
AP Data Manager (APdm) Product Summary

Why is there a Need for AP Data Manager?

Your businesses data is one of the most valuable assets that it possesses. But what happens when your hardware engine begins to downgrade in performance simply because there is too much data stored on your live environment.

Business environments that utilise applications that store vital data such as WMS, ERP, TMS and CRM are often caught suffering from slow opening windows, extended search periods and sluggish interactive response times. Many businesses spend large amounts of money on hardware upgrades that may, in fact, be unnecessary.

AP Data Manager brings to your business the capability to efficiently purge unwanted data as well as archive the data that is crucial to your business in an easy to install and simple to administer format.

Overview – AP Data Manager

AP Data Manager integrates with existing technologies and enables schedule-based automation to simplify administration of database table volumes. Simply set up the Archive and Purge control tables with data retention periods by table name and AP Data Manager will do the rest.

AP Data Manager archives active online data and moves it to offline repositories without user intervention, thus alleviating business disruptions.

AP Data Manager is portable across your entire business topology. It is designed to allow you to run archiving and purging for any application existing on an Oracle (8i and above) or SQL Server Database (Version 7 and above). Once the AP Data Manager is installed on the host server you will be directed via a series of drop-down boxes to initiate the server and database schema for which archiving and purging rules are to be established.

AP Data Manager uses a front-end user interface to set up the control and business rules that will be used by the server components to either archive or purge your data. The interface may be installed on any Windows platform that has a network connection to your database. The user is required to enter the names of the parent tables and their related child tables. The interface also allows you to filter the data by specifying business rules that will be applied before archiving or purging the tables. This information will be stored in the AP Data Manager control tables and can be modified at any time.

Data is archived or purged based on a retention date. AP Data Manager allows you to derive the retention date based on several optional time periods:- number or days, number of calendar years, number of financial years, number of semesters, or simply specify a date.

The server component has been designed to utilise the power of the native database procedures. This capability maximises performance because the code is interpreted and executed on the more powerful back-end servers where the actual RDBMS database resides. All SQL code has been optimised and tuned for extremely high throughput, minimizing system resources and allowing you to concurrently run AP Data Manager with host applications.

AP Data Manager is also capable of scheduling the archive or purge process within the database engine. | Optionally, the same scheduling procedure can be set up using the operating system of the database server.

AP Data Manager will record the number of rows archived or purged so you may view this information after each run to verify that no errors occurred and all data volumes for each table balance against expected and actual deleted rows.

AP Data Manager Client Component

The client component is an easy-to-use interface developed with Microsoft .Net that allows the user to connect to the database instance where archiving or purging needs to be applied. It is an AP Data Manager prerequisite to use this component to set up the control tables that will be used by AP Data Manager to archive or delete your company's data. This task does require complete knowledge of the host application's database schema. After all, who knows the data better than the owners of the application? The control tables simply consist of "parent tables" and "child tables". The parent and child relationship is equivalent to the "one to many" relationship that exists between two or more tables in your database schema.

For each parent table that is entered, the user is required to add the name of the date field that the application should use when archiving or purging records. Data may be filtered out by adding a value in the Where Clause. This type of functionality gives you the flexibility to dynamically define the business rules if and when they change.

The screenshot shows the 'AP Data Manager (Local)' application window. The 'Parent Tables' tab is active, displaying a table with the following data:

Enabled	Schema Name	Table Name	Link Column	Action	DB Link
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	ORDERS	ORDERKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	WAVE	WAVEKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	DROPID	DROPID	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	RECEIPT	RECEIPTKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	POD	PO_ID	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	PO	POKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	ADJUSTMENT	ADJUSTMENTKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	TRANSFER	TRANSFERKEY	Archive	ARC1
<input type="radio"/> Yes <input checked="" type="radio"/> No	WH1	ITRN	ITRNKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	TRANSMITLOG	TRANSMITLOGKEY	Archive	ARC1

Below the table are buttons for 'Add', 'Remove', 'Save', 'Move Up', 'Move Down', and 'Run'. The 'Parent Details' tab is also visible, showing configuration options:

- Where Clause:** status = '9'
- Date Column:** EDITDATE
- Interval Type:** DAYS
- Interval to Keep:** (dropdown menu)

A 'Verify' button is located at the bottom left of the 'Parent Details' section.

If a parent table has a data relationship to other tables (one to many) you simply add this information to the Child Tables section. Again, you have the flexibility to filter out the data on the child tables by adding business rules in the Where Clause section.

AP Data Manager (Local)

Connection Configuration Help WH1

Parent Tables | Scheduling | DB Links | Logs | Verification

Enabled	Schema Name	Table Name	Link Column	Action	DB Link
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	ORDERS	ORDERKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	WAVE	WAVEKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	DROPID	DROPID	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	RECEIPT	RECEIPTKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	POD	PO_ID	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	PO	POKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	ADJUSTMENT	ADJUSTMENTKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	TRANSFER	TRANSFERKEY	Archive	ARC1
<input type="radio"/> Yes <input checked="" type="radio"/> No	WH1	ITRN	ITRNKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	TRANSMITLOG	TRANSMITLOGKEY	Archive	ARC1

Add Remove Save Move Up Move Down Run

Parent Details | Child Tables | Logs | Verification

Table Name	Where Clause	Link Column
ORDERDETAIL		ORDERKEY
PREALLOCATEPICKDETAIL		ORDERKEY
PICKDETAIL		ORDERKEY
LOTXIDHEADER	IOFLAG = '0'	SOURCEKEY
LOTXIDDETAIL	IOFLAG = '0'	SOURCEKEY

Add Remove Save Move Up Move Down

The data is archived or purge based on a retention date that is derived by the application based on the number of calendar years, number of financial years, number of semesters, number of days, or simply a specific date.

The screenshot shows the 'AP Data Manager (Local)' application window. The 'Verification' tab is active, displaying a table of database tables with their retention settings. Below the table are buttons for 'Add', 'Remove', 'Save', 'Move Up', 'Move Down', and 'Run'. The 'Parent Details' section shows a 'Where Clause' of 'status = '9'', a 'Date Column' of 'EDITDATE', an 'Interval Type' dropdown menu with 'FINANCIAL' selected, and a 'Keep from Date' of '11/03/2002'.

Enabled	Schema Name	Table Name	Link Column	Action	DB Link
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	ORDERS	ORDERKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	WAVE	WAVEKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	DROPID	DROPID	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	RECEIPT	RECEIPTKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	POD	PO_ID	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	PO	POKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	ADJUSTMENT	ADJUSTMENTKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	TRANSFER	TRANSFERKEY	Archive	ARC1
<input type="radio"/> Yes <input checked="" type="radio"/> No	WH1	ITRN	ITRNKEY	Archive	ARC1
<input checked="" type="radio"/> Yes <input type="radio"/> No	WH1	TRANSMITLOG	TRANSMITLOGKEY	Archive	ARC1

Parent Details

Where Clause:
 status = '9'

Date Column:
 EDITDATE

Interval Type:
 DATE
 DAYS
 YEARS
 SEMESTERS
FINANCIAL
 DATE

Keep from Date:
 11/03/2002

AP Data Manager Server Component

The server component has been developed to run on an Oracle RDBMS or Microsoft SQL Server. Both products have been developed using native stored procedures. This design gives maximum performance advantages over any other product. Many purge applications simply take too long to complete because they use third party applications that need to be parsed, interpreted, and then sent over a communications layer that simply takes too long or uses too much resources. AP Data Manager does not have this type of unnecessary overhead. The server component will be executed by the database engine that is located on your high-end server. Each statement has been fully optimized and tuned for both types of database engines.

The server component is designed to read the information entered into the control files by the AP Data Manager Interface and derive a retention date to apply to the data stored in the parent and child tables. When a table is to be purged, it simply deletes the data. When a table needs to be archived, it will first insert the data into the database specified by the database link and then delete the source data in one transaction using a two phase commit. This type of design ensures that data consistency is never compromised. Simply said, if the data cannot be inserted first, then the data will not be deleted from the source table.

The server component can be scheduled using the operating system scheduler. For example, on UNIX platforms, you can schedule the execution through the use of the crontab. However, the AP Data Manager Interface allows you to schedule the execution of the server component using the native RDBMS scheduler.

Benefits

AP Data Manager

- Easy to install
- Improves overall database performance
- Improves response times
- Improves productivity
- Improves storage utilization
- Reduces database tuning
- Reduces data storage costs
- Alleviates hardware upgrades

System Requirements

AP Data Manager Interface - Windows 2000 / Windows XP Home / Windows XP Pro / Windows Server
Optional Database Server for Archive database Oracle RDBMS / Microsoft SQL Server RDBMS