software AG

Adabas

Release Notes

Version 8.1.4

June 2014

Adabas

Table of Contents

1 Adabas 8.1 Release Notes

This release lifts many of the limits imposed by prior releases of Adabas, expanding Adabas storage capabilities and usefulness. This document provides a brief summary of the new and changed features included in Adabas 8.1, with links for more information to other areas of the Adabas documentation set.

Because the updates made to Adabas 8 are quite extensive, we recommend that all users read the *Planning Manual for Adabas 8* before proceeding with the Adabas 8 installation. This planning documentation describes the architectural changes and new features of Adabas 8, as well as any compatibility or migration issues between Adabas 8 and prior versions of Adabas. Note that some planned features for Adabas 8 are not yet available in this release.

⚠ **Important:** Be sure that you apply all supplied Adabas 8 maintenance and concatenate Adabas 8 patch-level libraries (L00n), as they are delivered to you. This will ensure that your Adabas 8 code remains up-to-date, supporting all Adabas 8 features as they are enhanced and maintained.

If you are upgrading to this Adabas release from a release prior to the most recent Adabas release (for example, if you are upgrading from Adabas 7.4.3 to Adabas 8.1 and skipping the intermediate upgrade to Adabas 7.4.4), please read the Release Notes for the releases you are skipping to get a complete understanding of all of the changes implemented in Adabas since you last updated your software.

This document covers the following topics:

2 Supported Platforms

Software AG generally provides support for the operating system versions supported by their respective manufacturers; when an operating system provider stops supporting a version of an operating system, Software AG will stop supporting that operating system version.

Before attempting to install this product, ensure that your host operating system is at the minimum required level. For information on the platform versions supported by Software AG products, access Software AG's Empower web site at *https://empower.softwareag.com*.

Log into Empower. Once you have logged in, you can expand **Products** in the left menu of the web page and select **Product Version Availability** to access the Product Version Availability application. This application allows you to review the supported platforms and releases supported for any Software AG product.

Use the fields on this application to filter its results. When you click the **Find** button on this application, a list of the supported Software AG products that meet the filter requirements is shown. You can clear all filter selections using the **Clear** button.

> **Note:** Although it may be technically possible to run a new version of this product on an old operating system, Software AG cannot continue to support operating system versions that are no longer supported by the system's provider. If you have questions about support, or if you plan to install this product on a release, version, or type of operating system other than those described in the Product Version Availability application as described above, consult Software AG technical support to determine whether support is possible, and under what circumstances.

3 Enhancements - Version 8.1.4

Version 8.1.4 of Adabas is largely a maintenance release, but includes the following enhancements.

Mainframe Licensing Updates

The licensing process for Adabas has been updated in this release to support changes in the standard Software AG mainframe product licensing software. Two new licensing libraries are now provided on the installation tape. Both libraries are distributed with names starting with the letters "MLC". In addition to these new libraries, a new licensing utility, LICUTIL, is provided that allows you to display and check your license file or to convert it to assembler source. This utility is provided in addition to the ADALICAM program used to license prior releases of Adabas.

As a result of this new licensing software, some changes have also been made in the Adabas installation. Steps in the Adabas installation procedures have been added or altered in support of this updated license process.

For general information on Software AG mainframe product licensing, read *Software AG Product Licensing*, in the *Adabas Operations Manual*. For specific information on licensing Adabas, refer to the Adabas installation documentation for the appropriate operating environment.

New Personnel Demo File

A new Personnel demo file is provided in this release of Adabas. This new demo file joins the Employees, Vehicle, and Miscellaneous demo files that have been provided in the past. The new Personnel demo file includes fields that demonstrate the use of the new expanded features of Adabas 8, such as large object support. This new Personnel demo file will also be used as the example file for future versions of Natural.

| **Note:** This new Personnel demo file must be installed on a UES-enabled database because it includes wide-character format (W) fields.

To review the FDT of the new Personnel demo file provided with Adabas, read *Adabas Personnel Demo File*, in the *Adabas Utilities Manual*.

Changes in Distributed INPL Jobs

The distributed INPL jobs (both the sample jobs and the SMA-generated jobs) that you use to load the Adabas INPL library have been changed. The Adabas INPL library stores code for the demos of some Adabas add-on products (such as the demo version of Adabas Online System). These distributed jobs now load the INPL library in a date-sensitive manner. In other words, the load process will now check the dates of your existing INPL library and will not allow older members to overwrite members with newer dates.

However, if you use your own Natural batch jobs to load the Adabas INPL library, you will need to modify them to be date-sensitive. To do this, specify the following `CMSYNIN` primary command input in your job (this setting assumes the Natural input parameters in the job are specified in comma-delimited mode, or `IM=D`):

```
B,,,,,,,,Y
```

The "B" setting indicates that the INPL action should load everything; the next six fields (comma-delimited) are defaults, the eighth field is specified as "Y" to indicate that dates in the INPL library should be checked, and the ninth field is not included in the specification because the default for that field will be used. For more information about Natural `CMSYNIN` input, refer to your Natural documentation.

> **Note:** When migrating an Adabas 7.4 installation, this procedure does not apply. Instead, you should replace the 7.4 INPL library members with the latest Adabas 8 INPL library members, regardless of the dates of the members, to avoid creating a library containing members from both releases.

Command Option 1 and Command Option 2 Checking

This version of Adabas reintroduces the command option checks that existed in previous Adabas versions. These checks detect when user programs are not correctly clearing the Command Option 1 and Command Option 2 fields (ACBCOP1/ACBCOP2 and ACBXCOP1/ACBXCOP2) and therefore may contain left over information from a previous call. This leftover information can cause incorrect results for the next call. If an invalid command option field is detected by the check, a response code of 34 will be returned.

Future versions of Adabas will introduce additional command options we strongly recommend that invalid command option settings be detected now and corrected. However, if you need to temporarily remove this check, an optional zap is provided in the ZAPOPT member to assist you. This special zap will not be available when Adabas 8.2 is introduced.

LGBLSET Changes

Changes have been made to the LGBLSET macro in this release. The changes that have been made include:

1. An increase in size of the Adabas Review work area. When the `REVIEW` parameter is set to "YES", a work area is set up for Adabas Review. The size of this work area has now been increased from 256 bytes to 512 bytes. Adabas Review 4.5 (and later releases) requires this larger work area.

2. When parameters `COR=YES` (indicating the Adabas System Coordinator and Adabas Transaction Manager product suite and their link routines are in use) and `DYNDBSVC=YES` (enabling Adabas SVC routing by database ID) are both specified in the link globals table, the following messages are now issued (via MNOTE):

```
*,DYNDBSVC=YES and COR=YES are mutually exclusive
*,COR/ATM does not support dynamic SVC lookup.
```

   These messages are followed by the following general message:

```
16,LGBLSET - INVALID PARAMETER SPECIFICATION.
```

   The Adabas System Coordinator and Adabas Transaction Manager product suite and their associated link routine exits do not support the Adabas SVC routing by database ID feature. To avoid problems with the Adabas System Coordinator and Adabas Transaction Manager product suite when the dynamic SVC by DBID routing feature is enabled, the messages above are issued, the assembly step of the globals table will receive return code 16, and the globals table load module will not be generated.

Therefore, if you are installing one of the following Adabas add-on products in z/OS or z/VSE environments, you will need to reassemble the new link globals tables and relink the Adabas link routines to pick up these changes:

- Adabas Review 4.5 or later

- The Adabas System Coordinator and Adabas Transaction Manager product suite and their associated link routine exits

If you are not installing one of these products, you need do nothing. For information on reassembling the link globals tables and relinking the Adabas link routines, read the section entitled *Installing Adabas With TP Monitors* in either your *Adabas z/OS Installation GuideS* or your *Adabas z/VSE Installation GuideE*.

ADASIP REPLACE Parameter in z/VSE Environments

A new REPLACE parameter has been introduced in z/VSE environments. This functionality was actually added in a zap, but is include with Adabas 8.1.4. This parameter allows you to control ADASIP behavior if an attempt to install a new SVC over a previously-installed SVC should occur. Options are to replace the current SVC (REPLACE=YES) or to issue warning messages instead (REPLACE=NO). The default is REPLACE=NO.

Caution: Setting the REPLACE parameter to YES should be done carefully. Replacing an SVC while your targets are running can produce unpredictable results.

For more information on this parameter, read *Initializing the Adabas Communication Environment*, in the *Adabas z/VSE Installation GuideE*.

Parallel Access Volume Support with IBM z/VSE V4.2

This version of Adabas supports Parallel Access Volumes (PAV), new functionality that will be enabled in IBM z/VSE 4.2. This functionality is announced and described in the IBM z/VSE V4.2 Announcement Letter. To enable this functionality for z/VSE 4.2, please refer to IBM's **z/VSE Home Page** for any required maintenance that may be necessary for this support. Adabas has been modified to do extent limiting in its use of channel programs to the storage subsystems that support PAV under z/VSE.

Entire Net-Work Administration and Entire Net-Work Client Now Included in Adabas Package

This version of Adabas is shipped with the code for Entire Net-Work Client (open systems software) and Entire Net-Work Administration (mainframe software). Entire Net-Work Client and Entire Net-Work Administration are Software AG middleware packages used for communication between Adabas or Event Replicator Server databases on the mainframe and open systems software packages such as Adabas Manager (including the Adabas Manager demo) or Event Replicator Administration. Entire Net-Work Administration is a limited version of Entire Net-Work for mainframes and includes the Simple Connection Line Driver.

Note: Entire Net-Work Client requires a license key. A limited license is shipped with your Adabas software to support the Adabas Manager demo. If you purchase a full version of Adabas Manager, you will need a full license of Entire Net-Work Client.

If appropriate Entire Net-Work mainframe and client products are not already installed on your system, install Entire Net-Work Administration on the mainframe and Entire Net-Work Clienton the client side. For complete information on these products, read the Entire Net-Work Administration documentation and Entire Net-Work Client Administration.

User Exit Enhancements

This version of Adabas has enhanced user exits 4 and 11.

- The linkage to user exit 4 has been clarified and additional parameters have been documented.
- A sample user exit 11 skeleton called UEX11 is now provided.

For complete information, read *User Exit 4 (User-Generated Log Data)* and *User Exit 11 (General Processing)* , in the *Adabas User, Hyperdescriptor, and Collation Descriptor Exits Manual*.

4 Enhancements - Version 8.1.3

This chapter describes the enhancements that have been made to Adabas for version 8.1.3.

ADAWRK Utility -- Work Area Recovery Reports

This release introduces a new utility, called the ADAWRK utility. This utility can be used when the database autostart fails and the database will not come up. It produces reports from records in Work part 1 that can help you determine whether:

■ You should run a restore/regenerate (ADARES utility, REGENERATE function) of the database, which can be time-consuming.

■ The database can be quickly repaired so it can be started and functional more quickly.

For complete information about the ADAWRK utility, read *ADAWRK Utility: Work Area Recovery Reports* , in the *Adabas Utilities Manual*

Adabas SVC Routing by Database ID

This release of Adabas introduces the ability for your application programs that use Adabas link routines in z/OS and z/VSE environments to route database calls through specific Adabas SVCs, based on the database ID in the call. SVC routing is managed through the use of a DBID/SVC routing table you supply. For complete information, read *Establishing Adabas SVC Routing by Adabas Database ID*, in the *Adabas z/OS Installation GuideS* (z/OS) or *Establishing Adabas SVC Routing by Adabas Database ID*, in the *Adabas z/VSE Installation GuideE* (z/VSE).

📄 **Notes:**

1. Adabas client-based add-ons, such as Adabas Transaction Manager, are not compatible with this feature since for client-based functionality to work, it must be channeled through only a single router for any given session, not across routers.

2. ADALNK linked with the ADASVCTB should only be used by application programs and should not be made available to the Adabas nucleus or to Entire Net-Work.

🛑 **Caution:** This feature should be used with caution. Transactional integrity is not guaranteed. If an application makes calls to multiple databases that are routed to more than one Adabas SVC, it becomes possible to issue ET, BT, OP, CL, RC, or other Adabas commands that may affect the transaction on one database, but not on the other databases running on different Adabas SVCs that were accessed previously. It therefore is the responsibility of the application program to ensure that all necessary logic is included to ensure transactional integrity across multiple databases where multiple Adabas SVCs are employed.

This feature introduces the use of two new LGBLSET macro (link globals table) parameters, DYNDBSVC and DBSVCTN. DYNDBSVC is used to turn the Adabas SVC routing feature on and off; DBSVCTN identifies the DBID/SVC routing table to use when the feature is active.

CICS High-Performance Stub Routine Support

Support for the CICS high-performance stub routine under Adabas 8 has been added in this release. The Adabas high-performance stub routine extends the direct call interface (DCI) facility that is available with the Adabas CICS command-level link component to applications written in languages other than Software AG's Natural (for example, Assembler, COBOL, PL/I).

> **Note:** The stub routine must be used with the Adabas CICS command-level link component. The stub routine will not function properly with the Adabas CICS/VSE macro-level link component. The LNCSTUB module delivered in the Adabas Version 8.1.3 library will also function properly with Adabas Version 7.4 CICS link routines.

For complete information about the CICS high-performance stub routine under Adabas 8, read *Installing the CICS High-Performance Stub Routine for Adabas 8*, in *Adabas z/OS Installation GuideS*.

PRILOGC Print Program Support

Adabas now provides the PRILOGC print program to read and report the contents of Adabas command logs in either the version 5 or the version 8 command log layout format.

For complete information about PRILOGC, see *PRILOGC: Printing the Command Log*.

> **Note:** In z/VM environments, solution ADA813L002 must be applied before you can use PRILOGC. For more information about this solution and its documentation, please contact your Software AG support representative.

ACBX Changes

The function of the ACBXERRE field has changed. Its functionality has been moved to a new two-byte field called ACBXERRF. The ACBXERRE field is now only a one-byte field (it was previously a three-byte field) and is now reserved for future use. If you have any applications that make use of the ACBXERRE field, please adjust them to use the new two-byte ACBXERRF field instead. The new ACBXERRF field is the field that now stores the buffer segment containing the error (if any) referred to by the ACBXERRA and ACBXERRD fields.

For complete information about the current ACBX structure, read *Extended Adabas Control Block (ACBX)*, in the *Adabas Command Reference Guide*.

Support for Triggers and Stored Procedures in z/VM Environments

This release introduces support in Adabas 8 for triggers and stored procedures in z/VM environments. This support was not provided in prior releases of Adabas 8.

> **Note:** You must apply solution ADA813L002 before you can use triggers and stored procedures in z/VM environments. For more information about this solution and its documentation, please contact your Software AG support representative.

A new EXEC file is provided (TSPBLDM EXEC) for this support. Using this new EXEC, you can now choose and assemble the Natural version you want to use (as long has you have Natural 4.1 or later installed).

Enhancements for Other Software AG Products

Updates have been made to Adabas 8 code to better support the use of Adabas 8 by other Software AG products, as described in the following table:

Product	Enhancements
Adabas Cluster Services	Updates have been made to Adabas 8 base code to better support Adabas Cluster Services 8.1 high availability enhancements. For complete information about Adabas Cluster Services 8.1 enhancements, read the *Adabas Cluster Services 8.1 Release Notes*.
Adabas Parallel Services	Updates have been made to Adabas 8 base code to better support Adabas Parallel Services 8.1 high availability enhancements. For complete information about Adabas Parallel Services 8.1 enhancements, read the *Adabas Parallel Services 8.1 Release Notes*.
Adabas Review	Updates have been made to Adabas 8 base code to provide support for Adabas Review 4.4 enhancements. Adabas Review 4.4 now makes use of the new CLOG layout available in Adabas 8 and includes performance enhancements. For complete information about Adabas Review 4.4 enhancements, read the *Adabas Review 4.4 Release Notes*.
Event Replicator for Adabas	Updates have been made to Adabas 8 base code to provide better support for Adabas-to-Adabas replication. In addition, Adabas 8 code was enhanced to support Event Replicator for Adabas 3.1 enhancements, which include enhancements to Event Replicator Administration 2.4 and Event Replicator Target Adapter 2.4. For complete information about Event Replicator for Adabas 3.1 enhancements, read the *Event Replicator for Adabas 3.1 Release Notes*.
Natural	Updates have been made to Adabas 8 code to support Natural version 4's Adabas Interface Mode (ADAMODE) enhancements.

5 Enhancements - Versions 8.1.1 and 8.1.2

This chapter describes the enhancements that have been made to Adabas for versions 8.1.1 and 8.1.2.

Logical Extent Limit Lifted

The limit of five logical file extents for each Adabas file extent type has been lifted. The maximum number of logical file extents that can be defined is limited only in that the extent information of all address converters, Data Storage, normal index, and upper index extents for the file must fit into the file control block (FCB). (The extent information is stored in a variable section of the FCB.) For example, on a standard 3390 device type, a file could have more than 40 extents of each type (or there could be more of one type if there are less for another). For more information about Adabas 8 logical extents, read *Adabas Logical Extents*, in *Adabas DBA Tasks Manual*.

Physical Extent Limit Lifted

The Associator and Data Storage components of your Adabas database may now each contain more than five physical (database container) extents. A new maximum of 99 physical extents is now set for each. However, your actual real maximum could be less because the extent information of all Associator, Data Storage, and Data Storage Space Table (DSST) extents must fit into the general control blocks (GCBs). For example, on a standard 3390 device type, there could be more than 75 Associator, Data Storage, and DSST extents each (or there could be more of one extent type if there are less for another). For more information about Adabas 8 physical extents, read *Adabas Physical Extents*, in *Adabas DBA Tasks Manual*.

Expanded GCB

To accommodate all of the other expansions in Adabas 8, the general control block (GCB) now spans two blocks instead of one block. These blocks are now referred to as the *general control blocks (GCBs)*.

MU and PE Limits Increased

The number of occurrences of each MU field or each PE group in a record has been increased from 191 to about 65,534. However, the actual number of occurrences is limited to the size of the datablock size, the device type and the file type (spanned or not spanned). All MU fields and PE groups and other fields must fit into one compressed record. If you are using spanned records (introduced with Adabas 8), more MU fields and PE groups can be stored.

In addition, subdescriptors and superdescriptor definitions can affect the number of MU fields or PE groups in the record. For example, if a superdescriptor is created as a combination of a PE group and one or more MU fields and the number of occurrences is high, performance and resource problems can occur.

> **Note:** Excessive use of extended MU and PE fields might cause performance and resource problems. These can result in a work storage overflow, resulting in Response code 9. If this should happen, increase the ADARUN LP size for the database.

The use of more than 191 MU or PE fields in a record must be explicitly allowed for a file (it is not allowed by default). This is accomplished using the new ADADBS MUPEX function or the ADACMP COMPRESS `MUPEX` and `MUPECOUNT` parameters.

If a file has been established with extended MU or PE limits, you should not read the occurrence count of an MU field or PE group into a one-byte field in the record buffer. If you try, Adabas returns response code 55, subcode 9. Therefore, any application program that reads the occurrence count using an `xxC` element in the format buffer (for example, FB='MUC.' or FB='MUC,1,B.') must be changed to read the occurrence count into a field with two or more bytes (for example, FB='MUC,2,B.' or FB='MUC,4,B.').

- Identifying MU and PE Occurrences Greater Than 191

Identifying MU and PE Occurrences Greater Than 191

MU and PE occurrences greater than 191 are indicated in compressed records (ADACMP utility) by a `x'C0'` byte at the beginning of the occurrence count. This byte is set by the ADACMP utility or the nucleus when the records are compressed. The `x'C0'` indicator byte is followed by a byte indicating the number of count bytes used for the MU or PE occurrence count that follows. For example, consider the following indicator:

```
X'C0020204'
```

In this example, x'C0' indicates this is an extended count; x'02' indicates that there are two count bytes, and x'0204' indicates that there are 516 occurrences of the field.

Expanded System File Numbers

Two-byte file numbers can now be specified for all Adabas 8 system files (CHECKPOINT, SECURITY, SYSFILES, and TRIGGER), with numbers up to 5000. Physically coupled files, however, still cannot have numbers higher than 255. Note, however, that you cannot use the ADACNV utility to revert the database to a version prior to Adabas 8 once you implement two-byte system file numbers.

Spanned Record Support

Record spanning is implemented. The logical record may now be stored in multiple physical blocks. This expansion is provided to accommodate the extended features of Adabas 8, such as the increase in MU and PE occurrences. For more information, read *Spanned Record Support*, in *Adabas Concepts and Facilities Manual*.

> **Note:** Spanned record support must be explicitly allowed for a file. You can currently do this using the ADADBS RECORDSPANNING function or the SPAN parameter of ADACMP COMPRESS function.

To support spanned records, a secondary address converter is required. This address converter is used to map the ISNs of secondary spanned records to the RABNs of the Data Storage blocks where the secondary records are stored. For more information, read *Spanned Record ISN Use*, in *Adabas Concepts and Facilities Manual*.

Large Object Field Support

A new field option, LB, is provided to identify a field as a *large object field*. A large object field (LB field) is an alphanumeric field that can have a theoretical size of up to 2 GB. Such fields can be used, for example, to store documents (for example, HTML, XML, Microsoft Word, or PDF documents), pictures (for example JPG or BMP files), or other data conglomerates in single fields in the database.

Adabas stores LB field values in a separate file, called a *LOB file*, that is tightly associated with the file containing the LB fields, which is called the *base file*.

For more information about LB fields, read *Large Object Option LB*, in *Adabas Concepts and Facilities Manual* and *Getting Started with Large Object (LB) Fields*, in *Adabas DBA Tasks Manual*.

LB field support also includes utility enhancements. Specifically, enhancements to the ADACMP, ADADBS, and ADALOD utilities have been made. For more information, read about the *Utility Enhancements*, elsewhere in this guide.

Direct Call Changes

Adabas 8 introduces a new format of direct calls that is more powerful and flexible than the classic Adabas direct call interface of Adabas 7 and earlier:

- The data to be written to or read from the database in one command can be spread over multiple discontiguous buffer segments. It need not be put together into one physically contiguous buffer.
- Each buffer or buffer segment can be larger than 32 KB. For practical purposes, it can be as large as the application program's working storage allows.

The new format of direct calls makes use of the **extended Adabas control block (ACBX)** and is called the ACBX direct call interface. The old format continues to use the classic Adabas control block (ACB) and is called the ACB direct call interface.

Adabas 8 understands both formats of direct calls. Your application programs can choose, on a call-by-call basis, whether to use the ACBX interface or the ACB interface. Your direct calls are only required to use the ACBX interface if the application needs to use one of the new features of the ACBX (multiple buffer segments, large buffers).

Adabas versions prior to Adabas 8 support only the ACB direct call interface. If they receive a call using the ACBX interface, a response code 22 will result (invalid command).

For more information, read *Calling Adabas*, in the *Adabas Command Reference Guide*.

New Extended Adabas Control Block (ACBX)

The new extended Adabas control block (ACBX) introduced in Adabas 8 supports the ability to read and write data on the database using multiple, discontiguous, large buffer segments. This is especially useful for writing or reading large objects (LB fields), but can be used for any fields.

The existing ACB (non-extended) continues to be supported and your existing applications will continue to work, but if you want to take advantage of some of the extended features in Adabas 8, you must use the new ACBX.

For more information, read *Adabas Control Block Structures (ACB and ACBX)*, in the *Adabas Command Reference Guide*.

New Adabas Buffer Description (ABD)

A new structure called an *Adabas buffer description* (ABD) is introduced with Adabas 8 and must be used in ACBX interface direct calls. ABDs cannot be used in ACB interface direct calls.

ABDs support the use of segmented, discontiguous buffers. Each ABD describes one buffer segment, identifying what kind of buffer segment it is, where it is located, what size it is, and other pertinent information.

For more information, read *Adabas Buffer Descriptions (ABDs)*, in the *Adabas Command Reference Guide*.

Buffer Enhancements

With the introduction of the ACBX direct call interface, Adabas 8 offers various buffer enhancements for direct calls:

- New multifetch, performance, and user buffers can now be specified if you are making an ACBX interface direct call.

- Adabas format, record and multifetch buffers can be split into multiple segments, which need not be contiguous in storage.

  Format and record buffer segments are specified in pairs that belong together. For calls requiring multifetch processing, the format, record, and multifetch buffer segments are specified in triplets that belong together.

- More than 32 KB of data can be specified per buffer or buffer segment.

For more information, read *Defining Buffers*, in the *Adabas Command Reference Guide*.

Furthermore, Adabas 8 offers format buffer enhancements for direct calls made using either the ACB or ACBX direct call interfaces:

- For large object (LB) fields, a zero length specified in the format element indicates that the amount of space available for the field value in the record buffer is variable and depends on the actual value of the field. The actual field value is determined at runtime and is stored in the first four bytes of the LB field value in the record buffer, preceding the field value itself. The length value is the sum of the length of the LB field value plus the four-byte length specification itself.

- For long alpha (LA) and LB fields only, an asterisk (*) can now be specified instead of a length in the format element. This indicates that the amount of space available for the field value in the record buffer is variable and depends on the actual value of the field. The actual field length is determined at runtime. However, unlike the zero length specification setting, *no* four-byte

length field precedes the LA or LB field value in the record buffer; the record buffer area corresponding to the format element contains only the value of the field itself.

For more information about these field length specifications, read *Length and Data Format* , in the *Adabas Command Reference Guide*.

■ A new format buffer indicator (*L*), referred to as the *length indicator*, can now be used to retrieve or specify the actual length of an LA or LB field value. This indicator is often combined with a format element with asterisk notation, where the latter specifies the field value proper and the former specifies the length of the value. For more information, read *Length Indicator (L)*, in the *Adabas Command Reference Guide*.

■ For LA and LB fields, the fixed length that can be specified in format elements has been extended from 253 bytes to a (theoretical) maximum of 2,147,483,647 bytes (2 GB).

For more information about these buffers, read *Defining Buffers*, in the *Adabas Command Reference Guide*

FDT Enhancements

The internal FDT structure in version 8 has increased and these larger FDTs may use more than four Associator blocks. The additional blocks required for a larger FDT are automatically allocated from the Associator free space. The fixed space for FDTs in the Associator will remain reserved to accommodate backward compatibility and conversion.

Several new options have been added that you can use when defining fields in the FDT:

■ A new LB (large object) option can be used to define large object fields.

■ A new NB (no blank compression) option can be used in the definition of alphanumeric and wide-character fields in the FDT. Its presence indicates that trailing blanks in a field should not be removed by Adabas. By default, NB is not specified, and Adabas does remove trailing blanks when storing alphanumeric and wide-character values.

For more information about these FDT updates, read *Field Options*, in *Adabas Concepts and Facilities Manual*.

Command Changes

If an "I" is specified in Command Option 2 for an LF command, all field information is returned in the new Adabas 8 internal format.

In addition to the direct call changes, previously mentioned, LF command functionality has been altered to accommodate LB fields. If an LF command with Command Option 2 set to "S" is run and large object fields (LB fields) are encountered, the LB field description is returned in an F-type field element. Bit 6 in the second format byte (at offset 7 or byte 8 in the element) is set to indicate that the LB (large object) option is set for the field. In addition, bit 1 of the second format byte indicates whether the LB field is defined with the NB option. For more information, read *LF Command: Read Field Definitions* in the *Adabas Command Reference Guide*.

SVC Enhancements

Adabas 8 includes a new Adabas SVC. This SVC is fully backward compatible. In other words, you can use the new Adabas 8 SVC with Adabas 7 (or earlier) databases.

> **Note:** You cannot use the Adabas SVC from previous Adabas releases with Adabas 8 databases. If you attempt to do this, the Adabas 8 database will not initialize successfully.

On z/OS systems, the new SVC includes performance improvements and improved error recovery routines. Note that the new SVC shifts work from SRB-mode routines to TCB-mode routines. Take this information into account when analyzing Adabas 8 SVC performance. With the new SVC, SRB-mode overhead is largely eliminated and TCB-mode overhead is somewhat increased, but the net result is an overall improvement in SVC performance.

The new SVC on z/OS systems requires that the Adabas nucleus, as well as other MPM servers (such as Entire Net-Work and the Natural Global Buffer Pool), be APF-authorized. This APF authorization prevents use of the Adabas SVC by unauthorized programs that try to set themselves up as servers (which can receive and respond to calls from client programs). Software AG highly recommends that you run APF-authorized because of the security risks you can incur if you do not. However, upon request, Software AG does have a zap you can apply that eliminates this requirement. Contact your Software AG support representative for assistance.

> **Note:** Some add-on products require APF authorization to use restricted z/OS services. APF authorization is still required in these cases.

Utility Enhancements

In general, all Adabas utilities have been updated to support the new extended features in Adabas 8. Some of this support appears in the form of new or modified utility parameters. In other cases, support was added internally that does not affect your use of the utility at all.

The following table describes the Adabas utilities whose user interface has changed in this release. For complete information about the functions of any Adabas utility, read *Utilities* in *Adabas Utilities Manual*.

Utility	Summary of Changes
ADAACK	Spanned record support is provided in the ADAACK utility. However, ADAACK assumes that an ISN passed to it is a primary ISN or is the only ISN for a record. If an ISN is the primary ISN for a spanned record, all of the associated segment records of the spanned record are automatically checked in the secondary address converter.
	When printing error information about a particular ISN, the ADAACK utility will now indicate whether the problem is with a primary or secondary ISN, if the record is spanned.
ADACDC	Spanned records are not supported by the ADACDC utility at this time. However, when the IGNORESPANNED parameter is specified in an ADACDC run, ADACDC processing ignores spanned records, issues a warning message, and continues its processing. A return code of "4" is returned.
ADACMP	The length of the internal format buffer created by ADACMP from the FORMAT parameter for the COMPRESS or DECOMPRESS function is no longer limited to 64K bytes.
	The following *new* parameters have been added to the ADACMP COMPRESS utility to support the MU/PE extension, spanned records, and LB fields:
	■ The DATADEVICE parameter specifies the Data Storage device type to be used for the segmentation of spanned records. If the SPAN parameter is specified, ADACMP will break long, spanned, compressed records into segments that are just a bit smaller than the Data Storage block size implied by the DATADEVICE parameter.
	If DATADEVICE is specified without the SPAN parameter, it is used to derive the maximum compressed record length that will be accepted. Longer compressed records will be considered in error and will be written to the DD/FEHL data set.
	■ The HEADER parameter indicates whether or not the ADACMP compression logic should expect segmented ADACMP record headers in the uncompressed input records. The default is NO. (Contrast this with the HEADER parameter introduced in this release for ADACMP DECOMPRESS.)
	■ The LOBDEVICE parameter identifies the device type that will be used for loading the *LOB file* produced by the COMPRESS function.
	■ The LOBVALUES parameter indicates whether the uncompressed input data may contain long LB values (larger than 253 bytes). If LOBVALUES=YES is specified (allowed only for files with LB fields), ADACMP writes long large object (LB) field values to a second sequential

Utility	Summary of Changes
	output data set (identified by the DD/AUSB1 JCL control statement), which is to be loaded into the LOB file accompanying the base file loaded from the first output data set. ■ The `MAXLOGRECLEN` parameter can be used to specify the size, in bytes, of a buffer used by ADACMP to assemble the physical uncompressed segmented records into logical compressed records. This buffer is allocated and used only if HEADER=YES is also specified. ■ The `MUPEX` parameter indicates whether the extended MU/PE limits are allowed for the file. If this option is not specified, the maximum number of MU fields and the maximum number of PE fields that can be specified is 191. ■ The `MUPECOUNT` parameter specifies the size of the value count field in the input record for the COMPRESS function . Valid values are "1" or "2". If "1" is specified, each value count field preceding the MU or PE values in the input data must be one byte with a value of no more than "191". If "2" is specified, each value count field preceding the MU or PE values in the input data must be two bytes. A value count may exceed 191 only if the `MUPEX` parameter is also specified. ■ The `SPAN` parameter allows the record to be spanned after it is compressed if the compressed record exceeds the data storage block size of the device. The following *changes* have been made to existing ADACMP COMPRESS parameters: ■ The report produced by the `DEVICE` parameter now includes an indication of whether or not the `MUPEX` parameter has been set for a file. ■ The syntax of the `FNDEF` parameter has been changed, to allow you to specify the number of occurrences of the MU and PE options. You can also now specify the NB and LB field options. ■ The MAXPE191 parameter is no longer supported for the ADACMP COMPRESS function in Adabas 8. When specified, a warning message is issued and processing continues. ■ If USERISN is specified with HEADER=YES, the ISN immediately follows the ADAH header as part of the logical record. ADACMP DECOMPRESS processing supports the extended MU and PE limits and spanned records as input. The size of the value count preceding MU or PE values in each decompressed output record depends upon the extended MU and PE support for a file. A two-byte value count is given for files with extended MU and PE support. A one-byte value count is given for files without extended MU and PE support. In addition, the following functionality is supported. ■ The decompression of LB fields is supported. A new LOBVALUES parameter has been added. If the value of this parameter is set to "NO", only LB fields up to 253 bytes can be output from ADACMP DECOMPRESS processing. If the value of this parameter is set to YES, any LB field can be output from ADACMP DECOMPRESS processing. If "YES" is specified for the LOBVALUES parameter, the INFILE parameter must also be specified to identify the file number of the *base file* of a *LOB file group* whose data is to be decompressed. During processing, as ADACMP DECOMPRESS reads and decompresses the records from the base file, it reads any referenced LB field values from the *LOB file*. ■ A new HEADER parameter has been added to the ADACMP DECOMPRESS utility to indicate whether or not the ADACMP decompression logic should produce the ADACMP segmented record headers (ADAH and ADAC) as part of the decompressed output. The default is NO.

Utility	Summary of Changes
	■ A new `MAXLOGRECLEN` parameter can be used to specify the size, in bytes, of a buffer used by ADACMP to assemble logical records that span one or more physical records of uncompressed output data. This buffer is allocated and used only if HEADER=YES is also specified. ■ The ISN parameter has been altered so that if it is specified with HEADER=YES, the ISN immediately follows the ADAH header as part of the logical record. Traditionally, the DD/FEHL error data set produced for ADACMP errors has truncated rejected records that exceeded the FEHL physical record length. In Version 8, the rejected records are segmented instead of truncated. Because of this change, the DD/FEHL LRECL setting must be at least 500 bytes.
ADADBS	Three new database services have been added to the ADADBS utility to support the extended MU/PE field counts and record spanning: ■ The MUPEX function allows you to specify the maximum number of occurrences allowed for MU or PE fields in a file. ■ The RECORDSPANNING function activates the use of spanned records in a file. ■ The SPANCOUNT function counts the spanned records in a file. In addition, large object (LB) fields can be added to the FDT using FNDEF definitions in the ADADBS NEWFIELD function. Finally, a new RESETPPT function resets the PPT blocks on the ASSO data set.
ADADCK	The ADADCK utility checks the header of a spanned record for plausibility, as follows: ■ The ISNs listed in the header are verified. The header contains the ISN of the primary record in the spanned record chain, the ISN of the previous spanned record in the chain, and the ISN of the next spanned record in the chain. ■ The spanned record identifier bits are checked. The header of a spanned record includes bits that indicate whether the record is a primary or secondary spanned record. Only one of these bits can be turned on in the header of any spanned record. A new MAXPISN parameter has also been introduced that allows you to set the maximum number of primary ISNs that will be checked for a spanned Data Storage file. The default is 1000.
ADAFRM	The ADAFRM utility can now be used to clear multiple PLOG headers from the PLOG, without requiring that you reformat the entire PLOG. To do this, you should specify the NUMBER parameter in conjunction with the FROMRABN parameter and set the SIZE parameter to "1". In addition, this utility can now handle the increased number of physical Associator and Data Storage extents (99) allowed in Adabas 8.
ADAICK	If you run the ADAICK DSCHECK function, the primary and secondary ISNs are now identified in the output.
ADALOD	The following new parameters are introduced for the ADALOD LOAD and UPDATE functions in support of spanned records and their associated secondary address converter:

Utility	Summary of Changes
	■ The AC2RABN parameter allows you to specify or update space allocation for the secondary address converter. The secondary address converter is used to map the secondary ISNs of secondary spanned records to the RABNs of the Data Storage blocks where the secondary records are stored.
	■ The optional MAXISN2 parameter allows you to specify the desired size of the secondary address converter (AC2) in ISNs. The secondary address converter is used to map secondary ISNs of secondary spanned records to the RABNs of the Data Storage blocks where the secondary records are stored.
	The following new parameters and parameter values are introduced for the ADALOD LOAD functions in support of large object (LB) fields and their associated *LOB files*:
	■ The optional LOBFILE parameter allows you to specify the file number of the *LOB file* associated with a *base file*. This parameter is used when loading base files.
	■ The optional BASEFILE parameter allows you to specify the file number of the *base file* associated with a *LOB file*. This parameter is used when loading LOB files.
	■ A new file type, LOB, specified on the FILE parameter, allows you to indicate that you are loading an Adabas LOB file with a predefined FDT.
ADAORD	To support spanned records, the following two new parameters have been added to the REORASSO, REORDB, REORFASSO, REORFILE, and STORE functions of ADAORD.
	■ The AC2RABN parameter allows you to specify the beginning RABN for the file's secondary address converter extent.
	■ The optional MAXISN2 parameter allows you to specify the desired size of the secondary address converter (AC2) in ISNs.
	The secondary address converter is used to map the secondary ISNs of secondary spanned records to the RABNs of the Data Storage blocks where the secondary records are stored.
ADAREP	The report produced by ADAREP now lists whether the MUPEX and spanned record options have been set for the database.
	In the "Contents of Database" section of the report, the following changes have been made:
	■ The padding factor has been removed from this table to make room for the larger extent values. However, it is still shown in the individual file details also provided in the report and it appears again if LAYOUT=1 is specified.
	■ When LAYOUT=1 is specified during report creation, the larger extent values, supported by Adabas 8, appear for each file.
	■ If a file is unable to build at least ten further file extents, ADAREP marks the file with an asterisk (*) on the right.
	In the "File Options" section of the report, a "T" indicates that two-byte MU/PE indexes are active for the file and an "S" indicates that use of spanned records has been activated for the file. In addition, the "Contains LOB Fields" column indicates whether the file contains one or more LB fields (an "L" appears if it does) and the "LOB File" column (the last column) indicates whether

Utility	Summary of Changes
	or not the file is a LOB file (an "L" appears if it is). Note that these two LB field columns are mutually exclusive; only one of them will be marked. In the "Physical Layout of the Database" section of the report, secondary address converter extents (for spanned records) are identified as "AC2" in the "Table File Type" column. In the "File Information" section of the report, a new field called "Two Byte MU/PE" indicates whether two-byte MU/PE indexes are active for the file. In the same section, the highest, maximum expected, and minimum secondary ISN are given as well as a new field called "Spanned Rec Supp", which indicates whether or not spanned records are activated for the file. In addition, the "Contain LOB Fields" field indicates whether or not the file contains one or more LB fields and the "LOB File" field indicates whether or not the file is a LOB file. In the Space Allocation section of the report, secondary address converter extents (for spanned records) are identified as "AC2" in the "List Type" column. Finally, three new checkpoints may be written by the Adabas nucleus: 75, 76, and 77.
ADASAV	The following new parameters are introduced for the ADASAV RESTONL FMOVE and ADASAV RESTORE FMOVE functions in support of spanned records and their associated secondary address converter: ■ The AC2RABN parameter allows you to specify the starting secondary address converter RABN for each file specified by FMOVE. ■ The MAXISN2 parameter allows you to specify the new number of secondary ISNs to be allocated for each file specified by FMOVE.
ADASEL	While there are no new parameters, ADASEL has been updated to support the MU/PE extension. In particular, when specifying indexes for MU or PE fields in an ADASEL SELECT IF statement, the indexes can now range from "1" through "65,534". Prior releases of Adabas restricted these indexes to values ranging from "1" through "191". ADASEL recognizes spanned records in its processing, but it cannot process files containing spanned records.
ADAULD	While there are no new parameters, ADAULD has been updated to support the MU/PE extension, LB fields, and and spanned records. In particular, two new statistics listing the number of record *segments* that were read and written during the run are produced by the ADAULD utility.

ADARUN Parameter Enhancements

The following ADARUN parameters have been added with Adabas 8:

■ The CLOGLAYOUT parameter has a new valid value, 8, which supports the Adabas 8 command long format. For complete information, read *CLOGLAYOUT : Command Logging Format* , in *Adabas Operations Manual*.

To support this new parameter, a new DSECT called LORECX that describes the CLOGLAY-OUT=8 record layout is also provided. For additional information, read *Command Log Formats* in *Adabas DBA Tasks Manual*.

■ Three new LOGGING parameters have been added to support logging of Adabas 8 ABDs and multifetch and user buffers. These subparameters are LOGABDX, LOGMB, and LOGUB.

■ A new CLOGBMAX parameter has been added. You can use this parameter to specify the maximum size of a logged buffer. If a buffer is longer, the logged buffer is truncated from the point at which its size exceeds the setting of the CLOGBMAX parameter. The CLOGBMAX setting affects the ADARUN LOGGING parameter specifications for both CLOGLAYOUT=5 and CLOGLAYOUT=8.

■ A new CLOGMAX parameter has been added. You can use this parameter to specify the maximum size of all logged buffers used by an Adabas command. When the sum of sizes of the logged buffers for an Adabas command reaches the value of the CLOGMAX parameter, the buffer exceeding the limit is truncated and all following buffers are omitted. The CLOGMAX setting affects the ADARUN LOGGING parameter specifications for both CLOGLAYOUT=5 and CLOGLAYOUT=8.

■ Prior versions of Adabas tied the use of EXCP or EXCPVR for database I/Os to the APF-authorization of the load library. EXCP was always used when running non-APF-authorized; EXCPVR was always used when running APF-authorized. If you wanted to use EXCP when running APF-authorized, you were required to apply special A$- or AY- zaps.

This release introduces a new ADARUN parameter, EXCPVR, available in z/OS environments. Using this parameter, you can specify whether EXCP or EXCPVR is used when running APF-authorized. For complete information on the use of this parameter, read *EXCPVR : Control EXCP or EXCPVR Use*, in *Adabas Operations Manual*.

Finally, updates to the old A$- or AY-zaps will no longer be provided, as the zaps are no longer necessary.

■ A new LNKGNAME parameter has been added. You can use this parameter to specify the name of the link globals table that should be used by an Adabas 8 batch/TSO link routine to obtain default information and to invoke any linked exits.

> **Note:** If you specify the `LNKGNAME` parameter, you must also specify the ADARUN `DBID` and `SVC` parameters.

■ A new valid value, RENTUSER, has been added to the PROGRAM parameter. You can use this value to specify that a user program is to be run using a reentrant Adabas batch/TSO link routine.

■ On BS2000 systems, a new parameter. SUBMPSZ. has been added that allows you to specify the common memory pool size in bytes for subtasks in products such as Adabas Review and Adabas Parallel Services.

For complete information about any ADARUN parameter, read *Adabas Initialization (ADARUN Statement)*, in *Adabas Operations Manual*.

Command Log Timestamp Changes

Timestamps in an Adabas 8 command log created using CLOGLAYOUT=8 are stored in machine time (GMT), whereas CLOGLAYOUT=5 timestamps are stored, as always, in local time. The LORECX record layout includes a differential time field that stores the difference between machine time and local time at the time the CLOG record is written. This field allows you to calculate the local time of a command log record. Because of the difference in timestamp formats, we do not recommend that you mix or merge command logs created using different CLOGLAYOUT settings (and, in fact, Adabas does not allow this in cluster or parallel services environments). For more information, read *CLOGLAYOUT : Command Logging Format* , in *Adabas Operations Manual*.

PRILOG Print Program Updates

The parameters for the PRILOG print program have changed:

▪ A new valid value, 8, has been added to the CLOGLAYOUT parameter. This value supports command logs created with ADARUN CLOGLAYOUT=8.

▪ New valid values (ACBX, MB, PB, and VERB) have been added to the FIELDS parameter. These valid values support the new and extended features of Adabas 8.

▪ A new parameter, DIMENSIONS has been added. This parameter allows you to specify the format of the printed output.

▪ New command selection parameters have been added that allow you to filter the PRILOG output.

For complete information about the PRILOG print program, read *PRILOG : Printing the Command Log* , in the *Adabas DBA Tasks Manual*.

User Exit 11 and Sample Exit UEX11UX1

A new user exit 11 and the sample exit UEX11UX1 are supported in Adabas 8. Sample user exit UEX11UX1 can be used in front of your existing user exit 1 to get Adabas 8 to invoke the user exit 1 as user exit 11. This sample user exit can be used only if the direct call uses the ACB direct call interface, not the ACBX direct call interface. For more information, read *User Exit 1 (General Processing)* in the *Adabas User, Hyperdescriptor, and Collation Descriptor Exits Manual*.

The new user exit 11 provides all the support of user exit 1, but for both ACB and ACBX-type direct calls. The only difference is that user exit 11 processes copies of the Adabas structures rather than the original structures themselves. For more information about user exit 11, read *User Exit 11 (General Processing)* in *Adabas User, Hyperdescriptor, and Collation Descriptor Exits Manual*.

Long Alpha (LA) Field Changes

Long alpha (LA) fields have been updated in Adabas 8 to include much of the support available for large object (LB) fields.

The following updates have been made to long alpha (LA) fields in this release:

- LA fields can now specify fixed field lengths greater than 253 in format buffers. For more information, read *Format Buffer Changes*, in the *Planning Manual for Adabas 8*.

- The new asterisk (*) field length specification is supported for LA fields in format buffers. For more information about the asterisk field length specification, read *Asterisk (*) Length Notation*, in the *Adabas Command Reference Guide*.

- The new format buffer indicator (*L*), referred to as the *length indicator*, can now be used to retrieve or specify the actual length of an LA or LB field value. For more information, read *Length Indicator (L)*, in the *Adabas Command Reference Guide*.

- For multiple value LA fields and LA fields within periodic groups, you cannot specify the base field without an occurrence index or with a "1-N" index. You must use a specific index or index range. For example, if L2 is an LA field with the MU option, the following format buffer specifications are *not* valid:

```
FB='L21-N.'
```

```
FB='L2.'
```

However, the following format buffer specification is valid, requesting the first three values of field L2:

```
FB='L21-3.'
```

I/O Optimization

In z/OS environments, various enhancements have been made to the Adabas 8 I/O routines, aimed at optimizing system resource use. Effective with this release of Adabas, Format-1 CCWs are used for I/O requests, whenever they are supported by z/OS (which includes all releases of z/OS with EXCPVR and z/OS 1.6 or later with EXCP). This assists in reducing the constraint on storage below the 16M line, since the I/O requests now occupy storage frames above the 16M line instead.

Wherever possible, Adabas I/O control blocks have been moved above the 16M line.

Finally, page-fixing requirements for each Adabas nucleus or utility using EXCPVR have been reduced because the amount of storage required to be fixed in Adabas 8.1.1 during each I/O request has been minimized. In addition, you can control how page-fixing occurs when EXCPVR is in use using the ADARUN PGFIX parameter introduced in Adabas 7.4.4. For complete information on the use of this parameter, read *PGFIX: EXCPVR Page Fixing Control*, in *Adabas Operations Manual*.

Installation Updates

While fully linked teleprocessing (TP) and batch monitor routines are provided with default values in Adabas 8, a new link globals module and associated LGBLSET macro have been introduced as well to tailor the installations of the TP and batch monitors under Adabas 8. This module and macro allow you to set the default values for the link routine components without editing Adabas 8 source and greatly simplifies the installation of the TP and batch monitors. In addition, various new modules have been added to support the Adabas 8 link routines.

To ensure that your existing Adabas 7 applications will still function in Adabas 8, the original Adabas 7 TP and batch modules are provided in Adabas 8, but with new names. You can continue to tailor these as you have in past releases.

> **Note:** All Adabas 8 CICS TP monitors use the task-related user exit (TRUE). The only non-TRUE installation you can perform is for Adabas 7 releases.

For more information about the TP and batch monitor installations, read *Installing Adabas With TP Monitors*, in *Adabas z/OS Installation GuideS*.

New Licensing Component

A new licensing component is provided. This component cross-checks an assigned license key with your software installation and CPU IDs, and issues warning messages if inconsistencies are found. As a result of this new licensing component, nucleus startup changes are also necessary.

For complete information on licensing Adabas, read *Software AG Product Licensing*, in the *Adabas Operations Manual*.

BS2000 Enhancements

The default device type for BS2000 in Adabas 8 is 2002. This device type is compatible with NK4 disks, whereas the old default device type of 2000 does not. In addition, the license file is assembled into a library which is in the nucleus BLSLIB chain.

Finally, a new ADARUN parameter was added for BS2000 environments: SUBMPSZ. This parameter allows you to specify the common memory pool size in bytes for subtasks in products such as Adabas Review and Adabas Parallel Services.

Hyperexit Support of Extended MU/PE Fields

Updated hyperexit logic is provided in this release to support the extended MU/PE fields in hyperdescriptor specifications.

In addition, Adabas 8 includes a Hyperexit Stub that allows your existing hyperexits to use the Adabas 8 parameter list without change. The Hyperexit Stub is intended as a temporary solution for those customers who do not wish to immediately update their hyperexits to use the new Adabas 8 parameter areas. For more information about all hyperexit support in Adabas 8, read *Hyperdescriptor Exits 01 - 31* in *Adabas User, Hyperdescriptor, and Collation Descriptor Exits Manual*.

6 Dropped Features

The following features have been dropped from Adabas in version 8:

■ The Adabas 7 link routine modules listed in the table below are no longer provided as of Adabas 8.1.3.

> **Note:** The use of these modules in Adabas 8 is still supported at this time. If you used these modules in prior versions of Adabas 8, you may continue to use them for the present. However, we recommend that you migrate to the Adabas 8 link routines as soon as you can.

Platform	Link Routine Module	Description
IMS TM	ADALNI7	Link routine for message processing programs (MPPs).
CICS	LNKOLSC	Dependent part of the Adabas command-level link routine.
	LNKOLM	Independent part of the Adabas command-level link routine.
	LNKTRUE	Adabas task-related user exit.
	LNKENAB	Adabas PLT-enabled program.
	DISPGWA	Display program for Adabas global work area (GWA).
Com-plete	ADALCO7	Base module
Batch/TSO	ADALNK7	Base module.
	ADALNKR7	Reentrant module.

■ The `MAXPE191` parameter of ADACMP COMPRESS is no longer supported. When specified, a warning message is issued and processing continues. For more information about ADACMP COMPRESS changes, read *ADACMP COMPRESS Changes*, in *Planning Manual for Adabas 8*.

■ Prior to Version 8, you could define Adabas containers on VSAM RRDS or linear data sets. Support for VSAM containers is no longer available.

■ Unlike the link components in prior releases, the teleprocessing and batch link components of Adabas 8 are not provided in source code. These components are provided only in load module form and have new installation and customization procedures.

■ IBM does not recommend CICS macro-level programming with their latest releases of VSE. For this reason, Software AG no longer distributes the CICS macro-level link routine, ADALNC in the VSE Adabas or WAL sublibraries. Customers still running CICS/VSE 2.3 or earlier may want to retain a copy of ADALNC from the Adabas 7.1.3 or WAL 7.1.3 sublibraries for use in these older releases of CICS/VSE.

■ In BS2000 environments, no source module for ADALNK is delivered with Adabas 8.1. However, the new ADALNK module is comprised of the components ADALNK, ADAL2P, and LNKUES and can be used for binding modified ADALNK components.

■ ADARUN CLOGLAYOUT=4 settings will no longer run, and setting 4 is no longer supported. Use CLOGLAYOUT=5 instead.

■ Support for user exit 1 is dropped, as you are encouraged to use user exit 11 instead. To ease migration, you can insert sample user exit UEX11UX1 in front of your existing user exit 1 to have it invoked as user exit 11.

7 Limitations and Restrictions

The following limitations and restrictions exist in Adabas 8. Enhancements to resolve these limitations may be considered in a future release.

1. At this time the LUIDX and LUIXNAM parameters of the LGBLSET macro (used to set default installation values for the Adabas link routines) are not supported, even though you may see them in the sample members in installation. The documentation for these parameters has been removed for now.

2. The following restrictions and limitations apply to large object (LB) fields in this release:

   - While LB fields are supported in Adabas 8, you can only store and retrieve an entire LB field in this release. The ability to read or write an LB field from left to right with multiple calls is an option being considered for a future Adabas version.

   - At this time, character conversion of LB field values from one code page to another is not supported. This functionality may be considered in a future release.

   - Some utility parameters are not supported for files containing LB fields. For more information, refer to the documentation for the utility in *Adabas Utilities Manual* and to the utility limitations and restrictions, provided later in this chapter.

   - At this time, large object (LB) fields can be define only with format A.

3. The new format buffer length indicator is only supported for LA and LB fields. Future versions of Adabas will consider supporting the specification of the length indicator for other fields too. For more information about the format buffer length indicator, read *Length Indicator (L)*, in the *Planning Manual for Adabas 8*.

4. At this time, when specifying an Adabas buffer description (ABD), you must specify the same value for the number of bytes sent to Adabas (ABDXSEND field) as you specify for the maximum buffer size (ABDXSIZE field).

5. The prefetch feature is not supported in ACBX interface direct calls -- it will not support ACBX calls with multiple buffers; you should use the multifetch feature instead. However, the prefetch feature still supports ACB interface direct calls.

6. At this time, system files do not support spanned records or the extended MU and PE field counts.

7. At this time, fields defined with the NB option must also be defined with either the NU or NC option.

8. The following restrictions and limitations apply to spanned records in this release:

   ■ The ADACDC utility does not support spanned records at this time.

   ■ The ADAULD utility does not support spanned records on ADAULD SAVETAPE runs.

   ■ At this time, ADAM files do not support spanned records.

   ■ System files do not support spanned records at this time.

   ■ The number of records that comprise a spanned record is limited. The Adabas nucleus allows up to five physical records (one primary record and four secondary records) in a spanned record. If you need more space, try relocating the Data Storage of the file to a different device type with a larger block size.

   For more information about spanned record support in Adabas 8, read *Spanned Record Support*, in the *Adabas Concepts and Facilities Manual*.

9. At this time, Adabas Review Pulse reports do not support ADARUN CLOGLAYOUT=8.

10. The following table lists restrictions and limitations of the Adabas 8 utilities:

Utility	Restrictions or Limitations
ADACDC	At this time, ADACDC cannot process spanned records. When spanned records are updated, only the changed records are written to the PLOG and WORK data sets. Since ADACDC presents the changes of the record in external form, it needs to decompress the logical record starting from the beginning. However, if some early piece is missing, the decompression will not work.
ADACMP	At this time, LB fields cannot be specified in the FORMAT parameter for either ADACMP COMPRESS or ADACMP DECOMPRESS.
ADACNV	You cannot use ADACNV CONVERT to convert a database directly from Adabas 7.1 to Adabas 8. To do this, you must first convert the database to Adabas 7.2 or 7.4 and then convert it to Adabas 8. Likewise, you cannot use REVERT to revert a database directly from Adabas 8 to Adabas 7.1; instead, you must first revert the database to Adabas 7.4 and then from 7.4 to 7.1. If a database makes use of any of the following extended features of Adabas 8, ADACNV will not allow you to REVERT the database to a version prior to Adabas 8: ■ More than five ASSO, DATA, or DSST extents ■ More than five file extents ■ Files that allow spanned records ■ Files that allow more than 191 MU and PE occurrences ■ Files that make use of LB fields

Utility	Restrictions or Limitations
	■ Files with fields that have the NB (no blank compression) option ■ System files with two-byte file numbers. If you want to complete the backward conversion (reversion), you must first remove any file with these new features from the Adabas database. The use of the following other new features provided in Adabas 8 do not prevent backward conversion to Adabas 7.4, but, of course, the new features cannot be used in Adabas 7.4: ■ Adabas commands issued via the ACBX interface (for example, with long or segmented buffers) ■ Commands using the new format buffer features (for example, the length indicator). **Note:** There may be files in a database that are not loaded but that have a Field Definition Table (FDT) stored in the FDT blocks. If ADACNV encounters such FDTs while converting a database to Version 8, they are deleted as part of its cleanup processing.
ADALOD	The MIXDSDEV parameter is not supported in an ADALOD LOAD run if the file you are loading is LOB file or contains spanned records or extended MU or PE fields. Spanned records are based on the data storage block size. If you use a mixture of device types (as allowed by the MIXDSDEV parameter), your spanned records would end up with different record lengths, which is not allowed. You cannot use the DDISN or DELISN parameters in an ADALOD UPDATE function to delete records in a *LOB file*. However, you can use these parameters against a *base file* of a *LOB group* unless one or more of the records to be deleted contain references to LOB values longer than 253 bytes which are stored in the *LOB file* (ADALOD will terminate with an error if such a request is made).
ADAULD	For this release, you cannot unload from a SAVETAPE if the file you are trying to unload contains spanned records.

8 Adabas Data Set Compatibility

Generally, Adabas utilities accept sequential input data sets that were produced as output data sets by utilities of the same version. Utilities of Adabas 8 also accept input data sets produced by utilities of versions prior to Adabas 8. However, utilities for versions prior to Adabas 8 cannot generally work with input data sets produced by Adabas 8 utilities.

Importing Files

A file can be imported (loaded, stored, or restored) into a database running with an earlier Adabas version as long as it does not use features that are supported only in a later Adabas version.

Save Data Sets

Generally, restoring a whole database is possible only with the same Adabas release used for creating the save data set. Restoring individual files is possible with the same or any later Adabas release used for creating the save data set.

Using the ADASAV utility of Adabas 8, you can restore files into an Adabas 8 database from a database save or file save data set created with the ADASAV utility from any prior Adabas version (for example, Adabas 7). Likewise, using the ADAREP or ADAULD utilities of Adabas 8, you can print a report or unload a file from a database save or file save data set created with the ADASAV utility from any prior Adabas version. However, you can only restore files from an Adabas 8 database save or file save data set into a database running with Adabas 7.4 if you have applied one of the following zaps to your Adabas 7 installation, as appropriate:

- For Adabas 7.4.4 installations, apply zap AU744032.
- For Adabas 7.4.3 installations, apply zap AU743099.
- For Adabas 7.4.2 installations, apply zap AU742186.

You cannot use an ADAREP or ADAULD utility from an Adabas version prior to Adabas 8 to print a report or unload a file from a save data set created by the ADASAV utility of Adabas 8. If you are using Adabas Delta Save, you cannot merge together delta save data sets created by different versions of the ADASAV utility. After the database has been converted to version 8, the next save operation must be a full save.

Unload Data Sets

Using the ADALOD utility of Adabas 8, you can load a file from an unload data set created using the ADAULD or ADACMP utilities from any prior version of Adabas. However, you can only use the ADALOD or ADACMP utilities from Adabas 7 to load or decompress a file from an unload data set created by the Adabas 8 ADAULD or ADACMP utilities if you have applied one of the following zaps to your Adabas 7 installation, as appropriate:

- For Adabas 7.4.4 installations, apply zap AU744033.
- For Adabas 7.4.3 installations, apply zap AU743100.
- For Adabas 7.4.2 installations, apply zap AU742187.

ADAORD DD/FILEA Data Sets

You can restructure databases and files from an Adabas version prior to Adabas 8 and store them in an Adabas 8 database using ADAORD STORE. However, you cannot store the restructured output of an Adabas 8 database or file in a database running with any prior Adabas version (for example, Adabas 7). If you attempt this, the following warning will be generated and ADAORD will end with a CC=4:

```
*** Warning: The input dataset is from V8 and will not be processed
```

Sequential Protection Logs

Any sequential protection log (PLOG) used for a REGENERATE or BACKOUT function under Adabas 8 must have been created with Adabas 8. Using a sequential PLOG created with Adabas 8 for a REGENERATE or BACKOUT function under any prior Adabas 8 version is not supported.

You cannot run the ADASEL utility of Adabas 8 to select data from a sequential PLOG created by any Adabas version prior to Adabas 8. In addition, you cannot run the ADASEL utility of a prior Adabas version (for example, Adabas 7) to select data from a PLOG created by Adabas 8.

You also cannot run the ADARES COPY utility of Adabas 8 to copy data from a sequential PLOG created by any Adabas version prior to Adabas 8. In addition, you cannot run the ADARES COPY utility of a prior Adabas version (for example, Adabas 7) to copy data from a PLOG created by Adabas 8.

9 Applying Zaps

⚠️ **Important:** Be sure that you apply all supplied Adabas maintenance and concatenate Adabas patch-level libraries (L00*n*), as they are delivered to you. This will ensure that your Adabas code remains up-to-date, supporting all Adabas features as they are enhanced and maintained. The latest zaps for this product are available in the Knowledge Center in Software AG's Empower (*https://empower.softwareag.com*) web site.

In general, zaps for Adabas components (such as Adabas nuclei, the Adabas router, Adabas utilities) can be applied and made active one component at a time.

- Adabas utility zaps should be applied to the load library. The utility can then be run or rerun to make use of the zap.

- Adabas nucleus zaps should first be applied to the load library. Then the nucleus should be stopped and restarted to activate the zap.

- Adabas router (on z/OS, ADASVC) zaps should first be applied to the load library. Then all Adabas nuclei and other MPM servers running on the router should be stopped and the router should be reinstalled to activate the zap. Finally, the Adabas nuclei and MPM servers should be restarted.

Finally, the distributed source library contains member ZAPOPT, which lists some optional zaps that you may choose to apply for the activation or deactivation of various features and optional user settings of Adabas. A ZAPOPT member will be included with each SM level distribution.

10   Software AG Product Compatibility

The following table describes Adabas 8 compatibility with other Adabas products, including prior releases of Adabas itself. You may need to upgrade your installation of the software if your existing release is not listed.

Product	Compatible Version Levels and Notes
Adabas (ADA)	When working with Adabas 8 databases, the Adabas 8 SVC, nucleus, and utilities must be used. The Adabas 8 utilities and nucleus cannot be used when working with Adabas 7.4 databases. However, the Adabas 8 SVC can be used for Adabas 7.4 databases as well. The Adabas link (Adalink) routines can be used across versions. For example, Adabas 8 link routines can be used to issue calls to Adabas 7 databases. Unless the new ACBX direct call interface is used, Adabas 7.4 link routines can be used to issue calls to Adabas 8 databases. Software AG recommends that you use the Adabas 8 link routines for all programs that issue Adabas direct calls. On VSE systems, the following zaps must be applied (as appropriate) so the Adabas 8 SVC can communicate with Adabas 7.4 or Adabas 7.1 databases: ■ AD744008 ■ AD743018 ■ AD713039
Adabas Bridge for DL/I (ADL)	Version 2.3.1 supports Adabas 8.
Adabas Bridge for VSAM (AVB)	Version 5.1.1 releases support Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8.

Product	Compatible Version Levels and Notes
Adabas Caching Facility (ACF)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. Adabas Caching Facility 8.1 must be run with Adabas 8. Likewise, Adabas Caching Facility 7.4 must be run with Adabas 7.4. You cannot run Adabas Caching Facility 7.4 with Adabas 8.
Adabas CICS Interface (ACI)	Version 7.4 releases support Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8. Version 8.1 and above fully support all Adabas 8 databases and expanded features.
Adabas Cluster Services (ALS)	Version 8.1.3 requires Adabas 8.1.3 or above. Version 7.4.2 requires Adabas 8.1.2 or below. When run together with Adabas 8.1.2, it fully supports the new features of Adabas 8.1.
Adabas Delta Save (ADE)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. Adabas Delta Save 8.1 must be run with Adabas 8. Likewise, Adabas Delta Save 7.4 must be run with Adabas 7.4. You cannot run Adabas Delta Save 7.4 with Adabas 8.
Adabas Fastpath (AFP)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. In addition, the 7.4 version of Adabas Fastpath will subsequently be supplied with optional fixes to allow it to be used with Adabas 8 databases that do not make use of the Adabas 8 expanded features (for example, spanned records, increased limits, or large object fields).
Adabas IMS Interface (AII)	Version 7.4 releases support Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8. Version 8.1 and above fully support all Adabas 8 databases and expanded features.
Adabas Manager (AMA)	Version 7.4 releases support only Adabas 7 databases. A future release will fully support Adabas 8 databases and expanded features.
Adabas Native SQL (SQL)	Version 2.3.1 supports Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8. Full support for Adabas 8 is planned.
Adabas Online System (AOS)	Version 8.1 and above fully support all Adabas 8 databases and expanded features. You can also access Adabas 7.4 databases using Adabas Online System Version 8, but you cannot access Adabas 8 databases using Adabas Online System 7.4.
Adabas Parallel Services (ASM)	Version 8.1.3 requires Adabas 8.1.3 or above.

Product	Compatible Version Levels and Notes
	Version 7.5.1 requires Adabas 8.1.2 or below. When run together with Adabas 8.1.2, it fully supports the new features of Adabas 8.1.
Adabas Review (REV)	Version 4.3.2 and above partially support all Adabas 8 databases and expanded features. For a complete description of the compatibility between Adabas Review and Adabas 8, read the section entitled *Adabas Review 4.4 Version Compatibility* in the Adabas Review 4.4 Release Notes. **Note:** You must apply all maintenance to fully enact this support. Download the latest Adabas Review zaps from Empower before attempting to use Adabas 8 features in any Adabas Review release.
Adabas SAF Security z/OS (AAF)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. In addition, the 7.4 version of Adabas SAF Security will subsequently be supplied with optional fixes to allow it to be used with Adabas 8 databases that do not make use of the Adabas 8 expanded features (for example, spanned records, increased limits, or large object fields).
Adabas Statistics Facility (ASF)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. It also supports Adabas 7 databases. Version 7.1 does not support Adabas 8 databases.
Adabas SQL Gateway (ACE)	Version 6.5.1 supports Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8. Full support for Adabas 8 databases and expanded features is planned.
Adabas SQL Server (ESQ)	Version 1.4.3 supports Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8.
Adabas System Coordinator (COR)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. In addition, the 7.4 version of Adabas System Coordinator will subsequently be supplied with optional fixes to allow it to be used with Adabas 8 databases that do not make use of the Adabas 8 expanded features (for example, spanned records, increased limits, or large object fields).
Adabas Text Retrieval (TRS)	Version 2.1.4 works with Adabas 8 when the Adabas Text Retrieval 2.1.4 hyperexit TRSHEX12 is enabled to run with the Version 8 interface by applying zap TR21454. This hyperexit will then only operate with Adabas Version 8. If you then want to run Adabas Text Retrieval 2.1.4 with Adabas Version 7 again, you must either undo the zap or use a copy of the hyperexit where zap TR21454 is not applied. If you do use TR21454, the following additional Adabas Text Retrieval fixes must be applied as prerequisite zaps: TR21420, TR21421, TR21422, TR21423 and TR21424. These zaps can be found in ServLine 24.

Product	Compatible Version Levels and Notes
	It is not necessary to use the Adabas 8 Hyperexit Stub in conjunction with Adabas Text Retrieval.
Adabas Transaction Manager (ATM)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. In addition, the 7.5 version of Adabas Transaction Manager will subsequently be supplied with optional fixes to allow it to be used with Adabas 8 databases that do not make use of the Adabas 8 expanded features (for example, spanned records, increased limits, or large object fields).
Adabas Vista (AVI)	Version 8.1 and above fully supports all Adabas 8 databases and expanded features. In addition, the 7.4 version of Adabas Vista will subsequently be supplied with optional fixes to allow it to be used with Adabas 8 databases that do not make use of the Adabas 8 expanded features (for example, spanned records, increased limits, or large object fields).
Entire Net-Work (WCP)	Version 6.1 and above fully support Adabas 8 databases and expanded features, as well as ACBX interface direct calls. Version 5.9 releases support Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8. In addition, only ACB interface direct calls are supported in version 5.9 releases; ACBX interface direct calls are not.
Entire System Server	Entire System Server Version 3.4 fully supports Adabas 8 databases and expanded features.
Event Replicator for Adabas (ARK)	Version 3.1 and above supports Adabas V81 databases.
Natural (NAT)	Version 4.2, with Entire System Server 3.3 fully supports Adabas 8 databases and expanded features. Version 4.1, with zap NA64099 applied (z/OS systems only), supports Adabas 8 databases that do not make use of the expanded features (for example, spanned records, increased limits, or large object fields) available in Adabas 8.
Predict	Version 4.5 fully supports Adabas 8 databases and expanded features. Version 4.4 does not support Adabas 8 databases.
EntireX Communicator/webMethods EntireX	Version 7.2.1 and later fully support the Adabas 8 SVC and the use of an Adabas 8 database as a persistent store.

11 End of Maintenance

For information on how long a product is supported by Software AG, access Software AG's Empower web site at *https://empower.softwareag.com*.

Log into Empower. Once you have logged in, you can expand **Products** in the left menu of the web page and select **Product Version Availability** to access the Product Version Availability application. This application allows you to review support information for specific products and releases.

12 Documentation

The documentation for this product is new with this release. If you have an Empower account, current, updated, and past versions of the documentation can be reviewed and downloaded by linking to the Software AG documentation, found on Software AG's **Empower** web site. If you do not have an Empower user ID and password yet, you will find instructions for registering on this site (free for customers with maintenance contracts).

This documentation includes:

- online HTML topics describing all aspects of the product;
- Adobe Acrobat Reader Portable Document Format (PDF) files created from the HTML topics;
- Adobe Acrobat Reader Portable Document Format (PDF) files for a series of manuals created from the HTML topics.

No hard-copy documentation is provided, but you can print the PDF and HTML files on your local printer.

Allowing Active Content in Windows Environments

With Service Pack 2 (SP2) for Windows XP and Service Pack 1 (SP1) for Server 2003 and later Windows releases, Microsoft introduced a range of powerful new security features that restrict active content that runs locally on your computer. Active content includes ActiveX controls, Java applets, and JavaScript. Software AG's documentation web pages contain some JavaScript, and the SEARCH, INDEX and CONTENTS capabilities are implemented as Java applets. As a result, when viewing documentation web pages that reside on your PC using Internet Explorer and Mozilla Firefox under Windows XP SP2, note that active content is blocked. You must explicitly and repeatedly allow active content if you want to make use of the documentation's full navigation features. Note that this behavior is only observed when reading web pages installed locally on your PC, including those on CD in the PCs CD-ROM drive.

The active content for which Software AG is responsible, that is, the JavaScript code in our HTML documentation pages, will not harm your computers. The risk in using the navigation applets is negligible: Software AG has received no reports from users concerning any harm caused to a computer by the applets. We therefore suggest that when reading Software AG documentation in a local context, you should allow active content via the Security settings in the browser (with Internet Explorer, usually found under Tools > Internet Options > Advanced).

Full details of alternatives can be found on the home page of the suppliers of the navigation applets: *http://www.phdcc.com/xpsp2.htm*.

Index

C

checkpoints
    written by Adabas nucleus/utilities, 27

D

documentation
    allowing active content in Windows, 52
    obtaining updates, 51

E

Empower
    documentation, 51
    end-of-maintenance dates, 49
    platform support, 3
end-of-maintenance dates, 49

P

platform support, 3
product support
    end-of-maintenance dates, 49
    obtaining updated documentation, 51
    supported platforms, 3

S

support
    end-of-maintenance dates, 49
    obtaining updated documentation, 51
    platforms supported, 3
support dates, 49
supported platforms, 3