Adabas Text Retrieval System

Installation on UNIX

Version 2.3.3

December 2018

ADABAS & NATURAL

This document applies to Adabas Text Retrieval System Version 2.3.3 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: TRS-OS-INSTALL-233-20181213

Table of Contents

Preface

This documentation describes the installation of Adabas Text Retrieval 2.3.3 on a UNIX/Linux platform.

Hereafter, the generic term Adabas Text Retrieval will be used.

Adabas Text Retrieval is the heart of Software AG's text retrieval architecture. It offers the full range of functionality expected of powerful information retrieval systems. Applications which access both formatted and unformatted data simultaneously can be developed using Adabas Text Retrieval.

Since Adabas Text Retrieval is an extension of Software AG's database management system Adabas, it inherits such advantages as high-performance data compression, on to restart, automatic recovery and 24-hour operation.

Adabas Text Retrieval manages the index information and not the content of the data. This means that document contents can be stored at any location (Adabas, sequential files, CD-ROM, PC, etc.).

Adabas Text Retrieval can be used via its call interface from inside Natural or any third generation language such as C, COBOL or PL/1.

This documentation provides product-specific instructions for installing Adabas Text Retrieval.

This documentation is organized under the following headings:

System Requirements	Prerequisites for installing Adabas Text Retrieval
Installing Adabas Text Retrieval	How to install Adabas Text Retrieval
Verifying the Installation	How to proceed after the installation

For important last-minute information, see the readme file that is provided with Adabas Text Retrieval. You can find it in the Adabas Text Retrieval product documentation at *https://empower.softwareag.com/*.

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

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at **http://documentation.softwareag.com**. The site requires credentials for Software AG's Product Support site Empower. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at **https://empower.softwareag.com/**.

You can find product information on the Software AG Empower Product Support website at **https://empower.softwareag.com**.

To submit feature/enhancement requests, get information about product availability, and download products, go to **Products**.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the **Knowledge Center**.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at **https://empower.softwareag.com/public_directory.asp** and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at **http://techcommunity.softwareag.com**. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

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 System Requirements

Disk Space Requirements

An additional 5 MB of hard-disk space is required for Adabas Text Retrieval.

Additional Software Requirements

For the installation of Adabas Text Retrieval, the following additional software requirements apply:

- Adabas Version 6.5 and later versions
- If using Adabas Text Retrieval within Natural, this version requires Natural Version 8.4 and later versions. Adabas and Natural must be installed before the installation of Adabas Text Retrieval.

3 Installing Adabas Text Retrieval

Installation Package

The installation package containing the Software AG product Adabas Text Retrieval is available as compressed tar file.

Installation and Setup

The following is a summary of the steps required to set up the Software AG environment and install Software AG products for UNIX.

Note: For an overview of the directory structure created and the environment variables which are set at installation, see ***SAG Environment***.

- General Installation and Setup
- SAG Environment

General Installation and Setup

In this section the following is assumed:

- The user account for the administrator of Software AG products is called "sag".
- The group to which the administrator and all users of Software AG products are assigned is called "sag".
- The home directory for the user "sag" is */opt/softwareag*.
- The root directory for Software AG products is */opt/softwareag*.

Note: When upgrading a product, it is strongly recommended that you back up your current product version.

Create the administrator's account

- You must create one administrator's account and one group for all Software AG products when you install your first Software AG product.

1. Define an administrator account to which all of the Software AG products installed at your site belong.
2. Since all environment definition files for the products are written in Bourne shell syntax, the Bourne (or Korn) shell is recommended as the login shell for the administrator account. This section assumes that the administrator account is called "sag".
3. Define a group to which the administrator and all users of Software AG products belong.

4. Create a login directory for the user "sag".

Note: To perform these steps, use an appropriate system administration tool.

Example:

It is assumed that user and group accounts are defined in the respective files in */etc*.

The following is a possible entry in the system file */etc/group*:

```
sag:*:21:sag
```

The following is a possible entry in the system file */etc/passwd*:

```
sag::100:21:SAG - Product Administrator:/opt/softwareag:/bin/sh
```

The following is a command which creates a login directory for the user "sag":

```
mkdir /opt/softwareag
```

Log in as user "sag"

- This section assumes that the user "sag" is the administrator for Software AG products. It is not recommended to log in as root.

Unpack the downloaded installation package to disk

1 

Note: Make sure that the administrator user and group have been created and defined.

Download the installation package from the Software Download Center. The package names follow this naming rule:

```
<product>_<version>_<operating system id>_Any.<format>
```

where

<product>	= product three letter code like "TRS"
<version>	= current version of the product such as "2.3.3.04"
<format>	= "tz" for compressed tar format on UNIX, "zip" for zip format on Windows
<operating system id>	
AIX	IBM AIX on POWER architecture
HP11R	HP-UX 11 on PA-RISC architecture
HP11IT	HP-UX 11 on Itanium architecture

LNXAMD64	Linux on x64 architecture
LNXS390X	LNXS390X Linux on S390x architecture
SOL	Solaris on SPARC architecture
W64	Windows on x64 architecture

2 Change to the SAG home directory and extract the installation package with the following command:

```
UNIX: zcat TRS_2.3.3.04_<operating system id>_Any.tz | tar xvf -
Windows: unzip TRS_2.3.3.04_<operating system id>_Any.zip
```

Check README files

- If README files are included, read them before proceeding.

Check images

- Ensure that all installed images are owned by the user "sag" and have the group ID "sag".

Create the environment file "sagenv.new"

- The script *setup.sh* in the $SAG directory will create the *sagenv.new* file.

  1. Review the contents of *sagenv.new* and customize it as necessary.
  2. Rename *sagenv.new* to another file name (optional). In the following examples, it is assumed that the environment file is called *sagenv*.

  Note: If you are performing an update installation and changes were made to your environment, replace only the modified product-specific part in your existing *sagenv* file.

Environment File *trsenv*

- The INSTALL subdirectory of the TRS installation contains the environment file *trsenv* with all necessary settings for TRS.

  - Review the contents of *trsenv* and customize it as necessary.

Modify user profiles

- Enter the following command line in the *.profile* file of each user who will use this environment permanently:

```
. <SAG-root-directory>/sagenv
```

Set up the product

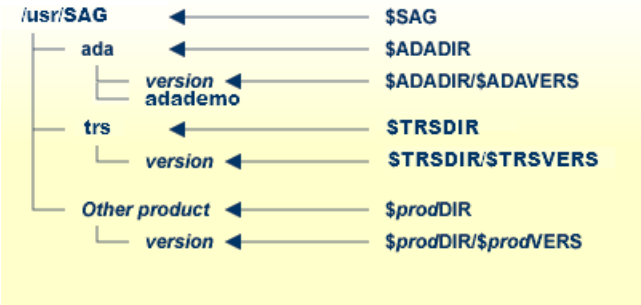
- You have completed the installation steps for TRS and can now set up the product.

  After reading the section *SAG Environment*, perform the steps required to use the product. If it is a new installation, the license file has to be copied to the appropriate place.

  1. Create directory $SAG/common/LKey.
  2. Copy the *trs23.xml* license file to the directory.

SAG Environment

The following figure shows the general directory structure generated during installation and the environment variables which reference the specified directories:



The environment variable `SAG` defines the root directory for all Software AG products and is usually the home directory of the administrator account.

For each product, the variable `$prodDIR` is set to the path of the main directory of the product specified, where prod is a three-letter product code in uppercase letters. For example, all files for Adabas Text Retrieval, whose product code is TRS, are contained in the directory $TRSDIR.

The name of the main directory is usually the same as the product code in lowercase letters. For example, the main directory for Adabas Text Retrieval is named trs. However, there are exceptions to this convention. For example, the product code for Entire Net-Work is WCP but the environment variables use the prefix NET instead. Also, the product code for Predict is PRD but the environment variables use the prefix DIC.

Version-independent parts of the product, such as examples or data, are stored in a subdirectory of the product main directory. For example, all Adabas demo data is contained in the directory $ADADIR/adademo.

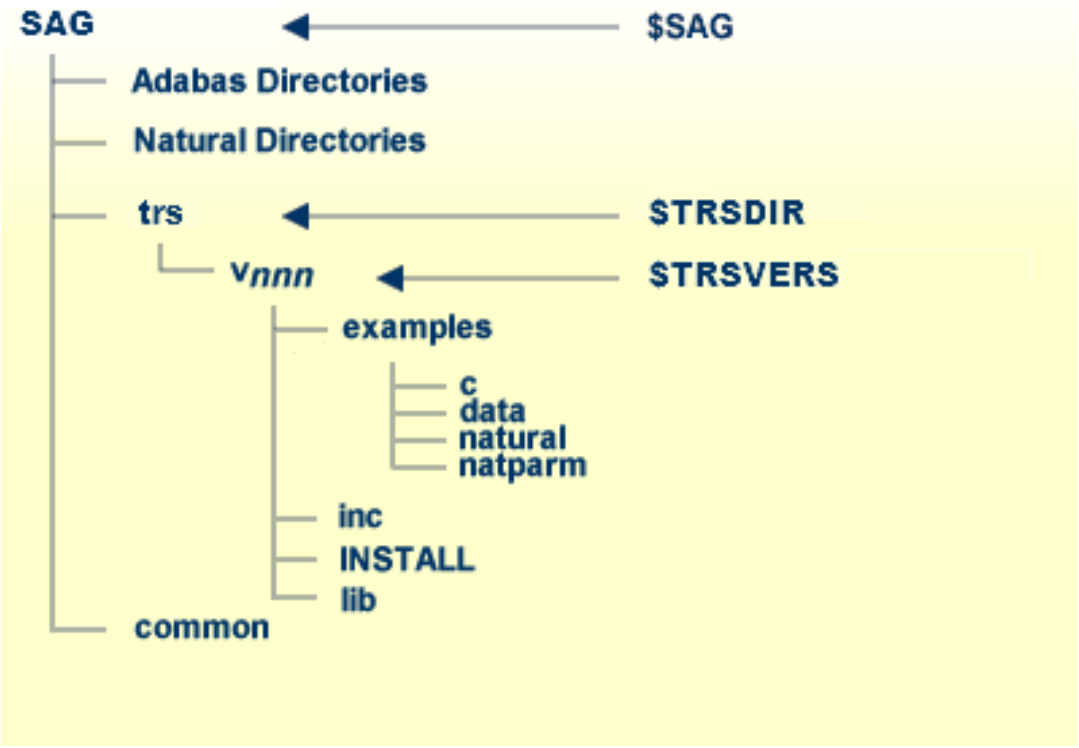
Version-dependent components of the product are kept in the version directory $prodDIR/$prodVERS. For example, the current version of Adabas Text Retrieval is stored in the directory $TRSDIR/$TRSVERS.

The environment variables `prodDIR` and `prodVERS` for all products specified during installation are set in the file *sagenv*. The same applies for any other environment variables needed for the various products.

4 Verifying the Installation

Checking the Directory Structure

After performing the general installation steps as described, the following Adabas Text Retrieval directory structure is generated:



The primary files in the directory structure are:

Directory	Description
SAG	The root directory for Software AG products.
trs	The top level Adabas Text Retrieval directory.
trs/v*nnn*	The version-dependent directory of the product.
common	SAG product common files (depending on installation type).

The files in the product subdirectories are as follows (below trs/v*nnn*):

Directory	File	Description
	README.TXT	Contains the latest technical details.
INSTALL	trsenv	Sets environment variables which are necessary for executing TRS.
inc	trsc.h	Header file to be included in applications written in C.
lib	trsc.a	Static library to be linked to applications written in C.
	trsc.[so|sl]	Shared library for applications built with C.
	nattrs.[so|sl]	Shared library for Natural applications.
	trshypa6.[so|sl]	Hyper Exit for Adabas Version 6.
examples/c	Makefile	Makefile for compiling and linking the C program trsmain.
	trsmain.c	Example program for calling TRS from a C application.
	uextempl.c	Template file for User Exits.
examples/data	trs_dfnr.fdt	Adabas FDT for Document file.
	trs_dfnr.fdu	Adafdu control statements for Document file.
	trs_vfnr.fdt	Adabas FDT for Vocabulary file.
	trs_vfnr.fdu	Adafdu control statements for Vocabulary file.
	document.inp	Data file for the example applications.
	load_data.bsh	Shell script to load the data from out.029 and out.28 into an Adabas database (UNIX).
	load_fdu.bsh	Shell script to load the Vocabulary and Document FDTs into an Adabas database.
	out.028	Vocabulary file.
	out.029	Document file.
examples/natural		This subdirectory contains all the Natural sources needed to establish the Natural example application.
examples/natparm	TRSPARM.LST	Example Natural parameter file.
examples/visualbasic	vbtrs32.bas	Example VisualBasic file.

Note: For HP-UX on PA-RISC architecture, shared libraries have extension ".s", other platforms have extension ".so".

Establishing the Environment

When Adabas Text Retrieval is used as a so-called Natural add-on product, the file *nattrs.[so|sl|dll]* must either be referenced by $NATUSER or it must be copied into the directory which is stated in the NATPARM utility under Binary Libraries for SAG Products. If possible, the installation procedure will try and copy the file into $NATDIR/extlib.

Migrating from Previous Versions

Due to some problems in heterogeneous environments (e.g., TRS on a swapping platform stores or retrieves data into or from a database on a non-swapping platform), the format of the `D7` field has been changed from `A12` to `B12`.

For this reason, the format of the D9 field has also been changed

```
from D9=D0(1,4),D1(1,2),D7(7,8)
to D9=D0(1,4),D1(1,2),D7(5,6)
```

The only way to migrate to the new format is to re-invert the documents.

Installing Adabas Text Retrieval

The information in Adabas Text Retrieval is stored in three logical Adabas files, which can be stored in one or more physical Adabas files:

- *vocabulary file (VFNR)*
- *document file (DFNR)*
- *document index (DSFNR)*

Adabas Text Retrieval is usually installed with two physical files, one for VFNR and one for DFNR/DSFNR. In all procedures, file numbers 28 and 29 were used as examples, but any valid file numbers can be specified. The same applies to the database ID. However, if you specify different numbers, then two Natural DDMs and two Natural programs must be adapted as described below.

The vocabulary file contains the word index. An example for a file layout is *trs_vfnr.fdu*. The necessary fields are described in section *File Structure* of the *Adabas Text Retrieval Reference Guide*. The document index file contains the internal document index created by Adabas Text Retrieval during the inversion process (also described in section *File Structure* of the reference information). The document file must contain user-defined formatted fields which are to be used for retrieval operations.

Create the Adabas Environment

If no Adabas database has previously been installed or if you want to dedicate an entire database to Adabas Text Retrieval, then you must create a new database. This procedure is described below.

Note: Please note that the database has to be started either before Adabas Text Retrieval is installed or by switching to a parallel session during the installation.

Create a demo database

Create a demo database (for example with the crdemodb tool or Adabas Manager).

Update Adabas parameters as required

For large production database applications, update the Adabas parameters in *adanuc.* as follows.

LAB	= 98304
LBP	= 6000000
LOGGING	= OFF
LP	= 400
LPXA	= 10
LS	= 100000
LWP	= 1000000
MGC	= 50
NC	= 1000
NH	= 10000
NISNHQ	= 10000
NOPLOG	
NT	= 6
NU	= 1000
TNAA	= 3000
TNAE	= 3000
TNAX	= 3000
TT	= 3000

The value of the `NU` parameter must be at least as large as that of the `NC` parameter.

Start database

Start the database with

```
adastart <dbid>
```

Note: If you encounter problems when starting the database, please use the following for error logging:

```
more  $ADADIR/db<dbid>/adanuc.log
```

Installing the Natural DEMO Application

Load the file definition (vocabulary file (28) and document file (29)

```
cd $TRSDIR/$TRSVERS/examples/data
load_fdu.bsh <dbid> <dfnr> <vfnr>
```

If you have created a new database, you must load the document file *out.029* and the vocabulary file *out.028* by running the following script:

```
cd $TRSDIR/$TRSVERS/examples/data
load_data.bsh <dbid> <dfnr> <vfnr>
```

If the data is not loaded, work2 and work3 have to be executed later during the Natural Application.

Install the Natural DEMO Application

Get correct *INI* and *CFG* file in $NATDIR/$NATVERS/etc by using the NATPARM utility and change the settings in *Configuration and Global Configuration File*. Create your own NATPARM module. A standard Natural parameter file can be imported with natparm which has all TRS parameters set at default value.

```
natparm
→ file:
       → import
              → file name:
                           TRSPARM
              → path:
                           $TRSDIR/$TRSVERS/examples/natparm
```

The following values must also be set for the OPRB (Database Open/Close Processing) parameter:

```
(DBID=nn,UPD=28-29)
```

where `nn` is the ID of your database, 28 is the file number of the vocabulary file (*VFNR*), and 29 the file number of the document file (*DFNR*).

Installing the Natural DEMO Application under Natural Security

Natural Security users should specify the following:

```
        OPRB: (NOOPEN)
→ DBMS:
        UDB (0)
→ LIMITS:
         MADIO: 0
         MAXCL: 0
→ SYSTEM FILE:
         FNAT   (<DBID>,<FILE>) (as in NATCONF.CFG)
         FUSER  (<DBID>,<FILE>)
→ WORKFILES:
         3   $WRKTRS3
→ file:
          → SAVE (parm file name)
          → EXIT
```

Start the Natural buffer pool with:

```
natstart.bsh
```

Start Natural with your parameter module using:

```
natural parm=<parm file name>
```

Import the TRS Demonstration Library

Note: Natural Security users need to define library TRSDEMO using SYSSEC maintenance. For Natural Security Version 5.1, please refer to section *Library Maintenance* in *Natural Security for UNIX*.

```
Services
→ Sysmain
     → Import
          → PATH:$TRSDIR/vnnn/examples/natural<CR>
                 OBJECT: *<CR>
                                        a<CR> (for import all)
                                        <CR>   (for structured mode)
                 LIBRARY: TRSDEMO<CR>
```

Note: Natural Security users may receive a NAT0972. If this error occurs, users should logon to TRSDEMO using the direct command `logon TRSDEMO` and then invoke DDM Services.

```
→ DDM Maintenance
     → E TRS-DOCUMENT<CR>
     → (ignore NAT3148 error)
                    <CR>
                    <ESC>
                    <CURSOR RIGHT>
                    <CR>
                    M[odify DDM-HEADER]<CR>
                          DBID: <dbid>
                    <tab>
                          FNR: <file-id>
                    <CR>
                    (ignore warnings for hyper exit - NAT6420)
                    <ESC>
                    <CR>
                    <ESC>
                    <CURSOR RIGHT>
                    <CURSOR RIGHT>
                    <CR> E[xit]<CR>
```

Now perform the same steps for the vocabulary file:

```
→ E TRS-VOCABULARY<CR>
    ...
<ESC>
Q[UIT]<CR>
<ESC>
```

Cataloging the demo application

```
logon TRSDEMO
catall
→ <TAB>
     <TAB>
     x<CR>
```

Run the Demo Application

Note: *work2* and *work3* only have to be executed if the data has not already been loaded during the installation.

```
logon TRSDEMO
work2
   Load documents from $WRKTRS3.
work3
   Invert the documents.
Execute the TRS-DEMO_Program “menu”                (Start Retrieval)
 → Enter “R” to select the function “Retrieval”    (Retrieval)
   → Enter:        Order: gt 0<CR>                (Result should be 315)
                   Title: adabas<CR>              (Result should be 65)
                   Abstract: ada,nat<CR>          (Result should be 1)
                   Abstract: a general<CR>        (Result should be 3)
                   Date: gt 0<CR>                 (Result should be 315)
```

To display the documents and check highlighting, press **PF6**.

Index