Adabas for Linux and Cloud

Adabas REST Administration

Version 7.3.0

February 2025

ADABAS & NATURAL

This document applies to Adabas for Linux and Cloud Version 7.3.0 and all subsequent releases.

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

Document ID: ADAOS-RESTFUL-730-20250228

Table of Contents

Preface v
1 About this Documentation 1
Document Conventions 2
Online Information and Support 2
Data Protection 3
2 General Information 5
Adabas Administration and Monitor Handler 6
Job Control 6
File Transfer Handler 6
Technical Insight 7
Adabas Batch Administration 7
Merge Adabas RESTful Server 10
3 Installation and First Steps 11
Password Management 12
Adabas Installation Management 12
Changing the Location of the Adabas Data Directory 13
Starting the Server 13
Ensuring that the Server is Working 14
4 RESTful Security 15
Secure connection with SSL 16
Adabas User Authentication 16
Adabas User Authorization 17
JSON Web Token 20
Creating a Keystore File (.JKS) 20
5 Adding Support for Multiple Adabas Versions 23
Enable different Adabas version of databases shown in Adabas Manager 24
6 Adabas REST Server Configuration 25
Prerequisites 26
Configuration 26
Runtime 28
Authentication 28
7 System Service 31
Linux System Service 32
Windows System Service 32
8 Messages 33
Administration Error Codes 34
System Component Error Messages 57

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.
Adabas REST Server Configuration	Describes how to configure the Adabas REST server before you use it for the first time.
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 2
- Online Information and Support 2
- Data Protection 3

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://documentation.softwareag.com**.

Product Training

You can find helpful product training material on our Learning Portal at **https://learn.softwareag.com**.

Tech Community

You can collaborate with Software GmbH experts on our Tech Community website at **https://techcommunity.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 GmbH news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at **https://github.com/softwareag** and **https://containers.softwareag.com/products** and discover additional Software GmbH resources.

Product Support

Support for Software GmbH 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 GmbH 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 Administration and Monitor Handler 6
- Job Control 6
- File Transfer Handler 6
- Technical Insight 7
- Adabas Batch Administration 7
- Merge Adabas RESTful Server 10

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

- Password Management 12
- Adabas Installation Management 12
- Changing the Location of the Adabas Data Directory 13
- Starting the Server 13
- Ensuring that the Server is Working 14

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, which is managed by the configuration file *realm.properties*, which contains a list of users and passwords.

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.

Changing the Location of the Adabas Data Directory

By default, the Adabas Data directory (ADADATADIR) is the Adabas installation directory. You can change the default location of the Adabas Data directory with the `change_adadatadir.sh` or `change_adadatadir.bat` command. For details on how to change the Adabas Data directory, see *Installation on Windows* > *Completing the Installation* or *Installation on Linux* > *Completing the Installation*.

You can then modify the location of the Adabas Data directory used by the Adabas REST server in the REST server configuration with the following service script:

```
./service.sh data_dir <new Adabas Data directory location>
```

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

- Secure connection with SSL 16
- Adabas User Authentication 16
- Adabas User Authorization 17
- JSON Web Token 20
- Creating a Keystore File (.JKS) 20

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 the following type of user authentication:

- User and password file called Realm.

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

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_userAdding user to password file ...
2022-09-20 08:40:20 Adabas REST client version v7.1.1.0
2022-09-20 08:40:20 Trace 'info' log send to ↩
/opt/softwareag/AdabasRestAdministration/logs/client.log
Enter username: sag2
Enter password: manage
User sag2 added to realm file
2022-09-20 08:40:42 client ended
```

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

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

The *service.sh* (for Linux platform) or *service.bat* (for Windows platform) script should be used to add Administration permissions. The script is located in *<install-dir>/AdabasRestAdministration/bin*

```
service.bat add_admin admin_name
```

where `admin_name` represents the user to be added as an administrator role.

The list of Administrators with permissions are located in the file *<install-dir>/AdabasRestAdministration/configuration/administration.xml*.

The file is auto-generated, containing usernames and generated checksum. It should not be edited manually.

In the example below, the users `admin` and `sag` have Administration permissions:

```
<Users>
   <User name="admin" checksum=<generated-checksum> />
   <User name="sag" checksum=<generated-checksum> />
</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.

Provide Database Modification Permission

To restrict access for any modification made on the database, set the `NoModification` configuration parameter in *config.xml*. This parameter enables or restricts access to create and delete any database.

The following example enables the permission to create and delete any database:

```
<Module>
  <Admin NoModification="false"></Admin>
</Module>
```

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

- Enable different Adabas version of databases shown in Adabas Manager 24

This version of Adabas RESTful server supports administration of multiple Adabas versions. Currently, Adabas Version 7.2 with internal structure level 24 and Adabas Version 7.1 with internal structure level 23 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>
```

Enable different Adabas version of databases shown in Adabas Manager

Option 1:

1. Install the Adabas RESTful server together with each different Adabas version installations with different port number.
2. Setup the Adabas Manager with 2 different port number in "Host Config".

Option 2:

1. Only need to install 1 Adabas RESTful server to handle different Adabas version
2. Setup each of the Adabas installed ADADATADIR to a common directory

   ```
   ./change_adadatadir.sh <New ADADATADIR>
   ```

3. Setup the Adabas RESTful server configuration

   ```
   ./service.sh data_dir <New ADADATADIR>
   ```

4. Restart the Adabas RESTful server

6 Adabas REST Server Configuration

- Prerequisites 26
- Configuration 26
- Runtime 28
- Authentication 28

The Adabas REST Interface is a sub-component of the Adabas installation. It is used to access data from any programming language that supports HTTP requests and to request and monitor Adabas administration tasks.

You must install The Adabas REST Interface separately from the Adabas product installation.

This chapter is organized as follows:

Prerequisites

The Adabas REST server is initially installed as a service task. To manually uninstall or install the service task, use either the *system_service.sh* or *system_service.bat* script.

To configure the server, you must first access an Adabas database using one of the following methods.

To connect to an Adabas database with	Use
Adabas TCP/IP	*dbid*(adatcp://*host*:*port*)
Entire Net-Work	*dbid*(tcpip://*host*:*port*) To access a remote Adabas database via Entire Net-Work, configure the corresponding Software AG Directory Server.
local Adabas calls	*dbid*(adatcp://*host*:0)

Configuration

The server is configured during the Adabas RESTful administration installation process. The configuration parameters are defined in *config.xml*. This file is located in the configuration subdirectory of the Adabas RESTful administration installation. The configuration file looks as follows:

```
<RestServer>
    <Server>
        <Content directory="examples" />
        <Service port="8190" type="http" />
        <Service port="8191" type="https">
            <KeyStore file="keys/keystore.jks" />
            <KeyPassword password="test123" />
        </Service>
        <LoginService module="" webTokenExpires="24">
            <Administrators ↩
file="${SAG}/configuration/administrator.xml"></Administrators>
            <Users file="${SAG}/configuration/users.xml"></Users>
            <AuthenticationServer module="" type="file">
```

```
                <Realm file="${SAG}/configuration/realm.properties"></Realm>
            </AuthenticationServer>
        </LoginService>
        <Shutdown passCode="shut123" />
    </Server>
    <Directory url="file:xtsurl.cfg" />
    <Mapping>
        <Database dbid="24" file="4" />
        <Database dbid="23" file="250" />
        <Database dbid="100(adatcp://databasehost:61100)" file="250" />
    </Mapping>
    <DatabaseAccess global="false">
        <Database dbid="100" />
    </DatabaseAccess>
</RestServer>
```

`<Server>` contains the basic attributes for running the REST server:

- `<Content>` defines the directory of the static HTML files used to provide the REST service. The example subdirectory provides an example application to help you learn more about accessing Adabas using the REST interface.
- `<Service>` sets the TCP/IP listening port for HTTP and HTTPS requests. The access URL for the example application looks as follows: `https://localhost:/`*`port`*.

  For HTTPS access, define the `<KeyStore>` and `<KeyPassword>` attributes to set the necessary SSL certificates.

  Note: The provided *keystore.jks* file is only an example keystore with self-signed certificates. You must not use them in a production environment.

- `<LoginService>` defines the authentication method used by the REST server.

  The possible values for `type` are: `file`, `pam` (using system login for UNIX), and `system` (using system login for Windows).
- `<Shutdown>` defines the password for the shutdown command of the REST server. This password is an additional validation measure against unauthorized shutdown of the server. You can shut down the server manually with the `service.sh stop` script.

`<Directory>` sets the reference to the Software AG Directory Server. All directory information required to accomplish communication between clients and servers is obtained from the Directory Server. A file reference such as `file:`*`directory`*`/xtsurl.cfg` points to a file which contains the access URLs of remote databases. A URL reference such as `tcpip://`*`host`*`:4952` points to the Directory Server URL.

`<Mapping>` connects the unique Adabas map definitions to the REST service. The Adabas Client for Java uses map definitions to map database names and long name fields to short names. Use Natural DDMs to define the long names. To create long field names, use the Adabas Data Designer.

The Adabas data maps are stored in an Adabas file. Each database can contain multiple mapping files, but at least one mapping file must define the long name references.

`<DatabaseAccess>` determines whether direct database references are allowed. Direct database references use the Adabas database ID to request Adabas data. The two valid attributes for `<Database>` are: `dbid` (to set the database ID) and `url` (to set an access path to a remote database). If the `global` attribute is set to `true`, all known local databases can be accessed directly. If `global` is set to `false`, you will not be able to access the database directly using http://*rest server*:*port*/rest/db/*dbid*. You must explicitly enable access by setting the database ID using the `Databse` configuration.

Runtime

The Adabas RESTful server is part of the system service startup process.

The *wrapper.log* file in the log directory stores the output of the server log. The *server.log* file stores trace and debug information.

You can also start the server manually with the `service.sh run` script on UNIX/Linux and `service.bat run` on Windows platforms. The manual start opens a console window where the REST server generates the output.

The REST server example page can be accessed via `http://localhost:`*port* or `https://localhost:`*port*. The default user name is "admin" with the password "manage". Change the password or remove the default user password of the admin from the *realm.properties file*. You can add new users with the *service.sh* or *service.bat* script.

Authentication

The Adabas RESTful server supports file-based authentication.

The security.conf file configures the JAAS security classes used to set up the authorization. The following modules are predefined in the delivered configuration:

Module	Description
Adabas	Create file-based authorization with the *realm.properties* file.
LocalUnix	Use the Software AG local access security library (SSX) to enable system authentication.
LocalWindows	Use the local system authentication.

- Adabas
- UNIX Platforms

- Windows Platforms

Adabas

With the Adabas module, you can create file-based authorization with the *realm.properties* file. The REST server supports a realm with minimal authentication.

In order to set up authentication with MD5 or SHASUM hash encoding, modify the *realm.properties* file from the configuration subdirectory. This file contains the username and password information. By default, the user is "admin" with password "manage".

The format of the password file is as follows: `user name:user password, roles of the user`. The user password can be prefixed with the hashsum MD5: or SHA: algorithm name, which represent the md5sum or shasum hash of the password.

For example:

```
Administrator: MD5:70682896e24287b0476eff2a14c148f0, sagadmin, jobadmin
sag:  MD5:20384856e54267b7488eefea1ac1a8fa, saguser
user: MD5:d47f18dc7780fe47c24759714e2cd58f, saguser
```

The user roles are defined in the configuration file. For example, you can define the role for job control as follows:

```
<Job role="jobadmin" use_role="false" />
```

This configuration can also apply to file upload and download rights for `<Directory>` and for controlling administration tasks with `<Admin>`.

To add a new user to *realm.properties*, start the `server.<sh or bat> add_user` command, and enter the new user and password.

UNIX Platforms

The Software AG SSX modules use the Pluggable Authentication Module (PAM) to authenticate the username and password on UNIX.

For detailed information on how to use PAM, see *Software AG Security eXtensions Administrator's Guide > Using the Pluggable Authentication Module (PAM) on UNIX*.

To configure PAM with Adabas REST service, add the PAM configuration file in config.xml for `module`:

```
<AuthenticationServer module="file" type="pam"></AuthenticationServer>
```

Where “file” represents the PAM configuration file to use.

For example:

```
<AuthenticationServer module="login" type="pam"></AuthenticationServer>
```

If AuthenticationServer module is an empty string, the default built-in login configuration file will be used.

Windows Platforms

For local Windows authentification, use the Waffle Windows Authentication Framework.

7 System Service

- Linux System Service 32
- Windows System Service 32

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 Linux) 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.

Linux System Service

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

Windows System Service

On Windows platforms, the user who starts the Adabas RESTful administration must have administration rights. Run the command prompt as administrator and run the `system_service.sh or bat` command.

```
C:\SoftwareAG\AdabasRestAdministration\bin>system_service.bat install
"START_DIRECTORY: C:\SoftwareAG\AdabasRestAdministration\bin\"
"SAG           : C:\SoftwareAG\AdabasRestAdministration\bin\\..\.."
2022-09-20 10:51:31 Adabas Operator RESTful service version v7.1.1.0
2022-09-20 10:51:31 Build date=20-09-2022_08:14:35
2022-09-20 10:51:31 Logging to file ↩
winfile:///C:\SoftwareAG\AdabasRestAdministration\bin\\..\..\AdabasRestAdministration\logs\service.log
2022-09-20 10:51:32 Service name : Software AG Adabas REST server v7.1.1.0 (123)
2022-09-20 10:51:32 Install service: Software AG Adabas REST server v7.1.1.0 (123)
2022-09-20 10:51:32 Service path   : C:\SoftwareAG
2022-09-20 10:51:32 Service user   :
2022-09-20 10:51:32 Service group  : None
2022-09-20 10:51:32 Service connected
2022-09-20 10:51:32 Install service successfully: Software AG Adabas REST server ↩
v7.1.1.0 (123)
2022-09-20 10:51:32 Service command finished
```

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

8 Messages

- Administration Error Codes 34
- System Component Error Messages 57

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

Administration Error Codes

- General Error Messages
- Error Messages from the REST Server
- Error Messages from Adabas

General Error Messages

PAIF00001 **Adabas interface**

Action Check log and platform, contact support.

PAIF00008 **Adabas interface check**

Action Adabas interface is not supported, upgrade version.

PAIF00009 **Buffer check**

Action Provide correct buffer.

PAIF00010 **Adabas product installation environment**

Action Requested Adabas installation path is not correct, provide correct one.

PAIF00012 **Software AG installation environment evaluation**

Action Incorrect path name given.

PAIF00020 **Unknown exception thrown inside application**

Action Contact support for further evaluation.

PAIF00021 **Adabas RESTful API**

Action Unsupport RESTful API request, check request.

PAIF00023 **Parsing of FDT field**

Action Unknown error during parsing of FDT field occurs. Check output and contact support.

PAIF00024 **Send request for all files**

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

PAIF00025 **Missing SAG environment directory**

Action No Adabas environment is found. Define Adabas environment (ADAPROGDIR) directory.

PAIF00026 **Invalid directory for SAG environment**

Action No Adabas environment is found. Define valid Adabas environment (ADAPROGDIR) directory.

PAIF00027 **Adabas ADADATADIR environment path is not defined**

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

PAIF00028 **Unknown FEOF type given**

Action Use valid keyword for FEOF type.

PAIF00029 **Unknown reset type given**

Action Use valid keyword for Reset type.

PAIF00030 **Interface error, failed to create Adabas admin instance**

Action Provide a valid atabase target.

PAIF00031 **Existing SAG environment already registered**

Action The SAG environment is already registered.

PAIF00032 **Invalid buffer for database creation**

Action Unknown error during creation of database. Check output and contact support.

PAIF00033 **Encrypted database creation with AIF structure level**

Action The given structure level is not supported for encryption database usage.

PAIF00034 **AIF request error**

Action Unknown error during sending AIF request. Check output and contact support.

PAIF00035 **RBAC operation is invalid**

Action Unknown RBAC operation is used, only valid operation are allowed.

PAIF00036 **Failed to read ADABAS.INI file from path in configuration**

Action Check for valid ADABAS.INI file and provide valid `AdabasData` location path in the configuration file.

PAIF00037 **Failed to read DB{0}.INI file from path in configuration**

Action Check for valid DB{0}.INI file and provide valid `AdabasData` location path in the configuration file.

PAIF00038 **Invalid Adabas utility provided**

Action The Adabas utility specified does not exist. Please use a valid Adabas utility.

PAIF00100 **The specified RBAC role does not exist**

Action Please specify an existing RBAC role or create the role.

PAIF00101 **The specified RBAC user does not exist**

Action Please specify an existing RBAC user or create the user.

PAIF00102 **The specified RBAC operation does not exist**

Action Please specify a valid RBAC operation.

PAIF00103 **The specified RBAC object does not exist**

Action Please specify a valid RBAC object.

PAIF00104 **The specified RBAC entity does not exist**

Action Please specify a valid RBAC entity.

PAIF00105 **The specified RBAC permission for user role does not exist**

Action Please specify a valid user role permission.

PAIF00106 **The specified RBAC permission for user role does not exist**

Action Please specify a valid permission.

PAIF00107 **The specified RBAC role already exists**

Action Please specify a new role to create.

PAIF00108 **The specified RBAC user already exists**

Action Please specify a new user to create.

PAIF00109 **The specified RBAC operation already exists**

Action Please specify a new operation to create.

PAIF00110 **The specified RBAC object already exists**

Action Please specify a new object to create.

PAIF00111 **The RBAC user-role relationship already exists**

Action Please specify a new permission to grant.

PAIF00112 **The RBAC permission already exists**

Action Please specify a new permission to grant.

PAIF00113 **Operation is not allowed on a read-only database**

Action Please adapt the database permissions.

PAIF10248 **Internal error to parse invalid parameter info**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10254 **Invalid Parameter in parameter info**

Action Provide only valid parameter info.

PAIF10256 **Failed to locate aifcommand in ADAPROGDIR**

Action Check for valid aifcommand in ADAPROGDIR location. Contact your nearest support centre for further information.

PAIF10257 **Failed to parse binary init of environment**

Action Unknown error has occurred. Contact your nearest support centre for further information.

PAIF10258 **No database file entry found**

Action No Adabas file control block found

PAIF10259 **Request return error:**

Action Unknown request return error. Check output and contact support

PAIF10260 **Error encountered to delete database file**

Action Check error description.

PAIF10261 **Error to renumber file from to {1}**

Action Check if file number {1} exists.

PAIF10262 **Error to refresh file**

Action Check if file number exists.

PAIF10263 **Request for database file returned error**

Action Unknown error in request response. Check output and contact your nearest support centre for further information.

PAIF10264 **Incorrect container type used**

Action Provide a valid container type (ASSO / DATA) in the request

PAIF10265 **Database name too long**

Action The given database name is too long. Provide a shorted database name, less than 16 characters

PAIF10267 **Nucleus log file is not created**

Action Check for valid nucleus log file and contact your nearest support centre for further information.

PAIF10268 **Operation is not known**

Action Invalid database operation used. Provide only valid database operation

PAIF10269 **Failed to get next free database ID**

Action Unknown error in request response. Contact your nearest support centre for further information.

PAIF10270 **Internal error, Adabas admin handler missing**

Action Missing Adabas admin handler. Check output and contact your nearest support centre for further information.

PAIF10271 **Error evaluating Adabas option:**

Action Provide a valid option keyword, please see Adabas documentation for valid option keywords.

PAIF10272 **Value does not fit into parameter**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10273 **Parameter not valid:**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10274 **Parameter extent not valid:**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10275 **Error evaluating parameter:**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10276 **Error parsing parameter buffer**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10277 **Error sending parameter request**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF10278 **Invalid Adabas parameter provided:**

Action Check for invalid Adabas parameter used. Please use only valid parameter keywords

PAIF11000 **Cannot convert type to uint64:**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF11001 **PAIF11001=Cannot convert type to uint32:**

Action An internal error has occurred. Contact your nearest support centre for further information.

PAIF11002	**PAIF11002=Cannot convert type to byte:**
Action	An internal error has occurred. Contact your nearest support centre for further information.
PAIF90001	**Unknown error:**
Action	Check response output and contact your nearest support centre for further information.

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.

EAIF00028 **Input update/insert tag 'Store' or 'Records' missing**

Action Internal error to update/insert tag. Check output and contact your nearest support centre for further information.

EAIF00029 **Input text/plain cannot converted to type**

Action Internal error for type conversion. Check output and contact your nearest support centre for further information.

EAIF00030 **Isn or search need to be given**

Action No Isn or search input provided. Please provide the isn or search input parameter

EAIF00031 **No records received from query**

Action Check for valid Data 'Store' or 'Records' provided.

EAIF00032 **No records found**

Action No records found in the read request.

EAIF00036	**Passed dbid not found**
Action	The Database ID provided is not found
EAIF00037	**Invalid database id reference:**
Action	The Database ID reference provided is invalid
EAIF00038	**Reference '{0}' not allowed**
Action	Reference is not allowed. Provide correct DatabaseAccess configuration in config.xml
EAIF00039	**database id out of range**
Action	The given database id is not in the valid range
EAIF00040	**unsupported protocol:**
Action	The provided protocol is not supported
EAIF00041	**parse protocol error of**
Action	Error to parse the protocol. Provide correct protocol configuration in config.xml
EAIF00042	**Error loading configuration**
Action	Error to load configuration file. Provide a valid config.xml file in the default directory `configuration` in the Adabas Rest Administration installation directory.
EAIF00043	**Authentication configuration type is no longer supported**
Action	Provide a supported authentication login service type in the configuration. Contact your nearest support centre for further information.
EAIF00044	**Error to initialize JSON Web token**
Action	Provide correct configuration to initialize the Web token. Contact your nearest support centre for further information.

EAIF00045 **Job definition invalid (name empty)**

Action Provide a valid name for the Job definition

EAIF00046 **Job name is already available**

Action Job name provided is already available. Provide a new Job name.

EAIF00047 **Job not found**

Action Unable to locate Job provided. Provide a Job name that already exists

EAIF00048 **Delete execution ID fail, ID not found**

Action Failed to delete job execution from provided ID. Provide a valid ID to delete.

EAIF00049 **Error job not found**

Action Job name does not exist. Provide a valid job name.

EAIF00050 **Job storage not defined**

Action Job storage has not been defined. Provide valid Job storage configurations in config.xml

EAIF00051 **No valid database target defined**

Action No valid database target provided. Provide a valid database target configuration in config.xml

EAIF00052 **Job execution entry not found**

Action No Job execution entry was found

EAIF00053 **Connection not active**

Action Internal error, no active connection. Contact your nearest support centre for further information.

EAIF00054 **Job storage not defined**

Action Internal error, failed to configure a valid Job storage. Contact your nearest support centre for further information.

EAIF00055	**Container for creation are not given**
Action	Failed to create database without container list. Provide valid `ContainerList`
EAIF00056	**Container for creation are missing**
Action	Failed to create database due to incomplete container list. `ContainerList` requires four containers to be provided.
EAIF00057	**Error generating call to**
Action	Unknown error generating call to. Check output and contact your nearest support centre for further information.
EAIF00058	**Adabas renumber parameter not given or negative**
Action	File renumber parameter is not given or negative value. Provide a valid value that is bigger than 0.
EAIF00060	**Reference not found:**
Action	Reference for location is not found. Provide correct `Directories` configuration in config.xml
EAIF00070	**No Adabas cluster support has been configured**
Action	No Adabas cluster configuration found. Provide correct configuration for `Cluster Node` in config.xml
EAIF00080	**Encryption algorithm for creation is missing**
Action	Failed to create Database due to missing Encryption algorithm. Provide the correct keyword for the Database Encryption algorithm
EAIF00081	**Adabas nucleus log file not found:**
Action	Unknown error, nucleus log file is not found. Check for valid file and contact your nearest support centre for further information.
EAIF10000	**RBAC invalid usage of Assignment**
Action	Incorrect usage of Assignment, please check the request body for valid parameters.

EAIF10279	**[{0}] One value should be less than 16, another value should be greater or equal to 16**
Action	Invalid value provided for. Please specify the appropriate Blocksize

EAIF10280	**Invalid unit value**
Action	Invalid unit value provided. Please use the accepted values of : [M or B]

Error Messages from Adabas

AIF00000	**Normal usage**
Action	None

AIF00001	**Adabas interace**
Action	Interal error, please send Exception and trace log to support for detailed information.

AIF00002	**Adabas interface container generation operations**
Action	Disk space is left, add additional space

AIF00003	**Adabas database online check**
Action	Adabas database is not available

AIF00004	**Adabas database evaluation**
Action	Could not open ADABAS.INI. Please check ADADATADIR configuration

AIF00007	**Adabas create demo database tool**
Action	Invalid utility option for crdemodb

AIF00009	**Adabas environment check**
Action	Adabas environment is not set correctly

AIF00010 **Adabas operations**

Action Used Adabas interface version does not match to Adabas database structure level

AIF00011 **Adabas operations**

Action Database is in a state that an AUTORESTART need to be done or is done at the moment

AIF00013 **Adabas operations on LOB file**

Action Adabas operations are not allowed on LOB file. Use regular file

AIF00014 **Adabas file definition table generation**

Action The number of Adabas fields for new FDT is to high

AIF00015 **Adabas file ISN reusage on system files**

Action Reusage operations are not allowed on system files. Use regular file

AIF00016 **Adabas ciphering on system files**

Action Adabas ciphering is not allowed on system files. Use regular file

AIF00017 **ADAM usage**

Action ADAM files cannot be refreshed. Use regular file

AIF00018 **Adabas parameter value setting**

Action Given parameter value exceed maximum possible value

AIF00035 **FCB read**

Action Interal error, please send Exception and trace log to support for detailed information.

AIF00036 **Adabas container creation**

Action Given Adabas container does already exists

AIF00037 **Adabas database creation**

Action Given Adabas database id does already exists

AIF00038 **Access to Adabas GCB**

Action Access to the GCB is failed.

AIF00039 **Retrieving free space table**

Action Interal error, please send Exception and trace log to support for detailed information.

AIF00041 **Access to Adabas CSA**

Action Access to the CSA is failed.

AIF00042 **Set Adabas security mode**

Action Error setting the security mode of the Adabas database

AIF00148 **Adabas operations for online databases**

Action Adabas database is not active. Start the database

AIF00149 **Database container remove request**

Action Adabas database container removale is only possible on offline databases

AIF00151 **Adabas file changes**

Action Adabas file is locked for Adabas utility operations only. Unlock file

AIF00152 **Adabas interface**

Action A parameter given is invalid

AIF00153 **Adabas interface**

Action The given database ID is not valid in the used context.

AIF00154 **Adabas interface**

Action The given database file is not valid in the used context.

AIF00156 **Adding a LOB file for a Adabas file**

Action A base file, used for adding lob file, is missing

AIF00157 **Adding a LOB file for a Adabas file**

Action Given base file number and lob file number are equal. Use different file numbers

AIF00158 **Any Adabas operation**

Action the Adabas database is read-only. Change read-only state

AIF00159 **Read on ASSO container**

Action Failed to read records from ASSO container

AIF00160 **Loading a Adabas file**

Action Not enough container space to load the Adabas file

AIF00161 **Creation of database**

Action No permission to create Adabas containers

AIF00162 **Adabas file load**

Action File is either already available or not able to create

AIF00163 **Adabas lob file creation**

Action File is either already available or not able to create

AIF00164 **Adabas file load**

Action Cannot create a UCB entry. Number of entries is exceed.

AIF00166 **Adabas lob file creation**

Action Base Adabas file not available

AIF00167 **Set Adabas file option**

Action Not able to read Adabas file control block

AIF00168 **Adabas file load**

Action No space available in the Adabas ASSO containers

AIF00169 **Adabas file load**

Action No space available in the Adabas DATA containers

AIF00171 **Adabas file load**

Action No space available in the Adabas normal index

AIF00172 **Adabas file load**

Action No space available in the Adabas upper index

AIF00173 **Reset the USB range**

Action Failed to read the UCB entries

AIF00174 **Change Adabas container**

Action Failed to read ASSO blocks

AIF00179 **Rename database**

Action Failed to write the ASSO block

AIF00180 **Retrieve Adabas container information**

Action Failed to open the DATA file

AIF00181 **Adabas container operations/changes**

Action Failed to lock the database container before change

AIF00183 **Adabas shutdown, cancel or abort request**

Action The database is in progress to shutdown or cancel

AIF00184 **Delete Adabas database**

Action Access to delete Adabas resources is denied

AIF00187 **Creation of Adabas file, lob file or replication file**

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

AIF00190 **Loading of Adabas regular file or lob file**

Action DATA space storage is exhausted

AIF00195 **ADAM usage**

Action There are too many ADAM blocks used

AIF00196 **Record count calculation**

Action Internal error, contact support

AIF00197 **Adabas operations**

Action A stop or terminate is detected during operation run

AIF00201 **Adabas demo database creation**

Action Unknown error, contact support

AIF00202 **Adabas database creation**

Action Invalid block size given, use correct block size

AIF00203 **Adabas Database deletion**

Action Could not delete Adabas container. Check file permissions

AIF00204 **Retrieve Adabsa container information**

Action Not able to access WORK file

AIF00205 **Checkpoint file record modifications**

Action Failed to write checkpoint entry

AIF00206 **Adabas container deletion**

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

AIF00207 **Adabas container creation**

Action On the file system or RAW device is not enough space to create the container

AIF00208 **Adabas container creation**

Action The given RAW device is not correct.

AIF00209 **Adaba database creation**

Action Given path for ADADATADIR or container ASSOx or DATAx is not correct

AIF00300 **During Adabas utitility initialization**

Action The ADAPROGDIR configuration/environment variable is not correct

AIF00301 **Adabas error handling initialization**

Action Internal error, please contact support

AIF00302 **Adabas interface operations**

Action Either ADABAS.INI or `DB<dbid>.INI` cannot be found

AIF00303 **Adabas interface operations**

Action The ADADATADIR configuration/environment variable is not correct

AIF00304 **Adabas version query (deprecated)**

Action In the installation the ADAVERS is not set correctly or the configuration is wrong

AIF00306 **Adabas configuration read**

Action Configuration in INI files are missing or incorrect

AIF00307 **Adabas nucleus parameter set new value**

Action Failed to set value in INI file

AIF00308 **Adabas configuration file store**

Action Failed to write INI file

AIF00309 **Database id configuration search**

Action Adabas Database id is not registered in ADABAS.INI. Register Database id

AIF00310 **Read structure level out of ADABAS.INI**

Action Item for structure level is missing

AIF00400 **Read of checkpoint entries**

Action Internal error, contact support.

AIF00502 **Adabas file operations**

Action Cannot find the given Adabas file

AIF00503 **Adabas file operations**

Action Call tried on Adabas system files which is not allowed

AIF00504 **Adabas file operations**

Action Operation request is tried on Adabas lob files which is not allowed

AIF00507 **Set Adabas file option BT**

Action Adabas file option BT is not allowed to be set if Adabas is online

AIF00511 **Set Adabas file record spanning**

Action The Adabas file does not have a lob file defined. Define lob file

AIF00512 **Disable Adabas file record spanning**

Action Disabling of Adabas file record spanning is only allowed, if Adabas file is empty.

AIF00600 **Set an Adabas string based parameter**

Action The length of the string exceeds the maximum limit.

AIF00651 **Adabas interface request to start Adabas nucleus**

Action The start of the Adabas nucleus failed. Check Nucleus log

AIF00800 **Adabas interface**

Action Failed to allocate memory

AIF00801 **Adabas interface**

Action Failed to reallocate memory

AIF00850 **Adabas nucleus creation with encrypted container**

Action The AEL license file was not found

AIF00863 **Setting Adabas parameter**

Action Could not update INI files. Check permission

AIF00900 **Create demo database**

Action Example ordexp file is not available

AIF00901 **Create demo database**

Action Adabas utility to import demo data returns error

AIF01000 **Adabas set file operation**

Action A referential integrity error cause error

AIF01002 **Adabas file creation**

Action Referential intergrity is not allowed for system files

AIF01016 **Adabas file creation with reference integrity**

Action Adabas reference integrity file not available

AIF01100 **Setting of Adabas parameter userexit parameter**

Action User exit 1 and 11 are mutually exclusive

AIF02000 **Create an environment script file**

Action Error creating the assign file

AIF02001 **Delete an environment script file**

Action Error deleting the file

AIF02002 **Delete an environment script file**

Action File status is unknown

AIF02003 **Working with environment script files**

Action Internal error, contact support

AIF02004	**Working with environment script files**
Action	The corresponding assign files do not exist
AIF03000	**Adabas encryption unit**
Action	Failed to initialize cryptography subsystem, check output
AIF03001	**Adabas encryption unit**
Action	Given encryption key does not match to the container encryption
AIF09999	**Adabas nucleus and parameter operations**
Action	Internal error, contact support
`AIF10<rso>`	**Adabas call based operations**
Action	Please check Adabas response code documentation. Contact support
AIF20001	**Delete of Adabas container**
Action	Referenced Adabas container identifier is wrong
AIF20002	**Delete of Adabas container**
Action	Referenced Adabas container type is wrong
AIF20003	**Adabas file operations**
Action	Adabas file operations missing required parameter

System Component Error Messages

SYS00002	**Failed to initialize User**
Action	Failed to initialize unknown User. Create the specified User

SYS00003	**Authentication method is not yet implemented**
Action	The authentication method is not yet implemented. Use only the supported authentication method
SYS00004	**Specified user is already available in the list**
Action	Provide a different user that is not already in the list
SYS00005	**Specified user is not found in the list**
Action	Provide a valid user that is currently in the list
SYS00006	**Assigned PID file path is a directory, expecting a file**
Action	Check the default location for the PID file
SYS00007	**Fail to create PID file with same file path and name**
Action	Check the output of PID file path. Contact your nearest support centre for further information.
SYS00008	**Failed to create the PID file**
Action	Check the error message {1} for cause error. Contact your nearest support centre for further information.
SYS00009	**Error to {0} {1} watcher: {2}**
Action	Error when creating or adding an inotify watcher. Check the error message {2}. For Linux platform, check system limits for `fs.inotify.max_user_watches` and `fs.inotify.max_user_instances`. If there are insufficient resources, any modifications to the configuration file {1} are not reflected until the server is restarted.
SYS00029	**Error opening PID file**
Action	Check if PID file exists in the default location. Contact your nearest support centre for further information.