
AIMMS User's Guide - Advanced Data Management

This file contains only one chapter of the book. For a free download of the complete book in pdf format, please visit www.aimms.com or order your hard-copy at www.lulu.com/aimms.

Copyright © 1993–2011 by Paragon Decision Technology B.V. All rights reserved.

Paragon Decision Technology B.V.	Paragon Decision Technology Inc.	Paragon Decision Technology Pte.
Schipholweg 1	500 108th Avenue NE	Ltd.
2034 LS Haarlem	Ste. # 1085	80 Raffles Place
The Netherlands	Bellevue, WA 98004	UOB Plaza 1, Level 36-01
Tel.: +31 23 5511512	USA	Singapore 048624
Fax: +31 23 5511517	Tel.: +1 425 458 4024	Tel.: +65 9640 4182
	Fax: +1 425 458 4025	

Email: info@aimms.com
WWW: www.aimms.com

AIMMS is a registered trademark of Paragon Decision Technology B.V. IBM ILOG CPLEX and sc CPLEX is a registered trademark of IBM Corporation. GUROBI is a registered trademark of Gurobi Optimization, Inc. KNITRO is a registered trademark of Ziena Optimization, Inc. XPRESS-MP is a registered trademark of FICO Fair Isaac Corporation. MOSEK is a registered trademark of Mosek ApS. WINDOWS and EXCEL are registered trademarks of Microsoft Corporation. $\text{T}_{\text{E}}\text{X}$, $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, and $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ are trademarks of the American Mathematical Society. LUCIDA is a registered trademark of Bigelow & Holmes Inc. ACROBAT is a registered trademark of Adobe Systems Inc. Other brands and their products are trademarks of their respective holders.

Information in this document is subject to change without notice and does not represent a commitment on the part of Paragon Decision Technology B.V. The software described in this document is furnished under a license agreement and may only be used and copied in accordance with the terms of the agreement. The documentation may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior consent, in writing, from Paragon Decision Technology B.V.

Paragon Decision Technology B.V. makes no representation or warranty with respect to the adequacy of this documentation or the programs which it describes for any particular purpose or with respect to its adequacy to produce any particular result. In no event shall Paragon Decision Technology B.V., its employees, its contractors or the authors of this documentation be liable for special, direct, indirect or consequential damages, losses, costs, charges, claims, demands, or claims for lost profits, fees or expenses of any nature or kind.

In addition to the foregoing, users should recognize that all complex software systems and their documentation contain errors and omissions. The authors, Paragon Decision Technology B.V. and its employees, and its contractors shall not be responsible under any circumstances for providing information or corrections to errors and omissions discovered at any time in this book or the software it describes, whether or not they are aware of the errors or omissions. The authors, Paragon Decision Technology B.V. and its employees, and its contractors do not recommend the use of the software described in this book for applications in which errors or omissions could threaten life, injury or significant loss.

This documentation was typeset by Paragon Decision Technology B.V. using $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ and the LUCIDA font family.

Chapter 18

Advanced Data Management

The previous chapter discussed AIMMS' basic case management capabilities. These capabilities are sufficient for most projects. Understanding the advanced concepts introduced in this chapter will require a considerable time investment on your part. These concepts are only required when data efficiency and data security play a crucial role in your application.

Time investment required

This chapter introduces the more advanced concepts of *case types*, *data categories* and *datasets*. These concepts provide you with a flexible framework to create cases containing only a subset of the data in your model, to create data snapshots representing a particular functional aspect within a model, or to store common data, that is shared by multiple cases, at a single location. In addition, the chapter discusses advanced issues such as case security, AIMMS' facilities to import and export case data or to refer to the collection of cases and dataset from within the modeling language, and whether to use AIMMS cases or store data in a commercial database.

This chapter

18.1 Case types

A case type defines a *subset* of model identifiers which

- are to be stored in a case, and
- are sufficient to restore the global state of the application for a particular purpose.

What are case types?

You can use case types to decrease the (physical) size of cases on disk, when only a subset of all data is sufficient to restore the state of your model-based application. In addition, when the successful execution of your model consists of several phases (for data entry or for computing a (partial) solution), different case types can be used to create cases that contain only the data necessary to restart a particular phase of the modeling application.

The **Data Management Setup** window is the developer tool for specifying the collection of available case types which you think are relevant for your modeling application. As illustrated in Figure 18.1, the **Data Management Setup** window lets you manage a single list of case types. You can open it through

Data Management Setup

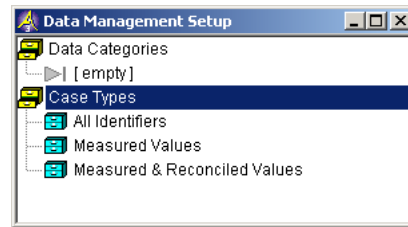


Figure 18.1: The **Data Management Setup** window

the **Tools-Data Management Setup** menu, or through the **Data Management Setup** button on the project toolbar.

Below the *Case Types* node in the **Data Management Setup** window you can add new case type nodes to the list of already available case types. With each case type, you can associate a collection of model identifiers. With any new project AIMMS will automatically create the case type *All Identifiers*, which represents the complete collection of model identifiers currently present in the model.

Creating case types

To associate model identifiers with a case type you can add a list of individual identifiers to the case type either

Adding model identifiers

- by dragging the identifiers from within the model explorer onto the case type node, or
- by modifying the properties of the case type.

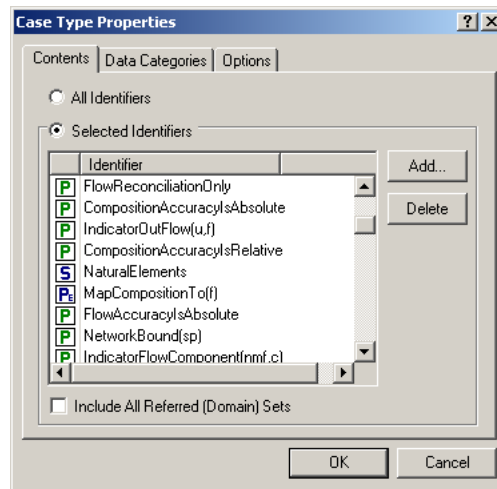
After you have added identifiers to a case type, you can always view or edit its current content in the case type **Properties** dialog box, as illustrated in Figure 18.2. When you select a (named declaration) section in this dialog box, you get the choice to implicitly or explicitly add all identifiers to the list that are contained in this section.

Cases in AIMMS represent a *global* state of your application for a particular purpose. AIMMS, therefore, only allows you to define case types for the main project. As a consequence, when adding identifiers to a case type, you can only add identifiers to the case type that

Case types and libraries

- are declared in the main project, or
- are part of the interface of a library included in the AIMMS project.

If your application requires that a case type also contains identifiers that are private to a library, you can accomplish this by defining a data category for

Figure 18.2: The case type **Properties** dialog box

that library containing the relevant private identifiers. Subsequently, you can add that data category to the case type, as discussed in Section 18.2.1.

When you save a case through the **Data-Save Case** or **Data-Save Case As** menus in the presence of multiple case types, the **Save Case** dialog box (illustrated in Figure 18.3) requires that you specify a case type in addition to the name of the case to be created. Similarly, the **Load Case** dialog box enables you to select from cases of all case types, or to filter on cases of a specific case type.

Selecting the case type

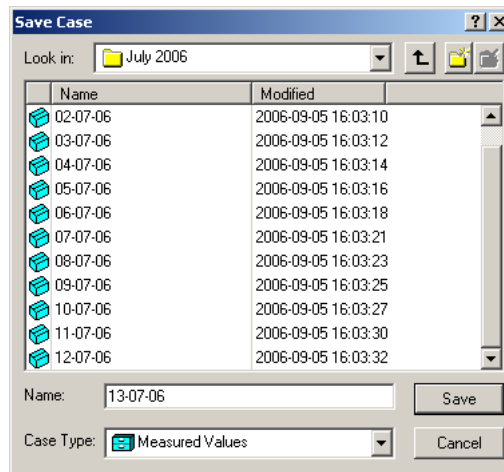


Figure 18.3: Saving a case with a specific case type

If you do not want your end-users to select a case type themselves either when saving or loading a case, you can preset the case type from within the AIMMS modeling language through the predefined element parameter `CurrentDefaultCaseType`. When this element parameter has a value, AIMMS will remove the case type drop-down list in the **Save Case** and **Load Case** dialog boxes, and use the case type specified through `CurrentDefaultCaseType` instead. For complete details on the parameter `CurrentDefaultCaseType` refer to Section 19.4.5.

Preset the case type

18.2 Data categories and datasets

A data category is a *subset* of model identifiers associated with a particular *functional aspect* of an application. For instance, you can create data categories that hold all identifiers defining the problem topology, or that define a supply and demand scenario within your application.

What are data categories?

A dataset is a *data instance* associated with a particular data category, similarly as a case is an instance of a case type. AIMMS lets you maintain multiple datasets with a data category, each dataset representing a particular version of the data. Thus, you can create datasets that define the problem topology for different regions, or that hold different scenarios for supply and demand.

What are datasets?

Data categories and datasets are in many aspects similar to case types and cases. The major difference is that a

Compare to case types and cases

- case type can include one or more data categories, and, as a consequence,
- cases can be built up from multiple datasets.

Section 18.2.1 discusses in detail how you can exploit this feature.

Similarly to case types, you must specify the data categories used in your model through the **Data Management Setup** tool (see also Section 18.1). Below the *Data Categories* node in the **Data Management Setup** window you can add a list of all the data category nodes that are necessary for your modeling application. In addition, AIMMS allows you to create data categories for all library projects included in your application. Figure 18.4 illustrates the list of data categories for the example application used throughout this User's Guide.

Creating data categories

You must associate a list of model identifiers with every data category, which together completely represent the particular functional aspect of the model expressed by that data category. There are two methods for associating model identifiers with a particular data category:

Adding identifiers

- you can simply drag a selection of identifiers from the **Model Explorer** (possibly obtained through the identifier selection tool) to the data category node, or

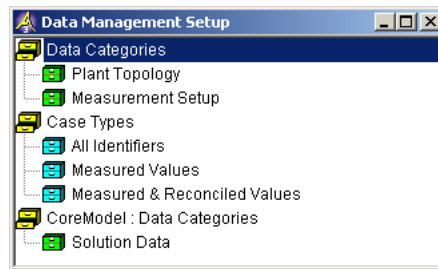


Figure 18.4: Data categories in the **Data Management Setup** tool

- you can open the **Properties** dialog box of the data category, and modify its contents in the **Contents** tab, as illustrated in Figure 18.5.

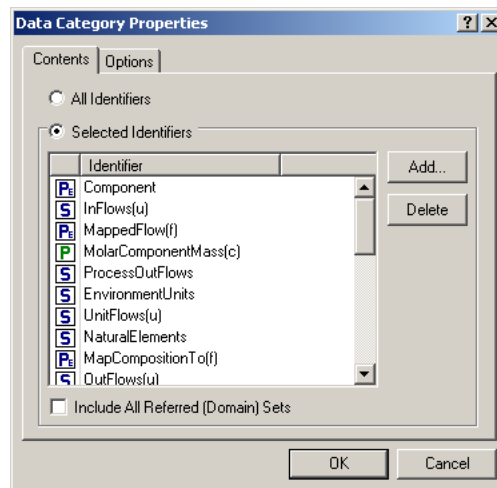


Figure 18.5: Data category **Properties** dialog box

After you have added identifiers to a data category, you can always view its current contents through the **Properties** dialog box of that data category.

For data categories associated with the main project, i.e. those listed under the *Data Categories* node in the **Data Management Setup** window, AIMMS only allows you to add identifiers that

Global data categories and libraries

- are declared in the main project, or
- are part of the interface of a library included in the AIMMS project.

This restriction ensures that a developer of a library project can change the internal implementation of the library without disrupting the case functionality of the main application.

If a data category is defined within a library project, you can add every identifier declared in the library to it. By including such a data category into a case type, as described in Section 18.2.1, you can ensure that the entire internal state of library, that is necessary to continue working with the functional aspect expressed by the data category, is saved in a case.

*Data categories
in libraries*

Whenever you have specified one or more data categories in the **Data Management Setup** tool, AIMMS will automatically add a *Datasets* root node to the **Data Manager**. Directly below the *Datasets* node, AIMMS will add a node for each data category that you have created with the **Data Management Setup** tool, whether defined in the main project or in any of its included library projects. If you have not (yet) created data categories, the *Datasets* node will not be present in the **Data Manager** at all. Figure 18.6 illustrates an example of the **Data Manager** associated with the data management configuration of Figure 18.4.

*Reflected in
Data Manager*

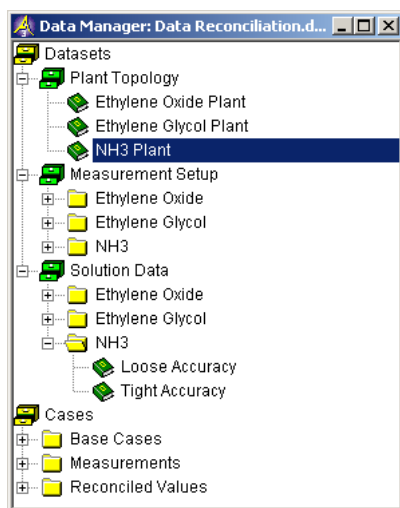


Figure 18.6: The AIMMS **Data Manager**

Below each of the data category nodes in the **Data Manager**, AIMMS allows you to add one or more datasets associated with that data category. In addition, AIMMS allows you to add additional folders below these nodes to provide further structure to a collection of datasets, as illustrated in Figure 18.6. However, unlike the collection of cases (which can be structured regardless of their case type) datasets are always strictly separated by data category in the **Data Manager**. Thus, you cannot move or copy datasets from one data category node to another, as these represent an entirely different subset of identifiers.

*Managing
datasets*

18.2.1 Using data categories and datasets in cases

In addition to constructing a case type from model identifiers only (see Section 18.1), AIMMS also allows you to build up a case type from one or more data categories, or even to combine both ways of constructing a case type. You can add a data category to a case type through the case type **Properties** dialog box, as illustrated in Figure 18.7. The dialog box will let you choose both

Case types and data categories

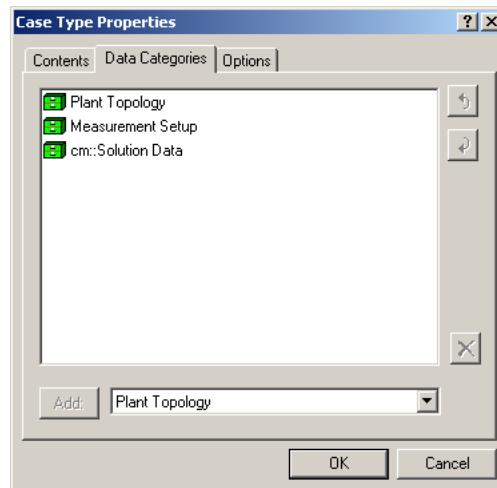


Figure 18.7: Case type **Properties** dialog box

from data categories defined in the main project and those defined in library projects.

With data categories in a case type, the complete collection of identifiers that is actually stored inside a case, consists of the union of

Case type identifiers

- the list of individual identifiers added to the case type, and
- the lists of identifiers associated with all data categories that have been added to the case type.

By default, AIMMS will store the data for all identifiers included in the case type in every case of that type.

However, when a case type contains one or more data categories, AIMMS also allows you to create *compound* cases that include references to associated datasets. As a result, AIMMS will no longer store the corresponding identifier values in the case itself, but in referenced datasets instead. Thus, dataset referencing allows the data stored in such a dataset to be shared by multiple cases.

Sharing data in cases

Both cases and datasets are stored on disk in a data manager file (see Section 18.6). The size of this file can be drastically reduced if you store shared data in datasets wherever possible, and include references to these datasets in your cases. As an example, if you have m topology datasets and n compatible supply and demand datasets, you can easily combine these datasets to create (input data for) $m \times n$ cases at almost no additional storage cost.

Sharing is space efficient

In addition to storage efficiency, the use of shared datasets between AIMMS cases will ensure that a change to an identifier in a shared dataset in a single case is automatically propagated to all other cases that include the same dataset as well. This prevents you from having to go through all the individual cases to repeatedly make the identical change.

Changes are shared

18.3 Working with datasets

In AIMMS, all data associated with the identifiers contained in a data category are referred to as the *active* dataset for that data category. If you have not yet loaded or saved a dataset for a particular data category, the associated active dataset is *unnamed*, otherwise the active dataset is *named* (after the name of the last loaded or saved dataset on disk).

Active datasets

When you save a named active dataset, AIMMS will save it to the associated dataset on disk by default (thus overwriting its previous contents). If an active dataset is unnamed, or when you try to save a dataset using the **Data-Save Dataset As** menu, AIMMS will open the **Save Dataset** dialog box illustrated in Figure 18.8. In the **Save Dataset** dialog box you must select the data category

Saving a dataset

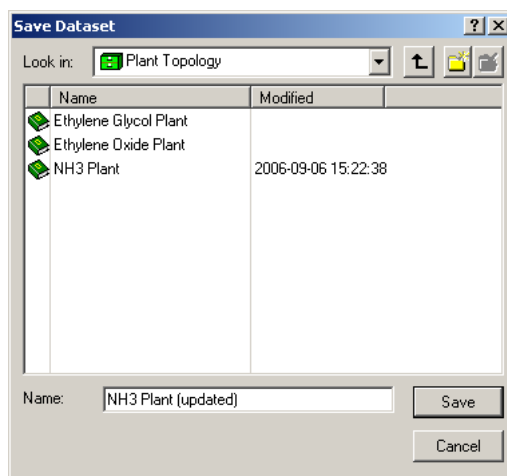


Figure 18.8: The **Save Dataset** dialog box

to be saved and enter the name of the associated dataset. Optionally, you can

select a folder below the data category in which the dataset is to be stored. After successfully saving a dataset through the **Save Dataset** dialog box, the active dataset will become named.

Through the **Data-Load Dataset** menu, AIMMS allows you to load the data associated with individual datasets into your model. As with loading the data of a case (see Section 17.2), AIMMS supports three modes of loading a dataset:

Loading a dataset

- as active,
- into active, and
- merge into active.

The actions of these modes of loading datasets are the same as for loading cases. As with cases, AIMMS will ask, before loading another dataset as active, whether the data in the current active dataset must be saved.

Within the **Data Manager** you can get an overview of the current (named) active case and datasets. As illustrated in Figure 18.9, AIMMS will display the current named active case and datasets in bold typeface.

View active case and datasets

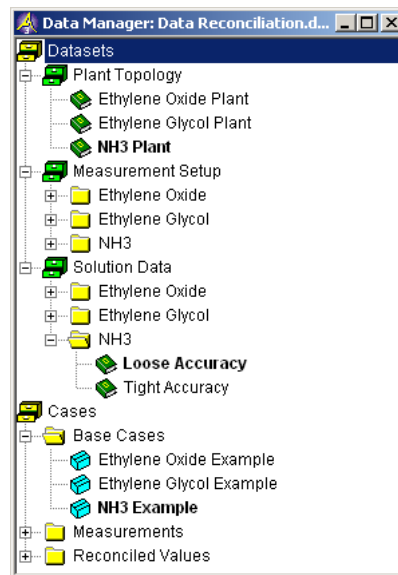


Figure 18.9: Active case and datasets

18.3.1 Datasets in compound cases

Whenever a case type is (partially) composed of one or more data categories, loading an associated compound case will affect the corresponding (named) active datasets, whereas saving a compound case will be affected by the current active datasets. This section will explain the fine details.

Compound cases

When you save a compound case, the active dataset for each included data category determines whether the corresponding data is saved on disk in a dataset or in the compound case itself. The following rules apply for each data category included in the case type.

Saving a compound case

- If an active dataset is named, the corresponding data is saved on disk in the named dataset, and a reference to the named dataset is stored in the compound case.
- If an active dataset is unnamed, the corresponding data is saved in the compound case itself.

If you use the **Data-Save Case As** menu to save a compound case, the **Save Case** dialog box only lets you specify the name of the compound case itself. If you want to store data category data in named datasets, and refer to these in the compound case, you must explicitly save these datasets through the **Data-Save Dataset As** menu before saving the case.

Case save as

When you load a compound case, the included data or dataset references in the compound case affect the active datasets after loading. The following rules apply for each data category included in the case type.

Loading a compound case

- If the compound case contains a reference to a dataset for the data category, then the data from that dataset is loaded, and the active dataset is named after that dataset.
- If the compound case itself contains the data for the data category, then the associated data is loaded from the case, and the active dataset becomes unnamed.

Whenever the case type and/or the data categories in that case type refer to the same identifiers, you should be aware that the order of loading is as follows.

Overlapping data

- First, the data stored in the compound case is loaded.
- Hereafter, the data of the included datasets are loaded using the order of the data categories as they appear in the **Data Management Setup** tool.

For all data categories not included in the compound case type, the associated active datasets are not changed by loading the compound case. You should note, however, that part of their data may be overwritten when the sets of identifiers associated with both a data category and the compound case type are overlapping.

Remaining datasets

In the **Data Manager**, you can modify the dataset references stored in a compound case, using the **Data** tab in the **Properties** dialog box of the case, as illustrated in Figure 18.10. Changing these dataset references is only useful if

Modifying dataset references

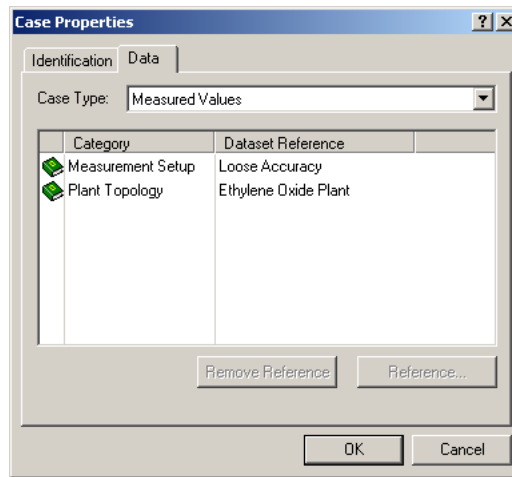


Figure 18.10: The **Data** tab in the case **Properties** dialog box

you intend to subsequently load the case using the newly specified datasets. *It has no effect on saving the compound case, since AIMMS will always overwrite the dataset references in a compound case according to the current active datasets, as described above.*

Modifying the dataset references manually through the **Properties** dialog box is particularly useful in combination with the possibility to create new cases by simply inserting new case nodes to the **Data Manager** tree. Combining both features allows you to quickly compose a large number of scenarios which all consist of input data taken from various combinations of existing datasets.

Composing scenarios

18.4 Advanced use of the Data Manager

In addition to viewing or modifying the contents of a data category or case type, the **Properties** dialog boxes for data categories and case types also let you optionally specify a user-defined *load procedure*. Through such a load procedure (which should only be needed for extraordinary case management

Load procedures

tasks) you can perform any further (implied) initialization statements that are necessary when a dataset or case has been loaded.

Each load procedure should have a single argument, an element parameter in either the predefined set `AllCases` or `AllDatasets`, referring to the case or dataset currently being loaded. The wizard used to select a load procedure in the **Properties** dialog box, will automatically restrict the choices to those procedures that match the required prototype.

*Required
prototype*

The load procedure is also called upon starting a new case or dataset. In that event, the argument of the load procedure refers to the empty element. You can use this feature, for instance, to empty the contents of the active case or dataset.

*New case or
dataset*

If you need further information on the specific data file passed to a load procedure, you can use the functions described in Section 19.4.5 to obtain such information about a data file. Through these functions you can get information such as the name of the data file, whether it is a case or a dataset, or any included datasets (if the data file is a case).

Datafile support

If there are load procedures for both a case type and its associated data categories, then, when a case of that type has been loaded, AIMMS will only execute the case load procedure. If you want the dataset load procedures to be executed as well, you should include the appropriate calls to these procedures in the case load procedure.

*Included
datasets*

18.5 Case file security

When your AIMMS-based application is used by multiple end-users, all sharing the same data management tree, read and/or write protection of the individual datasets and cases may become a relevant issue. AIMMS offers such protection by allowing you to create a database of end-users (see Section 21.3), and then letting datasets and cases be owned by individuals in this end-user database. Whenever an AIMMS application is linked to an end-user database, users must authenticate themselves before being able to use the application.

*Protecting your
data*

As explained in Section 21.3, each end-user in an AIMMS end-user database must be member of a particular user group. User groups can be ordered in a hierarchical fashion. With respect to datasets and cases, AIMMS allows you to assign different access rights to

Access rights

- the owner of the dataset or case,
- members of the group associated with the dataset or case,

- members of groups that lie hierarchically above or below the user group associated with the dataset or case, and
- all other users.

By default, any dataset or case will be owned by the user ID and group of the user who created it. In addition, the access rights associated with such a dataset or case will be the default access rights of the end-user (or group of end-users). These default access rights are assigned by the local user administrator in the end-user database (see also Section 21.4).

Default access rights

When you are the owner of a dataset or case it is possible to modify previously assigned access rights to a case. You can perform this task through the **Access** tab of the **Properties** dialog box of the dataset or case in the **Data Manager**, which will only be present if an end-user database is linked to your application. In the **Access** tab, displayed in Figure 18.11, you can modify the associated

Modifying access rights

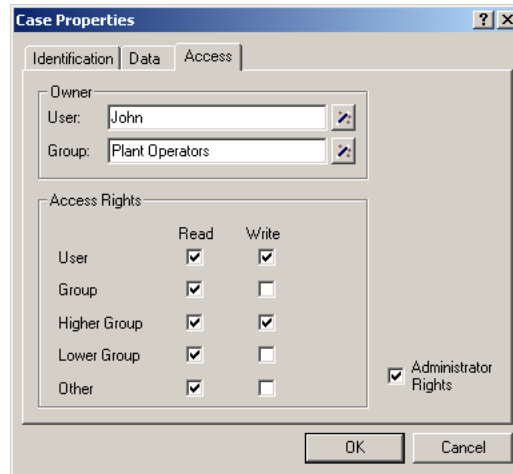


Figure 18.11: Access rights of a dataset or case

user ID and group that own the data file, as well as the access rights for each of the access categories listed above.

Normally, AIMMS will only allow you to modify the access rights of the datasets and cases that you yourself own. You can override this by checking the **Administrator Rights** check box displayed in Figure 18.11. This will pop up a password dialog box requesting the administrator password associated with the end-user database. If successful, you can modify the access rights of any dataset or case as if you were its owner.

Administrator rights

18.6 Data manager files

By default, AIMMS stores all cases and datasets associated with a particular modeling project in a single *data manager file* within the project directory. AIMMS allows you, however, to create or open another data manager file. This makes it possible that the cases and datasets of all end-users of your project be stored in a common data file.

Case and dataset storage

You can select another data manager file by means of the **File-Open-Data File** menu. Alternatively, you can create a new data manager file through the **File-New-Data File** menu. For every project, AIMMS remembers the last data manager file opened during an AIMMS session, and will reopen with the same data manager file at the beginning of a new session. Alternatively, you can indicate the data manager file with which you want to open a project as a command line argument in AIMMS. The complete list of AIMMS' command line arguments is provided in Section 20.1.

Selecting a data manager file

When your modeling application is linked to an end-user database, any newly created data manager file will also automatically be associated with that end-user database. When you try to select another data manager file, AIMMS will only allow this if the current end-user database of the application coincides with the end-user database associated with the selected data manager file.

End-user database

The **Import** and **Export** facilities in the AIMMS **Data Manager** allow you to transfer a collection of datasets and cases stored in the case tree to a new data manager file, and vice versa. The import and export facilities let you easily create a backup of your data onto a floppy, and offer you a means of sending a single scenario to an interested colleague.

Importing and exporting data files

Through the **Export** facility AIMMS can export the selected cases and datasets in the data manager to a new data manager file. If any of the selected cases contains references to datasets which you have not explicitly selected, such datasets will be exported as well. This ensures that any exported case will refer to exactly the same data, when imported by another user. The newly created data manager file is associated with the same end-user database as the currently open data manager file, and the exported cases and datasets have the same owner and access rights as before.

Exporting data files

Through the **Import** facility AIMMS allows you to import *all* cases and datasets within a given data manager file into the current case tree. If a certain import case or dataset already exists in the case tree in which it is imported, you have the choice of overwriting the existing entry or creating a new node. AIMMS will always create a new node if you do not have permission to overwrite an

Importing data files

existing node in the case tree. When AIMMS creates a new node for an existing entry, the name of the existing node is prefixed with the string 'Imported', followed by a number if there are more than one imported copies. Any existing entry that is overwritten will keep its current owner and access rights, while newly created cases and datasets will have the same owner and access rights as stored in the import file.

18.7 Accessing the case tree from within the model

When your modeling application depends on the use of multiple case types and/or data categories, performing data management in a consistent manner may be a task that is too involved for a casual end-user. In such cases, AIMMS allows you to setup a custom data management system yourself. From within the AIMMS language, you have access to the contents of the AIMMS case and dataset tree, as well as to all functionality for loading, saving and creating datasets and cases available through the standard end-user menus. This enables you to shield your end-users from choices which may be too involved for them to understand, and ensure that every created case is built up in a consistent manner.

Custom data management

After you created a customized data management system, you may want to restrict the end-user capabilities in the **Data Manager** on a task-by-task basis to prevent them from making inadvertent mistakes. You can accomplish this by modifying the appropriate options in the **Properties** dialog box of a case type or data category and in the global AIMMS **Options** dialog box. For instance, AIMMS allows you to completely hide datasets within the **Data Manager** if you want your case management scheme to depend on datasets, but do not want to bother your users. Modifying global AIMMS options is explained in full detail in Section [22.1](#).

Restricting end-user capabilities

All data categories, datasets and cases in an application are accessible in the AIMMS language through a number of predefined sets and parameters. They are:

Predefined model identifiers

- the set `AllDataCategories`, containing the names of all data categories defined in the data manager setup window,
- the set `AllCaseTypes`, containing the names of all case types defined in the data manager setup window,
- the integer set `AllDataFiles`, representing all datasets and cases available with a particular project,
- the set `AllDatasets`, a subset of `AllDataFiles`, representing the collection of all datasets available in the project,
- the set `AllCases`, a subset of `AllDataFiles`, representing the set of all cases available for the project,

- the indexed element parameter `CurrentDataset` in `AllDatasets` and defined over `AllDataCategories` containing the currently active datasets,
- the scalar element parameter `CurrentCase` in `AllCases`, and
- the scalar element parameter `CurrentDefaultCaseType` in `AllCaseTypes`.

In addition to the collection of predefined identifiers described above, AIMMS offers a complete range of data management related interface functions which you can call from within your model to perform data management tasks. These functions are described in Section 19.4.5. They allow you to perform tasks such as:

Interface functions

- obtaining additional information about the data categories, datasets and cases contained in one of the predefined sets described above, as available within the **Data Manager**,
- invoking functionality from the **Data Manager**'s end-user menus,
- invoking functionality from the end-user **Data** menu.

By combining the above, it is possible to guide your end-users through the process of selecting datasets for all data categories involved in a particular case, providing additional information as you see fit. Alternatively, if a case has a known and fixed structure, AIMMS allows you to build it up within the language without any user interaction.

Guide your end-users

18.8 The case tree versus databases

The features discussed in the previous section may have lead you to believe that you can use the AIMMS case tree as a database to store data owned and shared by multiple users. While this is true in principle, there are situations where the use of a true database is preferable over using the AIMMS case tree. This section discusses the issues which you should take into consideration before making a choice.

Case tree as database?

AIMMS cases are tailored to store the contents of one or more identifiers in your model quickly and easily. Therefore, storing and retrieving data through AIMMS case files is much faster than accessing the same data from a database server. In addition, setting up a link with a database is much more involved, as you need to specify a `READ` or `WRITE` statement for every individual table in the database. The increased speed of data retrieval from and to case files may be essential to gain end-user acceptance of your application.

Speed

The focus when storing data in a database is inherently different from storing data in an AIMMS case tree. Whereas the object of a database is to store and maintain a single version of a particular table to be shared by several applications, the AIMMS case tree is specifically set up to be able to easily maintain and switch between multiple versions of the data associated with a particular set of identifiers within a single model. This AIMMS feature allows you to easily perform a what-if analysis by running your model with different scenarios stored as separate case files. In addition, AIMMS allows you to simultaneously view identifier values from multiple case files within a single graphical object.

Different focus

AIMMS will ensure that any read or write action on a data file will not interfere with another user accessing the same data file at the same time. However, since AIMMS copies the data from a data file into memory, there is no guarantee that the data in the data file remains synchronized with changes made to the case file by other users. Whenever such synchronization is essential in your application, you are advised to use database technology.

Synchronization

Although the use of cases and datasets is the fastest and most convenient way to store and retrieve internal AIMMS data, it is not suitable for storage of data which the AIMMS application shares with other applications. When your AIMMS application needs input data produced by another application, or when the results of your model are input to other programs, you are strongly advised to store the data in a database.

Data sharing