Data Archiving for Adabas

Getting Started

Version 1.8.1

September 2017

This document applies to Data Archiving for Adabas Version 1.8.1 and all subsequent releases.

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

Document ID: ADR-GETTING-STARTED-181-20170930

Table of Contents

Preface

This document provides a general orientation to Data Archiving for Adabas.

Loading and Using the Data Archiving for Adabas Demo

1 Loading and Using the Data Archiving for Adabas Demo

Data Archiving for Adabas contains a getting started demo to help you quickly see how to use the product. This demo is real; it runs against test files that are created as part of the demo itself. The following sections explain how to use the demo.

Prepare the Demo Databases

The default demo execution uses databases 1 and 2. It is much easier to achieve a quick “getting started” if you can arrange to use these database numbers for the demo; you can use different database numbers if you wish.

Also, we assume it is easier for you to create and use completely new databases for the demo - to stay away from your day-to-day data while you are getting started. But you do not have to create new databases either, you can use existing ones.

Assuming your environment is new for the purposes of getting started we suggest you create two new databases with the numbers 1 and 2 (if possible).

Depending on your environment, you can use the Adabas Manager, Adabas Online Services, DBA Workbench or even utilities to create these databases. Or you may have two test databases with space available for creating some (small) test files in them. Note the database numbers, you will need them later.

Note: These test files do not need to be part of a database that is secured with PLOG, and they do not need to be backed up everyday, because the demo has a reload capability that you can use at any time to recreate the files in full.

Loading (and Reloading) the Demo

The Demo is not automatically loaded during the installation, you choose whether you want to load (or reload) it.

The Load/Reload consists of a number of processes (some of which can be selectively controlled):

- The load/reload of the Demo configuration data into the Repository file
- The load/reload of the Demo Source Files
- The identification of a Demo Target File

To load/reload the Demo environment, login into System Management Hub and perform the following steps.

To Load or Reload the Demo:

1 Select the *Data Archiving for Adabas* node. The Data Archiving for Adabas window will appear.

2 Click the *Demo* button and the Demo Load/Reload window will appear.

Enter the following information (or accept the defaults):

Section	Description
Demo	A tick in the checkbox (the default) specifies that the Demo configuration objects will be loaded/reloaded into the Repository. De-select if the Demo configuration is not to be loaded/reloaded. **Note:** For the initial installation, this checkbox must be ticked.
Computer	Select the Computer on which the Demo will be configured to run.
Plan	Specify the Plan name (default is DEMO) under which the configuration data will be loaded/reloaded. **Note:** Each time the configuration objects are loaded/reloaded into the specified Plan, they are dynamically adjusted to reflect a configuration for running on the Computer that you have selected. This means a multiple computer implementation (where the Repository is shared) may be re-loading the configuration so that the Computer running the Demo changes from one reload to the next. You can negate this effect by defining a different Plan name for each different computer that you want to run the Demo on.
Actions	The Demo creates three different Actions each using a different Data Selection Mode. The default names are non-modifiable.
Source Files	The Demo uses three Source Files called PEOPLE, CARS, and OTHER. Specify the database and file number where these Demo files are to be loaded/reloaded. A tick in the *Create/Reload File* checkbox on the right hand side (the default) specifies that these files are to be loaded/reloaded. Use this checkbox to selectively load/reload each individual file or de-select all files to avoid any load/reload. **Note:** For the initial installation, this checkbox must be ticked.
Target Files	The Demo uses a single Target File called PEOPLE_2. Specify a database and file number which can be used by the Demo for this Target File. The file will be created automatically if it has not been previously used by the Demo.
Activities	A tick in the *Delete Recently Completed activities* checkbox (the default) specifies that all previous Completed Activities involving the selected Computer will be deleted.

3 Click the *Execute* button to start the demo load/reload.

4 Check the *status* of the demo load/reload as follows:

The Source Files are loaded by performing an automatic recall from an archived copy of the Demo files kept within the product itself. This means you can actually watch the recall Activities running while the load/reload takes place. To watch this happen, do the following

- Select the Services by (*group:computer:daemon*) node within the Data Archiving for Adabas navigation tree.
- Within that, select the *DEFAULT:<computer>:DAEMON* node where *<computer>* is the computer you selected
- Within that, select the *Pending Activities*, *Current Activities* or *Completed Activities* nodes to follow the progress of the recall activities which have been automatically submitted to perform the Source File loads/reloads.
- Within the *Completed Activities* node you will see the completed recall Activities.

5 The demo is now loaded/reloaded and you are ready to run some Demo Actions.

Running the Demo Actions

The demo contains three sample Actions SYNTAX, TODOLIST and USERLIB that demonstrate the three Data Selection Modes (see Data Selection Mode).

Action SYNTAX: Uses Extraction Syntax for data selection

This action uses Extraction Syntax to select records for archive or transfer.

To display the Extraction Syntax used by Action SYNTAX:

1 Select the *Plans* node within the Data Archiving for Adabas navigation tree.

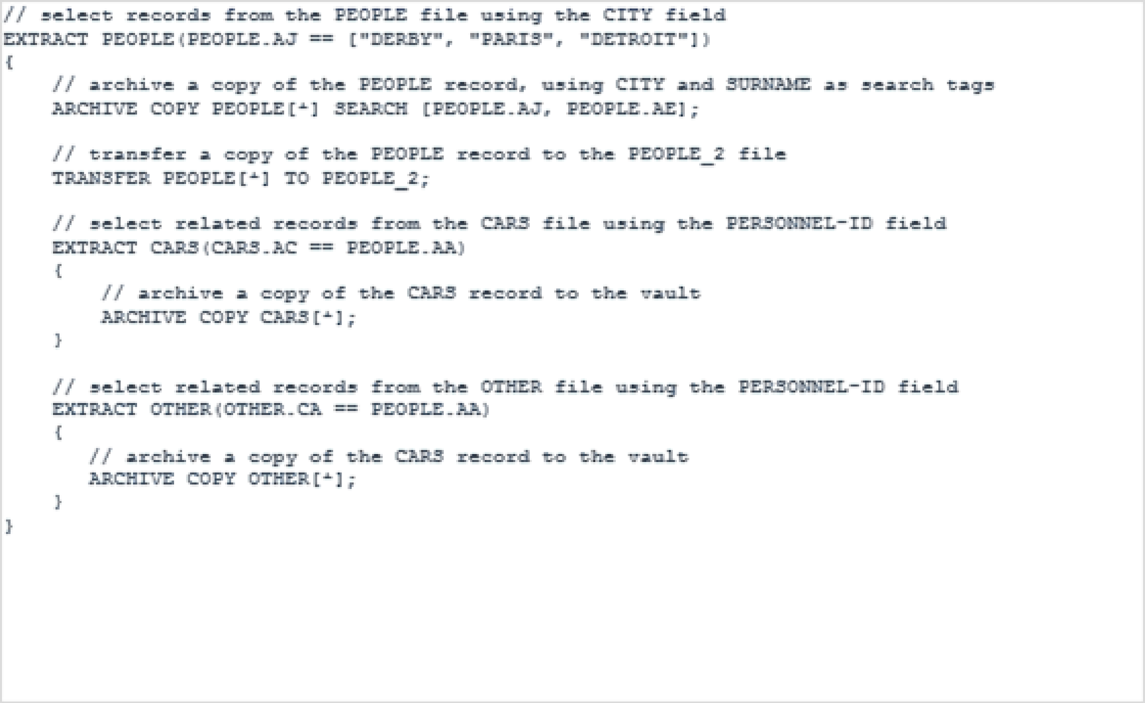
2 Select the Plan called DEMO (or whatever name you chose when performing the Demo Load/Reload).

3 Select the *Actions* node within the selected Plan.

4 Select the Action called SYNTAX.

5 Select the *Extraction Syntax* node within the selected Action. The Modify Extraction Syntax window will appear.

The following Extraction Syntax is displayed:



```
// select records from the PEOPLE file using the CITY field
EXTRACT PEOPLE(PEOPLE.AJ == ["DERBY", "PARIS", "DETROIT"])
{
    // archive a copy of the PEOPLE record, using CITY and SURNAME as search tags
    ARCHIVE COPY PEOPLE[*] SEARCH [PEOPLE.AJ, PEOPLE.AE];

    // transfer a copy of the PEOPLE record to the PEOPLE_2 file
    TRANSFER PEOPLE[*] TO PEOPLE_2;

    // select related records from the CARS file using the PERSONNEL-ID field
    EXTRACT CARS(CARS.AC == PEOPLE.AA)
    {
        // archive a copy of the CARS record to the vault
        ARCHIVE COPY CARS[*];
    }

    // select related records from the OTHER file using the PERSONNEL-ID field
    EXTRACT OTHER(OTHER.CA == PEOPLE.AA)
    {
       // archive a copy of the CARS record to the vault
       ARCHIVE COPY OTHER[*];
    }
}
```

This Extraction Syntax selects records from the PEOPLE source file where the CITY field (AJ) is DERBY, PARIS or DETROIT.

For each of the matched PEOPLE records:

- A copy of the PEOPLE record is archived to the Vault.
- A copy of the PEOPLE record is transferred to the PEOPLE_2 target file
- Records are selected from the CARS source file where the PERSONNEL_ID (field AC) matches the PERSONNEL_ID (field AA) of the PEOPLE record. A copy of each of the matched CARS records is archived to the Vault.
- Records are selected from the OTHER source file where the PERSONNEL_ID (field CA) matches the PERSONNEL_ID (field AA) of the PEOPLE record. A copy of each of the matched OTHER records is archived to the Vault.

To run Action SYNTAX:

1. Refer to Running an Action Manually for information on how to run the Action. Each run of an Action is called an Activity.
2. The progress of the Activity can be monitored as follows:

    - Select the *Services by (group:computer:daemon)* node within the Data Archiving for Adabas navigation tree.

- Within that, select the *DEFAULT:<computer>:DAEMON* node where *<computer>* is the Computer you selected when performing the Demo Load/Reload.
- Within that, select the *Pending Activities, Current Activities* or *Completed Activities* nodes to follow the progress of the Activity.
- Within the *Completed Activities* node you will see the completed Activity.

Refer to Monitoring Activities for more information on how to monitor Activities.

To browse the Vault where Action SYNTAX has written archive data:

- Refer to Vault Browse for information on how a Vault's content can be browsed. Hint: If default values were used for the Demo load/reload then the archive files will have been written to a Vault called DEMO using a Group/Plan/Action of DEFAULT/DEMO/SYNTAX.

To recall archived data created by running Action SYNTAX:

1. Refer to Vault Browse for information on how a Vault's content can be browsed and how to find a particular archive to be recalled. Hint: If default values were used for the Demo load/reload then the archive files will have been written to a Vault called DEMO using a Group/Plan/Action of DEFAULT/DEMO/SYNTAX.
2. Refer to Recall Archived Data for information on the recalling of archived data.

Action TODOLIST: Uses the API for Natural to generate an archive/transfer list

This action selects records for archive or transfer from a To-Do List generated by a Natural application that utilises the Data Archiving for Adabas Application Programming Interface.

Use of this action requires an INPL of the Natural library SYSADR. Refer to Application Programming Interface for more information on installing and using this interface.

To run Action TODOLIST:

1. Select the Plans node within the Data Archiving for Adabas navigation tree.
2. Select the Plan called DEMO (or whatever name you chose when performing the Demo Load/Reload).
3. Select the *Actions* node within the selected Plan.
4. Select the Action called TODOLIST.
5. Select the *To-Do List* node within the selected Action to view the current status of the To-Do list. The status should show “No list available”.
6. From Natural, run the SYSADR/TODOLIST program.

Unlike the Action SYNTAX demo where the Action is run manually from the Web Interface, the TODOLIST demo is driven entirely from the Natural application using the Data Archiving for Adabas Application Programming Interface..

The TODOLIST Natural program selects records from the PEOPLE source file where the CITY field is DERBY, PARIS or DETROIT.

For each of the matched PEOPLE records:

- A copy of the PEOPLE record is archived to the Vault.
- A copy of the PEOPLE record is transferred to the PEOPLE_2 target file
- Records are selected from the CARS source file where the PERSONNEL_ID matches the PERSONNEL_ID of the PEOPLE record. A copy of each of the matched CARS records is archived to the Vault.
- Records are selected from the OTHER source file where the PERSONNEL_ID matches the PERSONNEL_ID of the PEOPLE record. A copy of each of the matched OTHER records is archived to the Vault.

Follow the on-screen prompts of the Natural program and it will guide you through the creation and build-up of a To-Do list, the automatic triggering of the archive process, and finally, a recall of the archived data.

Note: The TODOLIST Natural program recalls the archives back to database number 2 and file numbers 611, 612 and 613. If you wish to change the recall target files then locate the section */* Set up the Recall files* within the Natural program and edit the values appropriately before running the program.

To browse the Vault where Action TODOLIST has written archive data:

- Refer to Vault Browse for information on how a Vault's content can be browsed. Hint: If default values were used for the Demo load/reload then the archive files will have been written to a Vault called DEMO using a Group/Plan/Action of DEFAULT/DEMO/TODOLIST.

Action USERLIB: Uses a record selection plug-in (User Lib/DLL)

For this action, record selection for archive or transfer is performed by a plug-in user Lib/DLL. The Lib/DLL has been created using the API for the C Programming Language.

For further details on creating your own User Library refer to the supplied Data Archiving for Adabas Application Programming Interface, API for the C Programming Language and the section titled *User-Lib Interface*.

To view the real-time API and Parameters:

1 Select the *Plans* node within the Data Archiving for Adabas navigation tree.

2 Select the Plan called DEMO (or whatever name you chose when performing the Demo Load/Reload).

3 Select the *Actions* node within the selected Plan.

4 Select the Action called USERLIB.

5 Select the *Real-Time API Parameters* node within the selected Action to view the name of the User Lib that the Action will call (*userlib.dll*/*libuserlib.so*) and the parameters that will be passed to it. In the case of this Action, the string *DERBY* will be passed.

The User Lib *userlib.dll*/*libuserlib.so* selects records from the PEOPLE source file where the CITY field is DERBY (the parameter passed to the user library).

For each of the matched PEOPLE records:

- A copy of the PEOPLE record is archived to the Vault.
- A copy of the PEOPLE record is transferred to the PEOPLE_2 target file
- Records are selected from the CARS source file where the PERSONNEL_ID matches the PERSONNEL_ID of the PEOPLE record. A copy of each of the matched CARS records is archived to the Vault.
- Records are selected from the OTHER source file where the PERSONNEL_ID matches the PERSONNEL_ID of the PEOPLE record. A copy of each of the matched OTHER records is archived to the Vault.

To run Action USERLIB:

1 Refer to Running an Action Manually for information on how to run the Action. Each run of an Action is called an Activity.

2 The progress of the Activity can be monitored as follows:

- Select the Services by (*group:computer:daemon*) node within the Data Archiving for Adabas navigation tree.
- Within that, select the *DEFAULT:<computer>:DAEMON* node where *<computer>* is the Computer you selected when performing the Demo Load/Reload.
- Within that, select the *Pending Activities, Current Activities* or *Completed Activities* nodes to follow the progress of the Activity.

  Refer to Monitoring Activities for more information on how to monitor Activities.

To browse the Vault where Action USERLIB has written archive data:

- Refer to Vault Browse for information on how a Vault’s content can be browsed.

Hint: If default values were used for the Demo load/reload then the archive files will have been written to a Vault called DEMO using a Group/Plan/Action of DEFAULT/DEMO/USERLIB.

To recall archived data created by running Action USERLIB:

1 Refer to Vault Browse for information on how a Vault's content can be browsed and how to find a particular archive to be recalled.

Hint: If default values were used for the Demo load/reload then the archive files will have been written to a Vault called DEMO using a Group/Plan/Action of DEFAULT/DEMO/USERLIB.

2 Refer to Recall Archived Data for information on the recalling of archived data.