTILITIES can be set in dynamic mode, not both together**

Action   The given option UTILITIES_ONLY or LOCAL_UTILITIES are mutually exclusive.

Error Messages from Adabas

AIF00000   **Normal successful completion**

Action   None.

AIF00001   **This return code indicates an unspecified error 1**

Action   Internal error, please send the exception and trace log to your nearest support centre for detailed information.

AIF00002   **There is no space left**

Action   There is no disk space is left, add additional disk space.

AIF00003   **Not found**

Action   The given Adabas database is not available.

AIF00004   **The open operation failed**

Action   Could not open ADABAS.INI. Check the ADADATADIR configuration.

AIF00007   **A syntax error occurred**

Action   Invalid utility option specified for crdemodb.

AIF00009	**An environment error occurred**
Action	The Adabas environment is not set correctly.

AIF00010	**A structure level mismatch was detected**
Action	The version of the Adabas interface does not match the Adabas database structure level.

AIF00011	**Pending Autorestart**
Action	The database is in a state that an AUTORESTART needs to be performed or is currently being performed.

AIF00013	**No file was specified**
Action	Adabas operations are not allowed on a LOB file. Use a regular Adabas file.

AIF00014	**The FDT is too big**
Action	The number of Adabas fields for the new FDT is too high.

AIF00015	**ISN reusage is not permitted**
Action	Reusage operations are not allowed on system files. Use a regular Adabas file.

AIF00016	**Cyphering is not allowed**
Action	Adabas ciphering is not allowed on system files. Use a regular Adabas file.

AIF00017	**An ADAM error occurred, for example, files with the ADAM feature cannot be refreshed**
Action	ADAM files cannot be refreshed. Use a regular Adabas file.

AIF00018	**A value to be returned exceeded the maximum possible value**
Action	The given parameter value exceeds the maximum possible value.

AIF00035　　　　　**An internal error occurred when trying to retrieve the FCB**

Action　　　　　An internal error occurred. Send the exception and the trace log to your nearest support centre for detailed information.

AIF00036　　　　　**The container already exists**

Action　　　　　The given Adabas container already exists.

AIF00037　　　　　**The database already exists**

Action　　　　　The given Adabas database id already exists.

AIF00038　　　　　**An internal error occurred when trying to retrieve the GCB**

Action　　　　　Access to the GCB failed.

AIF00039　　　　　**An internal error occurred when trying to retrieve the FST**

Action　　　　　An internal error has occurred. Please send the exception and trace log to your nearest support centre for detailed information.

AIF00041　　　　　**An internal error occurred when mapping to CSA**

Action　　　　　Access to the CSA failed.

AIF00042　　　　　**Internal error when trying to set security mode**

Action　　　　　An error occurred while setting the security mode of the Adabas database.

AIF00148　　　　　**The database not active**

Action　　　　　The given Adabas database is not active. Start the database.

AIF00149　　　　　**The database is active**

Action　　　　　An Adabas database container can only be deleted if the database is offline.

AIF00151　　　　　**The file is locked**

Action　　　　　The given Adabas file is locked for Adabas utility operations only. Unlock the file.

AIF00152	**Invalid parameters**
Action	A given parameter is invalid. Correct the invalid parameter.
AIF00153	**The database ID is invalid**
Action	The given database ID is not valid in the used context.
AIF00154	**The file number is invalid**
Action	The given database file number is not valid in the used context.
AIF00156	**No file was specified**
Action	A base file, used for adding a lob file, is missing.
AIF00157	**The same file number is not allowed for the file and for the LOB file**
Action	The given base file number and lob file number are the same. Use different file numbers.
AIF00158	**The database is read-only**
Action	The given Adabas database is read-only. Change the read-only state.
AIF00159	**Open ASSO**
Action	Failed to read records from the ASSO container.
AIF00160	**There is insufficent memory**
Action	There is not enough container space to load the Adabas file.
AIF00161	**No permission**
Action	There is no permission to create Adabas containers.
AIF00162	**The File(s)/Userid are not available at open**
Action	The file is either already available or cannot be created.

| AIF00163 | **The LOB File(s) are not available at open** |
| Action | The file is either already available or cannot be created. |

| AIF00164 | **A Utility Communicaton Block (UCB) overflow has occurred** |
| Action | Cannot create a UCB entry. The number of entries has been exceed. |

| AIF00166 | **The file is not already loaded** |
| Action | The base Adabas file not available. |

| AIF00167 | **Failed to read the FCB** |
| Action | Not able to read the Adabas file control block. |

| AIF00168 | **There is no space left for the AC** |
| Action | There is no space available in the Adabas ASSO containers. |

| AIF00169 | **There is no space left for the DS** |
| Action | There is no space available in the Adabas DATA containers. |

| AIF00171 | **There is no space left for the NI** |
| Action | There is no space available in the Adabas normal index. |

| AIF00172 | **There is no space left for the UI** |
| Action | There is no space available in the Adabas upper index. |

| AIF00173 | **Failed to read the UCB** |
| Action | Failed to read the UCB entries. |

| AIF00174 | **Failed to read the ASSO block** |
| Action | Failed to read ASSO blocks. |

AIF00179	**Failed to write the ASSO block**
Action	Failed to write the ASSO block.

AIF00180	**Failed to open the DATA file**
Action	Failed to open the DATA file.

AIF00181	**Failed to lock the database semaphore**
Action	Failed to lock the database container before the change.

AIF00183	**A shutdown or cancel is in progress**
Action	A database shutdown or cancel is in progress.

AIF00184	**Access denied**
Action	Access to delete Adabas resources is denied.

AIF00187	**The ASSO storage is exhausted**
Action	The ASSO storage space is not available for the file operation.

AIF00190	**The DATA storage is exhausted**
Action	The available DATA space storage is exhausted.

AIF00195	**There are too many ADAM overflow blocks**
Action	There are too many ADAM blocks used.

AIF00196	**An invalid RABN was specified**
Action	An internal error has occurred, contact your nearest support centre for further information.

AIF00197	**STOP/TERMINATE detected**
Action	A stop or terminate operation has been detected during processing.

AIF00201   **Create database/container failed**

Action   An unknown error has occurred, contact your nearest support centre for further information.

AIF00202   **An invalid block size specified/block size is out of range**

Action   Invalid block size given, use a correct block size.

AIF00203   **Delete database/container failed**

Action   Could not delete the Adabas container. Check the file permissions.

AIF00204   **Failed to open the WORK file**

Action   Not able to access the WORK file.

AIF00205   **Failed to write entry in checkpoint file**

Action   Failed to write the checkpoint entry.

AIF00206   **Container to be deleted is still in use**

Action   The Adabas container to be deleted is still used by Adabas data.

AIF00207   **No space left on device**

Action   There is not enough space to create the container on the file system or RAW device.

AIF00208   **No such file or directory**

Action   The given RAW device name is not correct.

AIF00209   **The given path is neither empty nor full qualified.**

Action   The given path for ADADATADIR or for the container ASSOx or DATAx is not correct.

AIF00300   **The ADAPROGDIR environment variable could not be found**

Action   The given ADAPROGDIR configuration/environment variable is not correct.

AIF00301	**No error handler was defined**
Action	An internal error has occurred, please contact your nearest support centre for further information.

AIF00302	**Opening the INI file failed**
Action	Either ADABAS.INI or DB<dbid>.INI cannot be found.

AIF00303	**The ADADATADIR environment variable could not be found**
Action	The given ADADATADIR configuration/environment variable is not correct.

AIF00304	**The ADAVERS environment variable could not be found**
Action	ADAVERS is not set correctly in the installation, or the configuration is incorrect.

AIF00305	**Read INI file failed**
Action	Reading the INI files failed.

AIF00306	**Missing or inconsistent entries in INI file**
Action	Configurations in the INI files are missing or incorrect.

AIF00307	**Failed to set value in INI file.**
Action	Setting a value in the INI file fail.

AIF00308	**Failed to write INI file.**
Action	Writing to the INI file failed.

AIF00309	**Database not found in ADABAS.INI**
Action	The Adabas database ID is not registered in ADABAS.INI. Register the database ID.

AIF00310	**No structure level found for the given DBID in ADABAS.INI**
Action	The entry for the structure level is missing.

AIF00400	**Incompatible format conversion or truncation error**
Action	An internal error has occurred. Contact your nearest support centre for further information.

AIF00502	**The given file was not found**
Action	Cannot find the given Adabas file.

AIF00503	**The given file is a system file**
Action	Calls to Adabas system files are not allowed.

AIF00504	**The given file is a LOB file**
Action	An operation was requested on an Adabas LOB file, which is not allowed.

AIF00507	**This operation is not allowed online**
Action	The Adabas file option BT cannot be set if Adabas is online.

AIF00511	**The given file is not lob file**
Action	The Adabas file does not have a LOB file defined. Define the LOB file.

AIF00512	**Record spanning not allowed**
Action	Adabas file record spanning can only abe disabled if the Adabas file is empty.

AIF00600	**The string length is greater than the amount of allocated memory**
Action	The length of the string exceeds the maximum limit.

AIF00651	**Failed to launch ADASTART, examine Adabas nucleus log for details**
Action	The start of the Adabas nucleus failed. Check the nucleus log.

AIF00800	**Failed to allocate memory**
Action	Failed to allocate memory.

AIF00801	**Failed to reallocate memory**	
Action	Failed to reallocate memory.	
AIF00850	**License file was not found**	
Action	The AEL license file was not found.	
AIF00863	**A write error has occurred**	
Action	Could not update the INI files. Check the file permissions.	
AIF00900	**The export copy for the demo files was not found**	
Action	The example ordexp file is not available.	
AIF00901	**Loading the demo files failed**	
Action	The Adabas utilityused to import demo data returned an error.	
AIF01000	**A referential integrity error has occurred**	
Action	An error has occurred because of a referential integrity error.	
AIF01002	**RI not with system file**	
Action	Referential integrity is not allowed for system files.	
AIF01016	**A reference to a missing file was detected - remove the referential integrity**	
Action	The Adabas referential integrity file is not available.	
AIF01100	**user exits 1 and 11 are mutually exclusive**	
Action	User exits 1 and 11 are mutually exclusive.	
AIF02000	**Could not create assign file.**	
Action	An error occurred creating the assign file.	

AIF02001	**Could not delete assign file.**
Action	An error occurred deleting the assign file.

AIF02002	**Could not close assign file.**
Action	The file status is unknown.

AIF02003	**Assign file is not open for access.**
Action	An internal error has occurred. Contact your nearest support centre for further information.

AIF02004	**Assign file does not exist.**
Action	The corresponding assign file does not exist.

AIF03000	**Failed to initialize cryptography subsystem**
Action	Failed to initialize the cryptography subsystem. Check the output.

AIF03001	**Key reference value mismatch - wrong key in use**
Action	The given encryption key does not match to the container encryption.

AIF09999	**This return code indicates an internal error**
Action	An internal error has occurred. Contact your nearest support centre for further information.

AIF10<rso>	**Corresponding Adabas response code**
Action	Please refer to the Adabas response code documentation for further information. Contact your nearest support centre.

AIF20001	**Container reference identifier wrong: {0}**
Action	The referenced Adabas container identifier is wrong.

AIF20002   **Container reference identifier missing**

Action   The referenced Adabas container type is wrong.

AIF20003   **Required request parameter {0} missing**

Action   Adabas file operations missing required parameter.