Adabas Vista

Adabas Vista Introduction

Version 8.1.1

November 2016

This document applies to Adabas Vista Version 8.1.1.

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

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

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

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

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

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

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

Document ID: AVI-OS-INTRO-811-20161128

Table of Contents

Preface v
1 File Partitioning 1
When is Partitioning Required? 2
Implementing Partitioning in the Application 2
Implementing Partitioning using Adabas Vista 2
Typical Adabas Vista Deployment Scenarios 2
2 File Translation 5
Translation Rules 6
Profile IDs and Target Categories 6

Preface

Adabas Vista provides file partitioning and file translation capabilities.

- Data can be partitioned across files without increasing the complexity of the application.
- Access can be translated to different files without the need to modify the application.
- Large volumes of data to be handled with increased flexibility and availability.
- Provides for growth by allowing the addition of new partitions or splitting of existing partitions that have grown too large.

File Partitioning

File Translation

1 File Partitioning

- When is Partitioning Required? 2
- Implementing Partitioning in the Application 2
- Implementing Partitioning using Adabas Vista 2
- Typical Adabas Vista Deployment Scenarios 2

When is Partitioning Required?

File partitioning may be useful for any of the following reasons:

- data volumes are too large to be managed within a single file;
- separate files need a consolidated view as well as an individual view;
- data archiving is needed with an efficient, high-performance recall capability.

Implementing Partitioning in the Application

The burden of partitioning has traditionally fallen on the application, making them extremely complex. For example, the application often has to determine the partitions targeted by an access or update, or to consolidate results from queries against several partitions.

Apart from the increased complexity, the more significant problem is that the usual level of application independence from the physical data model is lost. The cost of development and ongoing maintenance in these situations can be enormous.

Implementing Partitioning using Adabas Vista

Adabas Vista allows the application data to be partitioned into separate Adabas files without having to re-engineer the applications that access this data. This means that the application continues to refer to an Adabas file as though it is a single logical entity, even though the physical data may be partitioned and distributed across multiple files, databases or machines.

Typical Adabas Vista Deployment Scenarios

This section shows typical deployments of Adabas Vista:

Very Large Files

Very large files present an obvious use for partitioning. Although Adabas can hold a massive number of records within a file, the difficulties in managing such large amounts of data remain. The key to managing mass data is the ability to divide the data into more easily managed units.

In a distributed system, partitioning large files can bring benefits in terms of performance or resource utilisation. Partitioning data across multiple machines can help with load balancing and localisation of data. Overall data availability can also be improved because Adabas Vista has a partition outage feature that allows the application to continue working even though some of the partitions may not be available.

Consolidation

Applications and their data often change over time. Sometimes it becomes necessary to merge multiple files into a single file. This can present difficulties if the files are different or if the ability to access the original files must be retained.

Adabas Vista has a consolidation feature that allows a single file image to be imposed on multiple, previously unrelated files. Direct access to the individual files is maintained, and even though the files may be different, they support the same consolidated view.

Archiving

Many organisations have a requirement to keep data for a specified period of time. The sheer amount of data will often force an organisation to archive in order to keep the volume to a manageable level. If there is a need to access the archived data, then a process must also be established to recall it.

Most archiving systems access old data, copy it to archive, and delete it from the current file. This can take a considerable amount of time and may interfere with normal operation or access to the file.

Adabas Vista offers a simple and flexible approach to archiving and recalling data. Partitioning by date means that new data can be directed towards new partitions. Partition criteria might be based, for example, on handling a complete month or year's worth of data. Old partitions can be backed-up and removed completely or access to them reduced by altering the partition definitions. Should access to the older data be required, the old file can simply be restored and the partition definition altered to make it available to the application again.

2 File Translation

- Translation Rules 6
- Profile IDs and Target Categories 6

Translation Rules

Adabas Vista provides a file translation feature that can be used to change file numbers dynamically within Adabas applications at runtime. The file numbers are translated based on the translation rules defined in Adabas Vista.

Many organisations have a requirement that an application can use either test or production data without requiring that the application itself be modified. The ability to do so reduces the risk involved in modifying applications immediately prior to their being placed into production mode.

For example, a Natural program is compiled in a test environment using a test DDM to access file 25 in database 10. After testing has been completed, the program is to be placed into production using file 67 in database 41. By providing this information to Adabas Vista using translation rules, the switch to production mode can be made without any application modification.

Profile IDs and Target Categories

Translation rules are extended further by the use of Profile IDs and Target Categories. Profile IDs allow sets of rules to be grouped together. Target Categories allow rules for the same source file to be translated to different target files.

Profile IDs are typically used to group rules on the basis of business area or employee role. Target Categories are often used to switch between different versions of data as an application progresses from development into production.