

AIMMS 3.12

Release Notes for Win32 Build 3.12 PR - SU9

Visit our web site www.aimms.com
for regular updates

Paragon Decision Technology

January 29, 2014

Contents

Contents	2
1 System Requirements	3
1.1 Hardware and operating system requirements	3
1.2 ODBC and OLE DB database connectivity issues	4
1.3 Viewing help files and documentation	5
2 Installation Instructions	7
2.1 Installation instructions	7
2.2 Solver availability per platform	8
2.3 AIMMS 3 licensing	9
2.3.1 Personal and machine nodelocks	11
2.3.2 Installing an AIMMS license	13
2.3.3 Managing AIMMS licenses	15
2.3.4 Location of license files	17
2.4 OpenSSL license	19
3 Getting Support	22
3.1 Reporting a problem	22
3.2 Known and reported issues	23
4 Release Notes	25
What's new in AIMMS 3.12	25
AIMMS 3.12 PR - SU9, release date 2014-01-29	30
AIMMS 3.12 PR - SU8, release date 2013-10-16	31
AIMMS 3.12 PR - SU7, release date 2013-07-12	31
AIMMS 3.12 PR - SU6, release date 2013-04-03	32
AIMMS 3.12 PR - SU5, release date 2012-12-28	33
AIMMS 3.12 PR - SU4, release date 2012-12-17	33
AIMMS 3.12 PR - SU3, release date 2012-09-21	35
AIMMS 3.12 PR - SU2, release date 2012-08-27	35
AIMMS 3.12 PR - SU1, release date 2012-06-11	37
AIMMS 3.12 PR, release date 2012-03-07	39
AIMMS 3.12 FR3, release date 2011-12-22	40
AIMMS 3.12 FR2, release date 2011-08-24	43
AIMMS 3.12 FR1, release date 2011-05-25	45

Chapter 1

System Requirements

This chapter discusses the system requirements necessary to run the various components of your Win32 AIMMS 3.12 system successfully. When a particular requirement involves the installation of additional system software components, or an update thereof, the (optional) installation of such components will be part of the AIMMS installation procedure.

System requirements

1.1 Hardware and operating system requirements

The following list provides the minimum hardware requirements to run your AIMMS 3 system.

Hardware requirements

- 1.6 Ghz or higher x86 or x64 processor
- XGA display adapter and monitor
- 1 Gb RAM
- 1 Gb free disk space
- A parallel or USB port (only when the hardware dongle is used)

Note, however, that performance depends on model size and type and can vary. It can also be affected by the number of other applications that are running concurrently with AIMMS. In cases of a (regular) performance drop of either AIMMS or other applications you are advised to install sufficiently additional RAM. If you are exceeding the limits of x86, you are advised to move to x64.

The Win32 version of AIMMS 3.12 is designed to run under

Supported Windows versions

- Windows XP,
- Windows Server 2003,
- Windows Vista,
- Windows 7,
- Windows 8,
- Windows Server 2003, and
- Windows Server 2008.

The Win32 version of AIMMS 3.12 Unicode is supported under

- Windows XP,
- Windows Server 2003,
- Windows Vista,
- Windows 7,
- Windows 8,
- Windows Server 2003, and
- Windows Server 2008.

Running the Win32 version of AIMMS 3.12 versions under the 32-bit Windows versions 95/98/ME/2000/NT 4.0 is not supported.

1.2 ODBC and OLE DB database connectivity issues

Before you can start using the database connectivity features built into the Win32 version of AIMMS 3.12, the following ODBC and OLE DB components need to be installed on your computer:

Database connectivity

- the Microsoft Data Access Components (MDAC) version 2.5 or higher, and
- an ODBC driver for each database from which you wish to retrieve data from within your modeling application through ODBC.
- an OLE DB provider for each database from which you wish to retrieve data from within your modeling application through OLE DB.

This section discusses the compatibility issues between the various available ODBC and OLE DB components and AIMMS 3.

The MDAC components, as well as the Microsoft ODBC Jet drivers (which allow you to connect to, for instance, Microsoft Access databases), and OLE DB providers for the most commonly used databases, are available through the Microsoft web site. Please note that early versions of the Data Access Components (or of the ODBC Jet drivers) may exhibit problems that prevent AIMMS 3 from successfully connecting to certain databases. If you are using the ODBC connection layer, you are therefore strongly advised to install the latest version of the Microsoft Data Access Components.

Microsoft Data Access Components

The AIMMS 3 CD-ROM contains the installation executable of both Microsoft Data Access Components version 2.7 and the Jet 4.0 ODBC drivers and OLE DB providers. You can install these using the CD-browser which automatically starts up when you insert the AIMMS 3 CD-ROM into your computer. Alternatively, if you have downloaded the AIMMS 3 installation executable from the AIMMS 3 web site www.aimms.com, you can download the installation executable for the latest Microsoft Data Access Components from the Microsoft web site www.microsoft.com.

Data Access Components installation

If your model needs to retrieve data from an Oracle database, the following information may be relevant to you. The Oracle provided ODBC drivers for Oracle version 7 only support ODBC version 2.0. If you are using Oracle version 7, you should therefore use the Oracle ODBC driver that is distributed with the Microsoft Data Access Components. The Oracle provided ODBC drivers for Oracle versions 8 and higher support ODBC version 3.0, and can therefore use the ODBC 3.0 connection layer. Oracle versions 8 and higher also provides a native OLE DB provider. You can download the installation executables of the most recent version of the ODBC drivers and OLE DB providers for Oracle 8 and higher from the Oracle web site www.oracle.com.

*Oracle ODBC
drivers and OLE
DB providers*

We discourage the use of the ODBC drivers and OLE DB providers for Oracle that are distributed with the Microsoft Data Access Components, as these drivers only support a subset of the functionality provided by the native Oracle drivers. In addition, in our internal tests, they appear to have issues with certain ODBC/OLE DB functionalities that are used by AIMMS.

*Prefer native
Oracle over
MDAC drivers*

If you are using OLE DB for database connectivity, the following issues apply:

OLE DB issues

- Access stored procedures cannot be called (yet) with the Jet 4.0 OLE DB provider. There is no work around either. Try to replace stored procedures by an AIMMS DATABASE PROCEDURE. Access queries, which are not marked as 'stored procedure' by Access, are perfectly well possible through the AIMMS DATABASE PROCEDURE.
- Oracle's own OLE DB provider for Oracle generally performs better than Microsoft's OLE DB provider for Oracle.
- Using long column types (such as text or ntext) with SQL Server, may reduce your application's performance dramatically. Therefore (and because those column types hardly make sense in an AIMMS context), try not to use them. If you feel you really need them, it may be a wise choice to specify an ODBC data source for the database tables that contain these columns.
- Using the OLE DB provider for ODBC may result in errors. Try to use the database manufacturers' own OLE DB provider as much as possible to get better performance and more stable behavior.

1.3 Viewing help files and documentation

The AIMMS 3 User's Guide, the AIMMS 3 Language Reference, the AIMMS 3 Function Reference, and the AIMMS book on Optimization Modeling are available online as Adobe Portable Document Format (PDF) files. In order to view or print PDF documents, Adobe Acrobat Reader version 4.0 or higher needs to be installed on your computer.

*AIMMS
documentation*

The AIMMS 3 CD-ROM contains the Acrobat Reader 6.0 setup executable, and you can install it using the CD-browser which automatically starts up when you insert the AIMMS 3 CD-ROM into your computer. Alternatively, if you have downloaded the AIMMS 3 installation executable from the AIMMS 3 web site www.aimms.com, you can download the Acrobat Reader 6.0 installation executable from the Adobe web site www.adobe.com.

*Acrobat Reader
installation*

Chapter 2

Installation Instructions

This chapter discusses all relevant issues related to the setup of AIMMS 3 on your computer. To install AIMMS 3 on your computer, you can

Installation instructions

- run the AIMMS 3 setup program from the AIMMS 3 CD-ROM, or
- download the setup program for the latest release of AIMMS 3 from the download area of our web site, and run it from your hard disk.

The latter is the standard procedure for obtaining and installing AIMMS 3 Software Updates. Note that in order to run AIMMS properly, AIMMS requires some specific 3rd- party library versions to be installed (e.g. Microsoft Visual Studio runtime library, Microsoft .NET Framework). In the rest of this text, these libraries are called the prerequisites for running AIMMS.

2.1 Installation instructions

When you install AIMMS 3 from the AIMMS 3 CD-ROM, a CD-browser will be started automatically when you insert the AIMMS 3 CD-ROM into your computer. Using this CD-browser, you can

The AIMMS 3 CD-browser

- start the AIMMS 3 or AIMMS 3 Unicode setup program (these setup programs take care of installing the prerequisites as well),
- install any other third-party software or system components available on the AIMMS 3 CD-ROM which are required to use certain components of AIMMS 3 (as described in Chapter 1), and
- start the AIMMS 3 License Server setup program (for network licenses only).

The download area of our web site www.aimms.com provides the AIMMS 3 setup program of the latest AIMMS 3 releases. If you have selected the AIMMS setup program that you want to install, the corresponding prerequisites are selected as well. Note that pressing 'Download.msi' does not include the prerequisites. Pressing the 'Download' button does result in all selected files (including prerequisites) being downloaded.

Obtaining AIMMS 3 from the web

The AIMMS 3 setup program will guide you through the various steps that are necessary to successfully install AIMMS 3 on your computer. The AIMMS 3 setup program requires some prerequisites to be installed. When you install AIMMS 3 from an AIMMS 3 CD-ROM or from a downloaded package file, the AIMMS 3 setup program automatically takes care of installing the prerequisites.

The AIMMS 3 setup program

Certain parts of the setup of AIMMS 3 may require administrative privileges. More specifically, you will need administrative privileges for

Administrative privileges required

- the installation of the device drivers required to access the AIMMS hardware dongle (Windows only),
- the installation of, or updates to, any system components which access areas of the Windows registry in which you have no write access, or which write to the Windows system directory, if write access to that directory has been restricted, and
- the creation of writable directories in the common application directory on your computer where AIMMS will store its license configuration and any nodelock files associated with your AIMMS system, if write access to the common application area of your computer has been restricted.

The AIMMS 3 setup program requires that you provide the specific selection of AIMMS 3 components you wish to install (the setup will select all common AIMMS features by default (Typical), which is recommended)

Component selection

2.2 Solver availability per platform

In Table 2.1 you can find an overview of the solvers that are installed during a Typical and a Complete AIMMS installation on the several platforms on which AIMMS is supported. Note that it depends on your license whether you can use these solvers.

Solvers in installation

Please note that the following solvers require a computer with a CPU that supports the SSE2 instruction set: GUROBI, MOSEK 6 or higher and CPLEX 12.2 or higher.

SSE2

CBC and IPOPT are open source solvers made available in the AIMMS distributable as solver DLL and through COIN-OR (www.coin-or.org) as solver DLL and in source code format. Please check the 'OS Solvers' section in the License Agreement before using these open source solvers with AIMMS.

Open Source Solvers

Solver	Version	Win 32		Win 64 *	Linux 32 *	Linux 64 *
		Typical	Complete			
AOA		✓	✓	✓	✓	✓
BARON	7.5.3	-	✓	-	-	-
BARON	10.3	✓	✓	✓	-	-
CBC	2.6	-	✓	✓	✓	✓
CBC	2.7.5	✓	✓	✓	✓	✓
CONOPT	2.071 C	-	✓	-	-	-
	3.11 B	-	✓	-	-	-
	3.14 A	-	✓	-	✓	-
	3.14 G	✓	✓	-	-	-
	3.14 M	-	-	✓	-	-
	3.14 Q	-	-	-	-	✓
	3.14 V	✓	✓	✓	✓	✓
CPLEX	10.1	-	✓	✓	✓	✓
	11.2	-	✓	✓	✓	✓
	12.1	-	✓	✓	✓	✓
	12.2	-	✓	✓	✓	✓
	12.3	-	✓	✓	✓	✓
	12.4	✓	✓	✓	✓	✓
	12.5	✓	✓	✓	✓	✓
GUROBI	3.0	-	✓	✓	✓	✓
	4.0	-	✓	✓	✓	✓
	4.5	-	✓	✓	✓	✓
	4.6	✓	✓	✓	✓	✓
	5.0	✓	✓	✓	✓	✓
	5.1	✓	✓	✓	-	✓
IPOPT	3.9	-	✓	✓	✓	✓
IPOPT	3.10.1	✓	✓	✓	✓	✓
KNITRO	5.2	-	✓	✓	✓	✓
	6.0	-	✓	✓	✓	✓
	7.0	-	✓	✓	✓	✓
	8.0	✓	✓	✓	✓	✓
LGO	1.0	✓	✓	-	-	-
MINOS		✓	✓	✓	✓	✓
MOSEK	6	✓	✓	✓	✓	✓
PATH	4.4a	-	✓	-	-	-
	4.6	✓	✓	-	-	-
SNOPT	6.1	✓	✓	✓	✓	✓
	7.2	-	✓	✓	✓	✓
XA	13	-	✓	-	-	-
	14	-	✓	✓	✓	✓
	15	✓	✓	✓	✓	✓
XPRESS	19	-	✓	-	-	-

* Typical and Complete installations are the same for Win 64 and Linux 32/64.

Table 2.1: Solvers included in the several types of AIMMS installations

2.3 AIMMS 3 licensing

AIMMS offers the following two types of licenses:

*AIMMS 3
licensing*

- single-user licenses, and
- network licenses.

Each of these two types of licenses are protected in a different manner.

Single-user licenses can be used by a single user on a single computer. To enforce the single-user character, AIMMS 3.12 requires that single-user licenses be protected by either

*Single-user
license
protection*

- a hardware dongle, which, depending on the type of dongle, must be connected to a USB or parallel port of your computer, or
- a nodelock file, which must be activated to match the hardware characteristic of your computer.

When ordering the Windows version of the AIMMS software you can indicate whether you want your AIMMS system to be protected by a dongle or by a nodelock. Which choice to make is very dependent on your situation and the intended use of the AIMMS software. The Linux platform does not support dongles.

You can choose

Dongles offer you the most flexibility when you want to use AIMMS on multiple computers, and do not want the hassle of having to deactivate and activate a nodelock on these computers. On the other hand, dongles occasionally break, you can forget to take the dongle with you, they can be stolen, and, because of their size, get lost quite easily, especially if you are moving them around a lot.

*Pros and cons of
dongles*

Nodelock files are stored on the harddisk of your computer, and are, therefore, much less vulnerable to loss. Only if your computer is stolen, or in case of a harddisk crash, you must contact Paragon before being able to activate your nodelock on a replacement computer. On the other hand, if you are frequently working on multiple computers, you have to remember to deactivate the nodelock on the old computer, prior to being able to activate it on the new one, every time. In addition, you need access to the internet to activate or deactivate a nodelock.

*Pros and cons of
nodelocks*

If you decide to request a dongle for license protection, a physical shipment of the dongle to your site is required before you can start using AIMMS. If you request nodelock protection, we will send you the AIMMS license number and activation code by e-mail, after which you can start using AIMMS directly.

*Physical
shipments*

If you request your license to be protected by a dongle, an AIMMS dongle is sent to you along with your AIMMS 3 CD-ROM. If you upgrade from AIMMS 2.20, you can continue to use your existing green Activator dongles used by AIMMS 2.20, *but only with the Win32 version of AIMMS*. The green Activator as well as the grey Sentinel dongle must be connected to the parallel port of

AIMMS dongles

your computer. The purple Sentinel dongle must be connected to a USB port of your computer. The AIMMS 3.12 setup program will only install the required device drivers for accessing the grey and purple Sentinel dongles. If you still use the green Activator dongles supplied with AIMMS 2.20, you can obtain the required drivers separately from our FTP site <ftp.aimms.com>.

If you have ordered an AIMMS 3 network license, no license protection needs to be installed locally on your computer. Instead, you need the host name and port number of the server running the AIMMS 3 network license server. For more information about installing the network license server itself, please refer to the documentation of the AIMMS 3 network license server.

Network licenses

2.3.1 Personal and machine nodelocks

AIMMS offers two types of nodelocks:

- personal nodelocks, and
- machine nodelocks.

Two types of nodelocks

If you choose for nodelock protection you are free to choose between a personal or a machine type of nodelock. In this section you will find the characteristics of both types of nodelocks. If you are unsure which type of nodelock to choose, we recommend that you start with a personal nodelock, as you can change a personal nodelock into a machine nodelock at any time, but not the other way around.

Personal nodelocks are intended for use by a single AIMMS user, who still wishes to have the freedom to use AIMMS on multiple computers, for instance if you want to easily switch between your desktop computer at work, a notebook computer and your home computer. Personal nodelocks have the following characteristics:

Personal nodelock

- Personal nodelocks can be transferred to another computer 3 times per 24 hours. This allows you to take your AIMMS license home in the evening and back to work the next morning without any problems.
- Personal nodelocks have a limited lifetime of 60 days, and should be renewed within that period to extend the lifetime to its full 60-day period. If the nodelock is not renewed within its 60-day lifetime, this does not invalidate your AIMMS license in any way—you only have to renew your nodelock prior to being able to use your AIMMS system again. Note that the renewal process does not require any manual intervention, as AIMMS will try to automatically connect to our internet license database to renew your nodelock once every day you are using AIMMS.
- Both activation and nodelock renewal of personal nodelocks require a working connection to the internet. As a consequence, in the absence of an internet connection you can continue to work uninterrupted for

a period of 60 days, before an internet connection is required to renew your nodelock.

- With every activation or nodelock renewal AIMMS will also update your license files if new license files are available (e.g. if your system is in maintenance), and will inform you of any messages that are available for you in our database.
- Because of their volatile nature, PDT will replace a personal nodelock without any questions asked in case of loss of or damage to your computer.
- You can switch your personal nodelock to a machine nodelock at any time.

Machine nodelocks are intended for permanent use on a single computer. They are recommended for server applications, and can also be used for personal use if you are sure you will be using AIMMS on a single computer, or do not have internet access. Machine nodelocks have the following characteristics:

*Machine
nodelock*

- Machine nodelocks can be transferred to a replacement computer 3 times per 365 days.
- Machine nodelocks have an unlimited lifetime (unless deactivated).
- Machine nodelocks can be either activated online if your computer is connected to the internet, or offline through the license activation area on the AIMMS website.
- License files will only be retrieved when the machine nodelock is activated, or by explicit request.
- In case of failure, PDT will, in principle, only replace machine nodelocks on the same computer.
- Once you have chosen for a machine nodelock, it is not possible to switch back to a personal nodelock.

Although a personal nodelock and the software version check on the start page make a regular connection to the internet (the personal nodelock connects to a license database for nodelock renewal and the version check connects to a version database), we do respect your privacy and will not register patterns in your personal usage of the AIMMS software in any way. During activation no personal information will be transferred, only your computer name and some of its hardware characteristics. During deactivation we register the date and time of deactivation to enforce the transfer limit.

Privacy

The connection to our internet license database is implemented as a web service. Thus, if you are able to browse the web, you should also have no trouble activating an AIMMS nodelock. If your computer connects to the internet through a proxy server, AIMMS by default tries to detect and use the proxy settings also used by Microsoft Internet Explorer.

*Internet
connection and
proxy settings*

It should be noted that the use of auto-configuration scripts in determining the proxy server will fail if these use any other scripting language than Javascript. This is due to the libraries underlying the SOAP library used by AIMMS to connect to our license server. If you are in this situation, you should manually configure the proxy settings, as described below.

Automatic configuration scripts

If AIMMS does not succeed in automatically detecting the proxy settings that apply in your network environment, AIMMS also allows you to manually set the proxy settings during the activation process. If the online activation process does not succeed directly, AIMMS gives you the option to either continue with an offline activation process, or to manually supply the proxy settings that apply to your network environment through the dialog box illustrated in Figure 2.1. In this dialog box you can choose between

Manual proxy setting

- the *Current User* settings also used by Microsoft Internet Explorer (default),
- the *Local Machine* settings which are stored in the registry, if these are available on your machine, or
- *Custom proxy* settings that you have received from your IT department.

In the latter case, you can also (optionally) provide a user name and password to authenticate with the proxy server. In most cases, however, setting these will not be necessary, and Windows authentication will be sufficient.

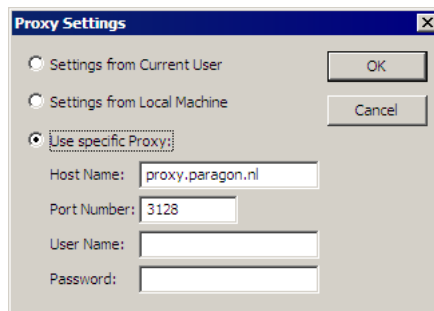
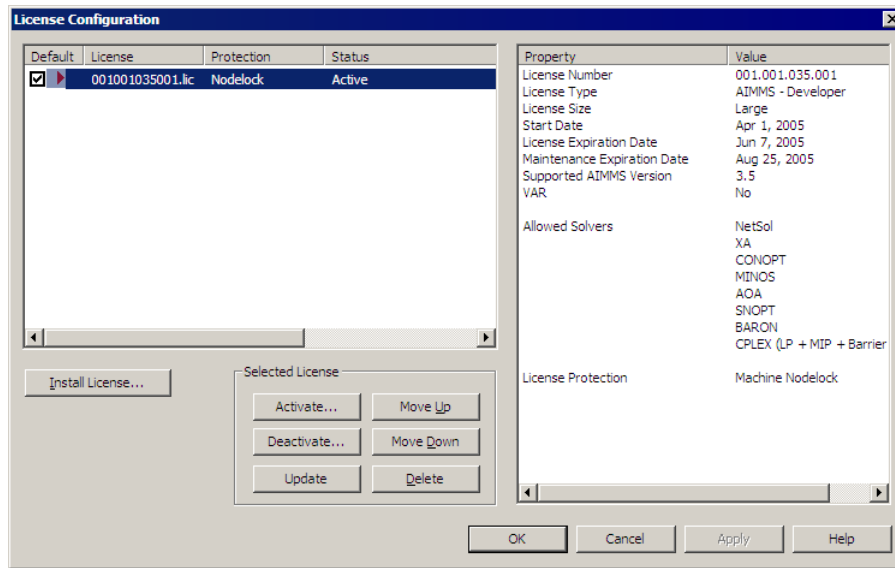


Figure 2.1: The AIMMS **Proxy Configuration** dialog box

2.3.2 Installing an AIMMS license

When you start up AIMMS 3.12 for the first time after installation, AIMMS will open the **License Configuration** dialog box illustrated in Figure 2.2. Through this dialog box you can install new AIMMS licenses and manage all AIMMS licenses that already have been installed on your computer.

Managing your AIMMS licenses

Figure 2.2: The **License Configuration** dialog box

To install a new license, press the **Install License ...** button in the **License Configuration** dialog box. This will start a wizard, that will guide you through the license installation procedure step by step. The wizard can help you to install

Installing a new license

- existing AIMMS 3.11 licenses,
- nodelocked licenses,
- dongled licenses,
- network licenses,
- evaluation licenses, and
- student licenses.

To successfully complete the installation of licenses of each type, you should make sure to have the following information available.

To install a single-user AIMMS license that is protected by a nodelock, you need the following information:

Single-user nodelocked licenses

- your AIMMS license number, and
- the associated activation code that you received from Paragon.

You have the choice to request a personal nodelock or a machine nodelock. A personal nodelock must be requested online, a machine nodelock can be requested online or offline. Refer to Section 2.3.1 for a more detailed introduction to personal and machine nodelocks.

To install a single-user AIMMS license that is protected by a dongle, you need the following items:

*Single-user
dongled licenses*

- an AIMMS dongle attached to a USB or parallel port of your PC, and
- the associated set of license files that you received from Paragon.

To install an AIMMS network license, you need the following information from your system administrator:

*Network
licenses*

- the name of the AIMMS network license server,
- the port number of the AIMMS network license server, and
- the name of the license profile to which you want to connect (optional).

To install an AIMMS evaluation license you need the following information

*Evaluation
licenses*

- your AIMMS evaluation license number, and
- the associated activation code that you received from Paragon when requesting an evaluation license.

You must have a working connection to the internet (not necessarily on the machine on which you installed AIMMS) to activate an evaluation license. Evaluation licenses expire 30 days after activation. Note that each evaluation license can be activated only once, and that you can only activate a single evaluation license per AIMMS release on a specific computer, regardless of the number of evaluation licenses you have requested on our web site.

To install an AIMMS student license you need the following information:

Student licenses

- your AIMMS student license number, and
- the associated activation code that you received from the university that purchased the AIMMS Educational Package.

You must have a working connection to the internet to activate a student license. Student licenses expire one month after the end of the current academic year. Student licenses can be activated multiple times.

2.3.3 Managing AIMMS licenses

AIMMS allows you to have multiple AIMMS licenses installed on your computer. You may have multiple licenses installed, for instance, for the following reasons:

*Managing
multiple AIMMS
licenses*

- you have requested a trial license for a new AIMMS version which you want to run next to your existing license,
- you have temporarily borrowed or hired an AIMMS license with more capabilities than your regular license,

- your system administrator has created multiple network license profiles, each of which you may want to use to run AIMMS.

In this section we will describe how you can instruct AIMMS which license to use.

In the **License Configuration** dialog box displayed in Figure 2.2, all AIMMS licenses installed on your machine will be displayed in the left pane of the dialog box. The license details of each license are displayed in the right pane of the dialog box. During startup AIMMS will consider all licenses in the left pane of the **License Configuration** dialog box which have the **Default** column checked, and will use the first valid license it finds starting from top to bottom. Using the **Move Up** and **Move Down** buttons you can change the order in which AIMMS will search the list.

Default licenses

Both personal and machine nodelocks can be transferred to other computers. Personal nodelocks can be transferred upto three times a day, allowing you to take your license with you wherever you want. Machine nodelocks can be transferred three times per year, to a computer replacing the computer on which the nodelock is currently installed. To transfer a nodelocked license, you must

Transferring licenses

- deactivate the nodelock on the currently active computer, and
- activate it on the computer to which you want to transfer the license.

You can deactivate an active nodelock using the **Deactivate** button in the **License Configuration** dialog box. Deactivation will only succeed if there is no conflict with the transfer limit for the given nodelock type. This makes sure that there will never be a problem activating a deactivated license. After successful deactivation the license will not be removed from the list but be marked as inactive. If the license is not active on any computer, you can reactivate the license through the **Activate** button.

In case you want to activate a nodelock on a computer, but have forgotten to deactivate the nodelock on a computer to which you currently have no access, AIMMS allows you, as a courtesy, to request an emergency nodelock 3 times per 365 days. Emergency nodelocks have a lifetime of 7 days, and during this time you can arrange for someone to deactivate the license on the computer containing the active nodelock. During the activation sequence, AIMMS will automatically ask whether you would like to receive an emergency nodelock when it discovers that the license is active on another computer.

Emergency nodelocks

2.3.4 Location of license files

AIMMS keeps its license and configuration files in the folder

Paragon Decision Technology

*Location of
license files*

of the common application area of your computer. On Windows XP, and Windows Server 2003, the common application area is located, by default, at

C:\Documents and Settings\All Users\Application Data

On Windows Vista and Windows 7, this folder is located under C:\ProgramData. The Paragon Decision Technology folder contains three subfolders

- Config, containing the license and solver configuration files,
- Licenses, containing all license files,
- Nodelocks, containing all nodelock files installed on your computer, and
- AnyUser, containing the license configuration files for all users on your computer (see below).

The AIMMS installation makes sure that these subfolders are writable for everyone, allowing you to install and uninstall licenses on your computer.

To prevent tampering with nodelocked licenses, AIMMS keeps track of the location of the nodelock files associated with a license. You should, therefore, not manually move or copy the AIMMS nodelock files as this may invalidate your nodelock.

*Do not move
nodelock files*

You can specify whether the license and solver configuration that AIMMS uses is the same for any user of the machine, or different for each individual user. To prevent problems when running AIMMS as part of a computer service, AIMMS 3.12 will by default use the same configuration for any user. To modify this behavior, you should edit the file UserDistinction.cfg in the common Config folder. In the file UserDistinction.cfg.default, straightforward directions are given on how to edit it. If no UserDistinction.cfg file exists, AIMMS will use the UserDistinction.cfg.default file instead. You can use this file as a base for setting up your own configuration.

*User specific
configuration*

In the scenario where all users of the same pc use the same license and solver configuration, the configurations that are modified by a user are stored in the AnyUser folder of the Common folder.

Any User

In the scenario where each specific user of the pc has its own license and solver configuration, the configurations that are modified by a user are stored in the local application data folder. On Windows XP and Windows Server 2003, this local folder is usually located at:

Current User

```
C:\Document and Settings\\Application Data\Paragon
Decision Technology
```

On Windows Vista and Windows 7 it is:

```
C:\Users\\AppData\Roaming\Paragon Decision Technology
```

When AIMMS needs to read the current configuration, it will first look in the (any)user folder as specified by the aforementioned UserDistinction.cfg file; if it cannot be found there, it will try to read the configuration from the common application data folder. When saving a modified configuration, AIMMS always writes to the (any)user folder.

*Accessing
configuration
files*

Inside the Config folder of the (any)user folder, each major AIMMS version (3.6, 3.7, 3.8, etc.) will create its own specific subfolder when it needs to write a configuration file. During an attempt to read, AIMMS will first look for the specific file in the folder that matches its own major version number, and otherwise it will subsequently search through the folders of previous versions. In other words, when you upgrade to a new AIMMS version, initially your configuration will be the same as the one you were using for the previous AIMMS version, but if you change something in the configuration, this will only affect the configuration of the AIMMS version you are working with.

*Version
dependent
configurations*

If you are using an AIMMS Network License, then your local machine does not need to have any license file installed. The only required file is the license configuration file, that contains the info of where the License Server is located on your LAN. When logging on to the License Server, the licensing info is sent directly from the server to the running AIMMS session, except for some secondary license related files:

*Network License
Client Files*

- the .SLV file (containing the default solver configuration),
- the .VID file (containing the VAR identification info), and
- the .CPX file (the CPLEX license file).

These secondary license files are temporarily copied to the folder NetworkCache which is located in the (any)user folder as described above.

You can specify a project dependent solver configuration by placing a solver configuration file with the name 'solvers.slv' in the project directory. AIMMS will first look for this file and if it cannot find it will look for other solver configuration files. See the AIMMS Help for more information.

*Project
dependent
configuration*

2.4 OpenSSL license

The separate *RPC* installation file, which you can download in addition to the main AIMMS software, contains the OpenSSL and SSLeay libraries to provide the HTTPS transport for AIMMS web services. The following licenses apply to OpenSSL and SSLeay.

*OpenSSL
included - in
separate RPC
installation only*

Copyright (c) 1998-2006 The OpenSSL Project. All rights reserved.

OpenSSL license

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OPENSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED

AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved. *SSLeay license*

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)" The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WAR-

RANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

Chapter 3

Getting Support

This chapter discusses the procedures that apply when you encounter problems in AIMMS 3. Please refer to the list of known and reported issues at the end of this chapter to verify whether a problem which you wish to report has already been reported before.

Getting support

3.1 Reporting a problem

When you encounter a problem in AIMMS 3, Paragon Decision Technology (PDT) will investigate the problem, and if applicable provide a fix for it. PDT always provides this service if your AIMMS version is the latest functional AIMMS release. In all other cases, you are only eligible to obtain a fixed release if your AIMMS license is in maintenance. Any problem fix will always be performed on the latest functional AIMMS release only, and may require that you upgrade your AIMMS system to that version.

Problem fixing

Whenever you encounter a problem, PDT needs the following information to process the problem.

Reporting a problem

- **Computer information**
 - Brand and model
 - Operating system and version number (including any installed Service Packs)
 - CPU type and speed
 - Amount of installed RAM
 - MDAC and ODBC/OLE DB driver versions if the problem is database-related
- **AIMMS information**
 - License number (**Tools-License-License Configuration** menu)
 - AIMMS build number (**Help-About AIMMS** menu)
- **Problem details**
 - A detailed description of the problem
 - The type of the problem
 - * Crash
 - * Incorrect functionality
 - * Cosmetic

- * Feature request
- The severity of the problem
 - * Crash
 - * No work around available
 - * Work around available
 - * Cosmetic
- **Reproducibility** (if applicable)
 - A description of the steps required to reproduce the problem
 - A copy of your project files along with any other files used by your project, if such is necessary to reproduce the problem

You should e-mail your problem report containing the above information to Support@aimms.com. Upon receipt, PDT will investigate your report and notify you of its status, as well as the actions that will be undertaken to fix the problem. You will be notified when an AIMMS 3 version will be released, in which the problem is fixed.

*Problem
processing*

Support requests other than bug reports will, in principle, only be dealt with at our regular consulting fee. This is especially true for when you request us to provide extensive modeling support. If you are new to AIMMS 3, and need some quick pointers to help you tackle a particular modeling problem, we may decide to honor such requests at our discretion. When you send support requests to our e-mail account Support@aimms.com, please always include your AIMMS license number.

*Modeling
support*

Reports collected via the AIMMS Error Report Service are actively used by Paragon Decision Technology to continuously improve the quality and reliability of AIMMS. This service is automatically called when a severe/fatal error appears. At [Why Report Errors](#) you can read why you should report these errors. At [Data Collection Policy](#) you can read what information is collected by the AIMMS Error Report Service and what we do with this information.

*Automatic Error
Reporting*

3.2 Known and reported issues

The following issues are known in the present AIMMS 3 release.

Known issues

- Due to a change in the AIMMS 3.8.6 RPC layer, just after the installation of the AIMMS 3.8.6 RPC layer, you might encounter an error message about the AIMMS handler service. Please see this [Knowledge Base article](#).

RPC

- Deploying a web service with attachments in Windows Vista or Windows 7, with a manually started AIMMS agent project, may result in temporary attachment files not being removed from the system's temporary directory. This happens because the AIMMS HTTP/RPC Handler Service runs under the SYSTEM account, while a manually started AIMMS agent project usually does not. If the non-SYSTEM account under which the AIMMS agent project runs does not have deletion rights on the SYSTEM temporary directory (which usually is the case), the temporary attachment files cannot be deleted. There are two ways to work around this problem:
 - 1) By using an automatically started AIMMS agent (which, in a deployment scenario, is the preferred way of starting up AIMMS agents).
 - 2) By stopping the AIMMS HTTP/RPC Handler Service and instead starting the `AimmsServiceHandler.exe` executable. This file is located in the AIMMS `Common\Bin`-folder.

Windows

Vista/Windows 7

Chapter 4

Release Notes

This chapter contains a description of the new features developed for AIMMS 3.12. In addition, it contains the release notes of all AIMMS 3.12 releases. For each release, the following items are listed:

Release notes

- the build number of the release,
- the release date, and
- a description of the changes and the issues resolved in this release.

By default, the installation procedure will install the single-byte AIMMS executable. This version is sufficient for most application developers. However, if you need to distribute a localized version of an AIMMS application to, for instance, Asian or Russian end-users, use of the common single-byte AIMMS version may not be sufficient anymore, as many languages in these regions cannot be represented by means of single-byte characters. To support localization to such languages, a Unicode version of AIMMS is available, in which all strings are represented internally through double-byte characters. More details on the AIMMS Unicode version can be found in Section 23.2 of the User's Guide.

*AIMMS 3.12
versions*

An overview of all known and reported issues which are still outstanding in the current release is given in Section 3.2.

What's new in AIMMS 3.12

This documentation reflects the state of the third AIMMS 3.12 Feature Release. Compared to the AIMMS 3.11 Production Release, the following major new and extended functionalities have been added to the system:

For AIMMS 3.12 FR3, the new features/extended functionality are:

- Towards a new data management scheme
- Monitoring data changes
- Uncertainty Constraints
- Solver Updates

*What's new in
AIMMS 3.12
FR3?*

The current data management facilities in AIMMS are built upon the concept of a single data management file, containing all the cases that an AIMMS project can access. These data management files can become huge over time, and, hence, may be expensive to share with other users. Also, simultaneous access to the same data management file is not fully supported. Therefore, in practice, the only feasible approach to share case data with other users, is by using the import and export facilities of data management files, which leads to a cumbersome work flow.

Towards a new data management scheme

With the upcoming release of the AIMMS PRO Platform, we recognize that the ability to collaborate and share case data becomes much more important. For this reason, we started work on an alternative data management scheme, in which cases are stored individually. While the default storage is on the (local or network) file system, the new data management scheme allows for the implementation of alternative storage backends (such as the integrated storage facilities implemented by the AIMMS PRO platform). The case load and save dialog boxes of the new data management scheme provide transparent access to any of the available storage backends.

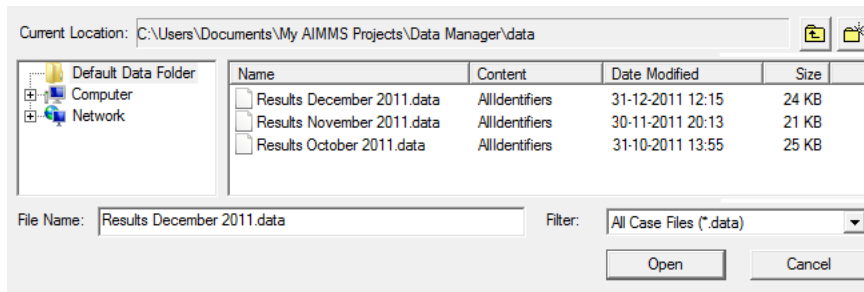


Figure 4.1: Data Management using disk files and folders

We have started the work on a new case load and save functionality. With AIMMS 3.12 FR3, we are making the basic functionality available; an almost feature complete replacement of the current data management; only the support of compound cases is not yet available. For this reason, the current data management scheme is still the default. You can use the new data management scheme by changing the option Data Management Style. The corresponding new functions can be found in the AIMMS Function Reference, in the section Data Management Functions (using disk files and folders).

Through a new set of functions you can create data change monitors, that allow you to detect whether the data in a subset of identifiers of your model, was changed. Although these functions are used by the new data management scheme to detect cases with unsaved data, you can use them independently in your model as well, for instance to detect whether identifiers have been

Monitoring data changes

changed since they were read from a database, and hence have to be written back. You can find the new functions in the AIMMS Function Reference in the section Data Change Monitor Functions.

The Robust Optimization add-on in AIMMS has been extended regarding uncertainty constraints and the convex hull region of uncertain parameters. It is now possible to specify uncertainty constraints which are applied only to specific constraints in the model. This is controlled by the new Constraints attribute that has been added for uncertainty constraints. This allows for more flexibility in specifying the level of conservatism in the robust model.

Uncertainty Constraints

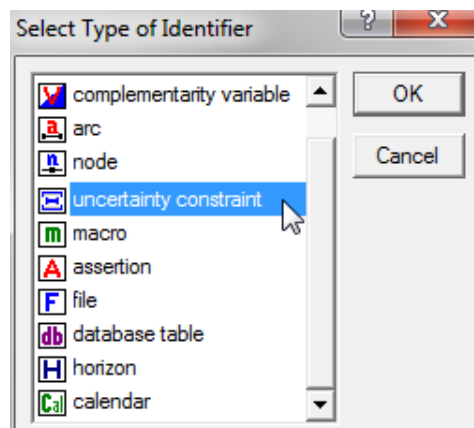


Figure 4.2: Adding an uncertainty constraint

A new `ConvexHullEx` operator has been introduced, which offers the user more flexibility in specifying scenario-dependent uncertain parameters. The user can now specify self-declared uncertain parameters to be used as the weights in convex scenario combinations. These weights can model dependencies between scenarios and can be used in the `Dependency` attribute of adjustable variables. More information about this new functionality can be found in the AIMMS Language Reference, chapter Robust Optimization.

ConvexHullEx

Three new solver versions have been added to AIMMS 3.12 FR3:

Solver Updates

- **GUROBI 4.6,**
- **KNITRO 8.0,** and
- **BARON 9.3**

The **BARON 9.3** version is also available for Windows 64-bit. In these new solver versions, the performance has been improved. Especially, the performance of the MINLP algorithm of KNITRO has been improved; it has a speed up of a factor 30 for some convex MINLP problems.

For AIMMS 3.12 FR2, the new features/extended functionality are:

- Pivot Table: Indentation Mode in row header
- Menu item: Close All Other Windows
- Function: IdentifierGetUsedInformation

What's new in AIMMS 3.12 FR2?

The layout of the row header in the Pivot Table has been extended with a new 'Indentation Mode'. This mode is especially useful if you want your row area to be more compact, such that you have more space available in the grid area. In the pictures below, the difference between the new Indentation Mode and the default Column Mode is illustrated.

Pivot Table: Indentation Mode in row header

			Month
Region	Cust...	Size	January, 2003
[-] North-America (non USA)			12.63
	[-] Customer-01		
		Medium	
		Large	
	[-] Customer-20		
		Small	12.63
[-] Netherlands			66.86
	[-] Customer-06		
		Small	28.91
		Large	8.56
	[-] Customer-09		
		Small	17.64
		Large	11.75
[-] Europe (non-NL)			66.91
	[-] Customer-03		
		Large	
	[-] Customer-17		
		Medium	50.88
		Large	
	[-] Customer-18		
		Large	17.64
	[-] Customer-19		
		Small	5.49
		Large	12.90

Figure 4.3: Indentation Mode with aggregator values

			Month
Region	Customer	Size	January, 2003
North-America (non USA)	[-] Customer-01	[-] Medium	
		Large	
	Customer-20	Small	12.63
Netherlands	[-] Customer-06	[-] Small	28.91
		Large	8.56
	Customer-09	[-] Small	17.64
		Large	11.75
Europe (non-NL)	[-] Customer-03	Large	
	Customer-17	[-] Medium	50.88
		Large	
	Customer-18	Large	17.64
	Customer-19	[-] Small	5.49
		Large	12.90

Figure 4.4: Column Mode

In 'Indentation Mode', the extra lines can (if specified as such) be used to display aggregator values. See the help and the Pivot Table example project for more details on this feature.

A new menu item has been introduced: **Window - Close All Other Windows**. With this command you can close all the currently opened (non-docking) windows, except the one that you are currently working on.

Close All Other Windows

The function `IdentifierGetUsedInformation` has been introduced, to allow the AIMMS developers to check whether an identifier is in use in any of the pages, menus or data categories/case types. By combining it with a for-loop you can use this function to quickly check for a set of identifiers whether each of them is still in use or not (the function arguments will return 1 or 0). After using this new function, you can use the already existing drag-and-find feature in the IDE to find out in which *specific* page, menu or data category the identifier is used.

Identifier-GetUsed-Information

For AIMMS 3.12 FR1, the new features/extended functionality are:

- Gantt chart extensions
- OpenOffice Calc support
- More page objects with tooltip support

What's new in AIMMS 3.12 FR1?

The Gantt chart has been extended with several new options, which allow you to create better looking charts. The most important one is that the Gantt chart is now capable of displaying partly overlapping bars in the same row, with an adjustable vertical offset, such that each individual overlapped bar is still (partly) visible. The other new options allow you to influence the height of the bars and/or the spacing between each row, which may result in a less 'crowded' looking chart. In Figure 4.5, most of the new options are illustrated.

Gantt chart extensions

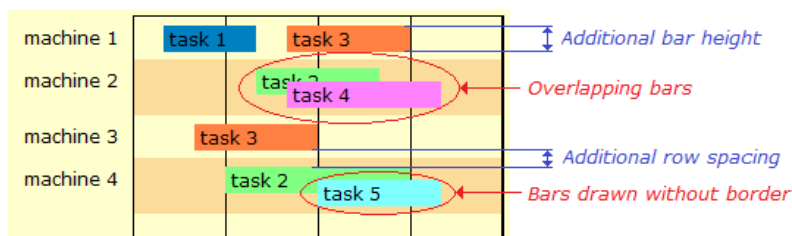


Figure 4.5: Gantt chart extensions

In addition to the existing Excel functions, AIMMS has now been extended to allow communication with OpenOffice Calc spreadsheets in the same manner. These new functions make it possible to transfer data between AIMMS and OpenOffice Calc, to run Calc macro's on your spreadsheets and to print your Calc sheets, all from within the AIMMS language. Migrating from Excel to OpenOffice Calc (and vice versa) requires almost no change to your existing AIMMS code. In general, only providing the new name of your workbook will be enough.

OpenOffice Calc support



The set of page object types that allows for user-defined tooltips have been extended with the following page object types:

More page objects with tooltip support

- Gantt chart
- Network object
- Table
- Composite Table
- Bar chart
- Curve

AIMMS 3.12 PR - SU9, release date 2014-01-29

In this AIMMS 3.12 Software Update, the following issues have been fixed or improved.

*3.12 PR - SU9
2014-01-29*

- Set definitions were evaluated during a CleanDependents statement, instead of before it, resulting in inconsistent data.
- Using the Microsoft Text ODBC driver for reading .csv files, raised an error with newer versions of this driver.
- Consider a condition such as $EP(i) < \epsilon$. AIMMS will translate such a condition into $REL_EP(i, \epsilon)$ to improve the performance. After modifying the data of EP, AIMMS might not detect that the corresponding constraint is out of date when regenerating a mathematical program.
- When multiple sources and multiple layers were defined in a **Network** object with a GIS background and not all layers were visible, the **Network** object showed some unnecessary flickering.
- Deleting a unit parameter and subsequently creating a unit parameter with the same name may lead to superfluous error messages.

Issues...

AIMMS 3.12 PR - SU8, release date 2013-10-16

In this AIMMS 3.12 Software Update, the following issues have been fixed or improved.

*3.12 PR - SU8
2013-10-16*

- The **Gantt Chart** object's navigation keys did not work when the chart was scrolled down.
- Using the argument wizard to add an argument that used an index in its domain that was already used as a local index for some other procedure, did not work.
- Using an indexed color parameter to specify the text color of an explicit index in the **Pivot Table** object, did not work.

Issues ...

AIMMS 3.12 PR - SU7, release date 2013-07-12

In this AIMMS 3.12 Software Update, the following issues have been fixed or improved.

*3.12 PR - SU7
2013-07-12*

- Setting the option `data_management_style` to Disk Files and Folders could result in an incorrect message during AIMMS startup, especially if you got an early message about the solver configuration.
 - Long error messages from `TestDataSource` could corrupt the internal memory leading to severe internal errors or fatal application errors later on.
 - Combining multiple cases and 'prior to case load procedure' could lead to a severe internal error when switching between active cases.
 - Editing values in the **Identifier View** window could result in a crash.
-
- Sometimes the GIS background image of a network object appeared as background of other areas of the page.
 - When running under AIMMS PRO, tooltips in a **Pivot Table** object could cause a crash.
 - The use of indexed unit parameters as the **Unit** attribute of other parameters and variables, could prevent data display of the inactive cases in a multiple case object on a page.
 - In the **CaseLoad** or **CaseSave** dialog box, when using data management style 'Disk Files and Folders', filtering on a specific Case Type did not work correctly when the case type itself was defined in a module or library. In other words, when the full name of the case type contains a module or library prefix.

Issues ...

...GUI

- The postsolve step used by **CBC** and **Gurobi** could return incorrect variable values. *... Solving*
- Procedure `GMP::Coefficient::Set` could result in a severe internal error if the row was not generated before.

- Expressions involving atomic/compound index translation might erroneously be translated into identifier references, resulting in incorrect data to be displayed in the GUI and display statements. *... Compiler and Execution*
- A `Raise Error` statement, guarded by a `block / onerror / endblock` statement, but not marked as handled, might not stop AIMMS.
- The use of UNDF in data of an identifier and used as operand in the AND operator, could result in unjust use as a truth value in indexed operations.
- The `FileCopy` function did not check whether the destination file was already opened by some other process on Linux.

AIMMS 3.12 PR - SU6, release date 2013-04-03

In this AIMMS 3.12 Software Update, the following issues have been fixed or improved.

*3.12 PR - SU6
2013-04-03*

- There is a new version of **Gurobi**, i.e. **Gurobi 5.1**. **Gurobi 5.1** is not supported on 32-bit Linux machines. *Improvements*
- **CPLEX** version 12.5 has been upgraded to 12.5.0.1.

- Sometimes the GIS background image of a network object appeared as background of other areas of the page. *Issues...*
- The **XML Schema Mapping** tools stopped working because of the upgrade to MSXML 6.
- For some of the calendar properties of a **Gantt chart**, the dialog did not accept identifiers that were declared in a library.

- The implementation of the procedure `GMP::Coefficient::Set` had become inefficient. *... Solving*
- Solution information is now available for **Gurobi** if the model is infeasible, the presolver is turned off, and the model type is LP or RMIP.

- In a read from file statement, filtering on an index over a defined set, could issue superfluous error messages of elements outside the set. *... Compiler and Execution*
- After deleting an element from a set and a call to `CleanDependents`, a **Pivot Table** that was already open, could crash.
- An element parameter `ep` with range `S`, whereby `S` is not a subset of Integers, and definition `if someCond then '0' endif` in conjunction with

other definitions involving the set `Integers`, could rise superfluous error messages.

- Consider a set `S` subset of `T`, `T` is empty. Consider a read statement: `Read A from file "myFile.txt";` (`S` will not be read in, as `S` is not mentioned in the identifier list of the read statement). The file `myFile.txt` has contents: `S := { a, b, c }`. Then AIMMS would erroneously assign the elements `{a, b, c}` to `T`.
- The function `SectionIdentifiers` would also return elements for prefixes which would result in superfluous errors when its result is checked against the predeclared set `AllIdentifiers`.
- Consider a set `S`, subset of `M`, index `i` in `S`. The set `S` has data `{ a }`, the set `M` has data `{ a, b, c }`. The set `S` may have a definition, but not a constant one; it cannot be determined at compile time that `b` will never be an element of `S`. A test like `i = 'b'` in the body of a procedure or in the definition of a parameter, variable or constraint could trigger a domain error message. This check is now with respect to a superset of `S` with constant definition, or with respect to its master set `M`.
- AIMMS now supports units declared in the `SI_Time_Duration` quantity such as "week". Such a time unit needs to be based on one of the granularities `{ seconds, minute, hour, day, month, year }`. In addition, there is a warning added against huge calendars.

AIMMS 3.12 PR - SU5, release date 2012-12-28

In this AIMMS 3.12 Software Update, the following issue has been fixed.

*3.12 PR - SU5
2012-12-28*

- When AIMMS needed to allocate huge blocks of memory internally, in combination with a solve, crashes and/or memory errors could occur.

AIMMS 3.12 PR - SU4, release date 2012-12-17

In this AIMMS 3.12 Software Update, the following issues have been fixed or improved.

*3.12 PR - SU4
2012-12-17*

- A new 'border color' property has been added on the 'General' tab of the **Pivot Table** object's properties. It can be used to specify the color of the line border (if present). *Improvements*
- Added support for the use of AIMMS in combination with Adobe Acrobat X (i.e. Adobe Acrobat 10 (or higher)). *Issues ...*
- Warnings issued by selected intrinsic functions were issued to the message window, but not to the error/warnings window when the return value of that intrinsic procedure was not checked and the option "intrin-

sic procedure error handling” was set to “Raise when not checked” or “Raise warning when not checked”.

- In rare cases, retrieving the number of OLE DB providers on your system could result in a fatal application error.
- When an ordinary parameter, say a, is used in the unit attribute of another parameter, variable or constraint, say parameter b, together with a unit u, with u <> [-], then opening the attribute window of a parameter, variable or constraint could sometimes result in a severe internal error.
- When using the Model Edit Functions and creating a procedure, using and removing that procedure and then creating a procedure that uses indexed string parameters alternated with rebuild statements, could lead to a severe internal error.
- Indexed parameters and variables with an index domain condition and referenced inside an expression, omitting one or more arguments, could cause a severe internal error. For instance:

```
PARAMETER:
  identifier : a
  index domain: (i,j,k,l) | d(j,k) ;
PARAMETER:
  identifier : b
  definition : sum( (i,j,k,l), a(i,,k,l) ) ;
```

- An indexed element parameter ep(i), containing a value 'noLonger', and 'noLonger' is no longer an element in the range of that element parameter, could still have that element 'noLonger' displayed when ep(i) is written in a write statement.
- A put file with its device attribute set to 'Window', and its mode attribute set to 'replace' could cause a crash.

- When using a non-default for the X-Axis tick spacing property of the **2D Chart** ActiveX object, the chart was shifted one position to the right. *... GUI*
- In an **Indexed Page** object, if during a procedure run both the set that is used for indexing and the contents of the inner pages changed, the inner pages that could be re-used were not updated correctly.

- The **Pivot Table** object was not always drawn correctly after changing an element in the first visible row of the row tree (or the visible first column in the column tree). *... Pivot Table*
- The **Pivot Table** object did not always correctly restore the position of artificial indices from the state file.
- The default sort order of identifiers in a **Pivot Table** object is now taken from the order in which they are specified in the property dialog.
- A **Pivot Table** object with artificial and dense indices, and the identifier index somewhere in the middle of the row or column header, could sometimes display an empty row or column.

- Creating a macro and subsequently specifying the arguments and definition of that macro in an attribute form, might lead to a severe internal error. *... Syntax Editor*

- A RMIP problem could be solved as MIP by CPLEX after modifying the model type using `GMP::Instance::SetMathematicalProgrammingType`. *... Solving*
- CPLEX could hang after changing the model type to RMIP while using parallel threads and callback procedures.
- In rare cases, the **Math Program Inspector** would not open if the model type was not specified for the math program.

- Unit parameters with a definition referencing the `.unit` attribute of parameters, variables, and constraints, where these unit parameters again might reference defined unit parameters, could cause a problem in the compilation order which would subsequently lead to superfluous compile time unit consistency warnings or runtime unit consistency errors. *... Units of Measurement*

AIMMS 3.12 PR - SU3, release date 2012-09-21

The following issues have been fixed or improved in this third Software Update of the AIMMS 3.12 Production Release.

*3.12 PR - SU3
2012-09-21*

- The navigation menu showed one ancestor too many. *Issues*
- A **Pivot Table** object with auxiliary indices specified by an element parameter would crash after a data change (e.g. a case load).

AIMMS 3.12 PR - SU2, release date 2012-08-27

The following issues have been fixed or improved in this second Software Update of the AIMMS 3.12 Production Release.

*3.12 PR - SU2
2012-08-27*

- **GUROBI** version 5.0 has been upgraded to 5.0.1. *Improvements*
- In page edit mode, when moving many page objects, the drawing of the outline goes much smoother.
- Reading and writing cases over a network could be rather slow. The files are now cached on the local disk.

- The IDE no longer allows you to set a breakpoint on an Assertion. *Issues ...*
- If the `.prj` file is read-only, but the `.amb` is *not* read-only, the attributes of the model can no longer be edited.

- The **Gantt Chart** object could crash during a drag operation, if the start and length of some of the bars were not properly defined. ... GUI
- In page edit mode, creating or moving a **Circle** object no longer messes-up the screen.
- A **Table** object in an **Indexed Page** object on a print page was not displayed correctly.
- In the Page Manager, if a Page Reference was linked to a page that is one of the 'ancestors' of the Page Reference itself, then this could result in an infinite loop in the **Tabbed Page** object or **Navigation** object.
- Value bars in the **Pivot Table** appeared as small lines in case the 'value bar maximum' had been specified as 0 while the value itself was strictly larger than 0.
- In the **Pivot Table**, the first visible row and column were not always restored correctly (after a data change).
- AIMMS no longer warned when a value is entered on an AIMMS page which is outside the range as declared in the model of the corresponding identifier.

- **BARON 10.3** was incorrectly linked against **CPLEX 12.3** instead of **CPLEX 12.4** which could result in a crash. ... Solving
- The `.violation` suffix of constraints and variables was not always filled if multiple threads were used by the solver.
- A solve with **GUROBI** after using `GMP::Column::SetAsObjective`, could result in an error.
- If the function `GMP::Instance::CreateSolverSession` was called with the same `Name` argument, without deleting the solver session in between, AIMMS could crash.
- Consider a constraint of the form:

```

if (v1 >= 0) then    ! Condition involving variables but they are all frozen
  x - y             ! then sub-expression linear
else
  y - x             ! else sub-expression linear
endif
>= 0

```

For such a constraint, AIMMS would erroneously conclude that it is non-linear.

- The unit attribute of a calendar did not respect namespaces which might lead to superfluous error messages when both the time unit and calendar were declared in the same module or library. ... Compiler and Execution
- The `%s` modifier in `FormatString`'s first argument now also accepts expressions of data type element, numeric and unit.
- Reading a section using the intrinsic Model Edit Function `me::ImportNode`, whereby the section contained procedures without a body, could lead to a severe internal error.

AIMMS 3.12 PR - SU1, release date 2012-06-11

The following issues have been fixed or improved in this first Software Update of the AIMMS 3.12 Production Release.

*3.12 PR - SU1
2012-06-11*

- The flavor, architecture and platform information can now be retrieved for a running AIMMS instance. It is available via the predeclared identifier `AimmsStringConstants`.
 - There is a new version of **CBC**, i.e. **CBC 2.7.5**.
 - There is a new version of **IPOPT**, i.e. **IPOPT 3.10.1**.
 - There is a new version of **GUROBI**, i.e. **GUROBI 5.0**. **GUROBI 5.0** can solve models with quadratic constraints and second-order cone constraints.
 - There is a new version of **BARON**, i.e. **BARON 10.3**. **BARON 10.3** can use the NLP solver **IPOPT** and the LP solver **CLP**. **BARON 9.3** has been removed from the installation, but is available upon request.
 - **CPLEX** version **12.4** has been upgraded to **12.4.0.1**.
- Improvements*
- Dragging and dropping between different AIMMS sessions (which is not supported) could lead to a crash.
 - The **New Case** dialog boxes now handle slow network locations much faster.
 - The **State File Manager** did not find all references to the listed states, and therefore marked too many states as 'Unused'. Furthermore, the scanning of the pages is a bit quicker now.
 - The Index drop-down in the select-identifier wizard could show too many irrelevant entries (e.g. invisible identifiers).
- Issues ...*
- The page properties 'Title' and 'Hidden' did not respond correctly on name changes of the corresponding identifier.
 - For GIS feature layers (e.g. GML and ESRI), setting the point style and size had no effect.
 - Specifying the empty string as the file for a (GML or ESRI) GIS source could lead to a crash.
 - The Identifier Info dialog, and the Declaration button in the Search dialog, did not find the position in the tree of predeclared identifiers.
 - The exclamation mark in front of a duplicated template in the Template Manager was not always correctly cleared when the template was re-saved.
 - Having an **Indexed Page** object that referred to a deleted inner page, could lead to a crash or other strange problems.
 - When resetting from a zoom action in an **ActiveX 2D Chart object**, the chart wasn't always presented as it was before the zoom action.
 - Drag-and-Drop from the Page Manager tree (or most other trees) to the text of an attribute window, did not copy the text.
- ... GUI*

- Columns in the row header of the **Pivot Table** object, of which the length was specified by a numerical parameter, or of which the element text was specified by a string parameter, were not always drawn with the proper width. *... Pivot Table*
 - If, in a **Pivot Table**, the setting 'Auto resize column widths' has been set to 'all data changes', the column widths are now also resized when the data of an identifier that controls the formatting (e.g. the number of decimals) of an identifier that is shown in the **Pivot Table**, is changed.
 - In the **Pivot Table**, the single element filtering box did not select any label while typing in the edit field at the top of the dialog box.

- **IPOPT** could fail if a model only contained fixed variables (i.e., variables for which the lower bound was equal to the upper bound). *... Solving*
 - `GMP::Solution::SendToModel` did not fill the `.violation` suffix of variables and constraints.
 - **MINOS** is now part of the Linux 32-bit and 64-bit installations.
 - If **CPLEX** hit a time, node or memory limit for a MIP while no integer solution was found yet, AIMMS would set the model status to 'IntegerInfeasible' instead of 'IntermediateInfeasible'.
 - The AIMMS NLP presolver would remove the `.lower` and `.upper` data of rows eliminated from the generated mathematical program.
 - Affine variables constructed from an adjustable variable and uncertain parameters declared in other libraries, can now be referenced.
 - Solving a robust optimization problem containing adjustable variables with an empty dependency, could lead to incorrect results.

- A scalar stochastic variable whereby the range contains references to a parameter could cause a severe internal error during compilation. *... Stochastic Programming*
 - The stochastic Benders method could produce erroneous results when stochastic parameters were referenced in the range attribute of stochastic variables. Now an error message, suggesting a workaround, is generated.
 - Stochastic Benders could use incorrect nominal values for stochastic data, which could result in a non-optimal solution.

- `FileTime` returned a time representation that did not include the correct daylight saving time correction. *... Compiler and Execution*
 - The option `Warning_index_domain_violation` will now check with regard to the root set in order to avoid superfluous warnings.
 - Ctrl-Shift-S to interrupt a current run sometimes responded too slowly.
 - Combining Robust Optimization with violation penalties could cause a severe internal error when opening the identifier cardinalities dialog.
 - Consider a procedure `Pr(inout:P1(i))`, whereby `P1(i)` is a 1-dimensional parameter, and a call to this procedure: `Pr(P3(j, i, k))`; where `j` and `k` are running indices. Then when passing a `P1` without non-default elements

back into P3 with non-default elements, could result in not removing the non-default elements of P3.

- Consider two sparse indexed unit parameters, say UP1(i) and UP2(j) with defaults unit1 and unit2 whereby unit1 or unit2 not equal to the unit [-]. Then a defined unit parameter UP3(i, j) defined as UP1(i) * UP2(j) could be computed erroneously.
- Some additional checks are added before executing a procedure scheduled using ScheduleAt.
- Special numbers (like undf, zero, na) could be parsed as keywords, which caused expressions that referenced those special numbers to be incorrect (red or wrong highlighting).
- CleanDependents affecting the .rowno of a constraint or the .colno of a variable, did not remove the generated mathematical programs which referenced those constraints/variables.

AIMMS 3.12 PR, release date 2012-03-07

In this AIMMS 3.12 Production Release, the following issues have been fixed or improved.

*Build 3.12 PR
2012-03-07*

- Reading numerical database data, using a **Database Procedure** or a **Database Function**, into string or element parameters, could lead to a loss in numeric precision, as only 6 decimals were used. This has been increased to 15 decimals, which is the maximum precision of the internal AIMMS numeric data type.
- Using database tracing in combination with database table or column names containing a percentage sign, could lead to a crash.
- The error/message window would pop up without any message, if there were more local *handled* errors than the option `maximal_number_of_execution_errors_reported`.
- In the Function Reference, the functions `CaseFileGetContentType` and `CaseFileSectionGetContentType` were not present. Instead, they were named `CaseFileGetContents` and `CaseFileSectionGetContents`, resp.
- Recompiling a model with an inline variable, with its definition referring to its own level value, could lead to a superfluous compilation error message.
- The solver status and the program status were not correct if **BARON** was running out of memory.
- An index domain condition on a constraint of the form $\text{ord}(t) = 7$ (where 7 is cardinality of set Time) could incorrectly cause the constraint to be not generated.
- The value of a scalar string parameter written to an XML file could be incorrect, if another string parameter was also child of the same `ElementMapping` node in the mapping file.

Issues ...

- Data of local identifiers is no longer available after a handled error in the debugger.
- The **Math Program Inspector** could crash in some situations in the 64-bit Windows version of AIMMS.
- The property 'Location in Initial View' for the suffix index of a **Pivot Table** object, was not defined as an option with keywords.
- AIMMS could crash when specifying the properties of a **Pivot Table** object, if the identifier index was an outer index, and open in the property tree editor, while at the same time an update was being triggered.
- In the **Pivot Table** object, you can now specify the background color per element (in the row and/or column tree).

AIMMS 3.12 FR3, release date 2011-12-22

Beside the new features mentioned in **What's New in AIMMS 3.12**, the following issues have been fixed or improved in this AIMMS 3.12 Feature Release 3.

*Build 3.12 FR3
2011-12-22*

- The contents of the **Math Program Inspector** can now be copied to the clipboard (by selecting a tab and pressing CTRL+C).
- The GMP-AOA algorithm now starts with a preprocessing step (using the nonlinear presolver) by default.
- The procedure `GMP::Instance::SetCallbackAddLazyConstraint` was added, which can be used to add lazy constraints inside a callback. It is supported by **CPLEX 12.3**. See the Function Reference for more information.
- The 'Insert/Delete Row/Column Allowed' option of the **Pivot Table** object has been split into two separate options (one for Insert and one for Delete).
- The **Pivot Table** object has been extended with an index filter, based on wildcard pattern matching. Only * and ? are allowed as wildcard characters. ? matches against a single arbitrary character, whereas * matches against a sequence of zero or more arbitrary characters.
- In a newly created **Pivot Table** object, subheaders will not be displayed anymore by default.
- The AIMMS startpage has been improved. Links to the AIMMS blog articles are now present, and your default web browser is now used to open any links to web pages (as opposed to before, when Internet Explorer was always used). Furthermore, the Google Group/Latest News/AIMMS Blog features are more robust against errors.

Improvements

- Tagged arguments, used in combination with unit expressions could cause the syntax highlighting to fail. An example of this is a call like:
`SomeProcedure(argumentName : 14 [s]);`
- The wizard for specifying the arguments of a procedure has been im-

Issues...

proved. The old version could lead to crashes in specific situations.

- The wizard for specifying the arguments for (external) procedures and functions has been improved.
 - If you had the the profiler active, and you were editing text containing compilation errors, performing a check (&commit) action, could cause AIMMS to crash.
 - Some unit-related wizards did not work correctly when used for quantities that were declared inside a (namespaced) module.
 - The **Add** button in the **Floating Index** object no longer adds elements to subsets of the referred set.
 - When running with the `--as-server` command line argument, a call to the function `DebuggerBreakpoint` is now ignored.
 - A timeslot format attribute of a calendar such as "%d %Am|myMonthNames| %y" uses `myMonthNames` as month name set. When such a month name set does not have any data and has empty index/parameter attributes, a severe internal error could occur.
 - Reading from database tables in an Oracle 11 database, could result in a crash.
 - Using a read-filter of 0 in an XML mapping file, could result in a crash.
-
- The two **Math Program Inspector** related options `feasibility_tolerance` and `bound_tolerance` have been renamed to `MPI_feasibility_tolerance` and `MPI_bound_tolerance`, respectively. Should you have set either of these options to a non-default value, you should set these values again when using the new AIMMS version.
 - The IIS computation in the **Math Program Inspector** now also takes variable bounds into account.
 - The **Math Program Inspector** did not always recognize dual degeneracy correctly.
 - The **Math Program Inspector** Bound Analysis now works correctly when applied to a pertubated model (e.g. after having determined an IIS).
-
- Data changes in identifiers that were used as element text in a **Pivot Table** object, caused the column width to be adapted automatically, even if the option `AutoSizeColumnWidths` had not been set to `All Data Changes`.
 - Selecting a cell range did not immediately work after a **Pivot Table** object had been initialized (the user had to explicitly click a cell first - to set the first cell of the selected range - and use `SHIFT+click` in another cell to actually make the selection).
 - The position of an artificial index in a **Pivot Table** object was not correctly restored after closing and opening a page.
 - In the **Pivot Table** object, a column tree without any indices now uses the column width from **Defaults - Column Width** on the **Indices** tab of the **Pivot Table Properties** dialog.
 - The `Store Entire Tree Layout` option disappeared after specifying a non-

... *Math
Program
Inspector*

... *Pivot Table*

default Standard Format for an explicit identifier in a **Pivot Table** object.

- In a **Pivot table** object, when fixing one of the indices of an identifier to a specific element or to an element parameter, showing a table with suffices and/or with multiple cases could go wrong.
 - After introducing + and - as shortcut keys to open/close an intermediate node in the row or column tree of the **Pivot Table** object, the + and - keys could not be used anymore to start editing a cell value.
 - The index filter of the **Pivot Table** object did not work correctly if it was applied to the multiple case index.
-
- Bound reductions done by the procedure `GMP::Instance::CreatePresolved` could be incorrect for models with indicator constraints. *... Solving*
 - The original **CPLEX** solution was not passed to AIMMS in case the post-solved MIP model was infeasible.
-
- While processing scenarios in `GMP::Instance::GenerateRobustCounterpart` for models using `ConvexHull`, the peak memory usage was too high. *... Robust Optimization*
-
- AIMMS suffered from a crash in the model editor, when analyzing the coloring for `<variable>.stochastic.upper`. *... Stochastic Programming*
 - When using stochastic parameters and/or stochastic variables in the objective definition of a mathematical program, superfluous error messages could be issued which referenced the identifier `IndexCopyActive-StochasticScenarios`.
-
- Root sets of actual and formal arguments of procedures and functions, when passed as handles, were not compared, which resulted in missing error messages. *... Compiler and Execution*
 - Consider a parameter `P(i)`; `i` is declared over a set `S` with an `Order` by expression `OS(i)`, and `P` has a definition of the form:


```
if ( i = first(S) ) then 1 else
P(i-1) * someFac(i) endif
```

 In addition, in a body of a procedure, there is a statement like


```
1: for ( i in fewS ) do
2: A := f(P(i));
3: endfor;
```

 In line 1: `i` is assigned a new range without ordering. In line 2: the definition of `P` is re-evaluated, only needed in the first encounter. Results could be erroneous, because when evaluating the definition of `P`, the set `fewS` could be used as range of `i` instead of `S`.
 - The command line option `--run-only` now exits AIMMS if the given procedure name cannot be found.
 - AIMMS now issues an error message on statements like `Write Integers to ...`, which would be very slow and not usefull.

AIMMS 3.12 FR2, release date 2011-08-24

In this second AIMMS 3.12 Feature Release, the following issues have been fixed or improved.

*Build 3.12 FR2
2011-08-24*

- There is a new version of the **CPLEX** solver: **CPLEX 12.3**. This version fixes a memory issue in **CPLEX 12.2** where **CPLEX** could issue out of memory errors prematurely on 64-bit Windows systems. Furthermore, **CPLEX 12.3** solves a greater variety of nonconvex quadratic programming (QP) models.
 - The solvers **CPLEX** (version 12.1 and higher) and **GUROBI** (version 4.5), can be used to calculate the unbounded ray for an unbounded LP model. This is controlled by the option `unbounded_ray`.
 - The solvers **CPLEX** (version 12.1 and higher) and **GUROBI** (version 4.5), can be used to retrieve a Farkas infeasibility proof for an infeasible LP model. This is controlled by the option `Farkas_infeasibility_proof`.
 - The refresh rate of the **Progress Window** has been improved.
 - The function `GMP::Solution::RandomlyGenerate` has been extended with two optional arguments, which can be used to generate random points around a starting point.
 - Tracing information is now also provided for the Truncate Table and Empty database statements.
 - The **Column Title** property and the **Tooltip** properties of the **Composite Table** object have been moved to a new property tab **Text**.
 - The 'Text Color' property of the Identifier Index (on the **Indices** tab of the **Pivot Table** properties) can now be specified as an element parameter in `AllColors` indexed over the set `AllIdentifiers`.
 - Two default settings for new **Pivot Table** objects (that are not displayed on a data page) have been changed: 'hide if single identifier' has been set to 'No' for new **Pivot Table** objects. In addition, the default location for the identifier index has been set to the 'Outer Index'.
 - The **Pivot Table** copy command (CTRL + C) now also copies unit information of the headers (if present).
 - When a **Pivot Table** object contains multiple identifiers with different domains and the identifier index is displayed in the row area, an extra horizontal line in the row area will now be added.
-
- After opening the **Body call** wizard of an external procedure, the file handle to the `.dll` file remained in use, even after the AIMMS project had been closed already.
 - In the AIMMS **COM object**, the function `RetrieveTable` could yield unexpected results when dealing with ordered sets. This could also be noticed in the AIMMS **Excel add-in**, since it relies on the **COM object** for its communication with AIMMS.

Improvements

Issues ...

- The file `comctl32.ocx` has been added to the installation file again, since this file is not present on pristine Windows XP installations.
 - The **Nodelock Manager** (`NdLckSrv.exe`) was not installed correctly, which could incorrectly result in the error message "You need a newer version of the Nodelock server (`NdLckSrv.exe`), that can handle an alternative location for AllUsers license files".
 - A statement of the form: `Empty aPar in someDbTable(i);` in which the scope of index `i` is not specified, could cause a severe internal error.
-
- In a **Gantt Chart** object with the new overlapping bars feature enabled, navigating from bar to bar using the arrow keys did not always work correctly. *... GUI*
 - A **Gantt Chart** object could mess up the undo stack, which led to errors when trying to undo changes.
 - The **Calendar** ActiveX object was changed, such that scrolling through the months (using the left and right arrows in the header field), does no longer change the associated date identifier(s). As a result, the `OnChange` procedure is not called anymore when doing so (since no dates have been changed). This change prevents problems with the month scrolling. Combined with a slow `OnChange` procedure, the scrolling didn't stop after one month.
 - A literal string specified as a tooltip in the **Composite Table** object, was not displayed.
 - The property dialog of the **Composite Table** object incorrectly accepted a literal 0 or 1 as a value for the property 'Column Defaults - Hide'.
-
- In the **Pivot Table** object, the displaying of nodes without any child nodes that were related to a dense index, could lead to a misalignment between the tree area and the grid area, resulting in data being displayed in the wrong row. *... Pivot Table*
 - After pivoting a header in the row tree of a **Pivot Table** object, it could happen that the header labels were not correctly centered.
-
- The **Progress Window** is now more responsive when using the proxy stub solver. *... Solving*
 - AIMMS could become unresponsive while solving a difficult problem with **Gurobi**. The option `progress_awareness` has been added for Gurobi 4 and higher to make AIMMS more aware of progress updates.
 - When the function `GMP::Instance::CreateDual` generated a GMP, the optimization direction was incorrect for a math program with a violation penalty.
 - The incumbent suffix of a math program showed an incorrect value after the solve, if the objective contained a constant term.
 - Combining violation penalties with Robust Optimization could lead to severe internal errors.

- In rare cases, when using the Excel functions, from within AIMMS, Excel could ask about saving a workbook seemingly endlessly, and produce a great many EXCEL.EXE processes in the task manager.
 ... Excel
- A call to `Spreadsheet::SetActiveSheet` with an Excel workbook as argument, now actually sets the focus to the new active sheet.

- A fatal error could occur in the following situation: A set S (with index i) is initialized in `MainInitialization` and a parameter $P(i)$ is initialized via a scalar value in its initial data attribute.
 ... Compiler and Execution
- The error message of `CaseCreatedDifferenceFile` has been improved for the situation in which the referred case does exist, but does not have data saved into it (yet).
- During the editing of a constraint, superfluous error messages were generated when extending the constrains index domain with a new index, and using that index in the definition of that constraint without an intermediate commit.
- When a horizon with index h is a re-ordered subset of a calendar, and its planning period is a small subset of the entire horizon, and a parameter $P(h)$ has a definition in which $P(h-1)$ is referenced, a severe internal error could occur or the results could be unexpected.
- When analyzing the coloring for `<a_variable>.stochastic.upper` in the model editor, AIMMS crashed.
- In AIMMS 3.12 a severe internal error could occur when removing an existing set range of a local element parameter.

AIMMS 3.12 FR1, release date 2011-05-25

In this first AIMMS 3.12 Feature Release, the following issues have been fixed or improved.

*Build 3.12 FR1
2011-05-25*

- All the prerequisites (except the Sentinel drivers and .NET) for an AIMMS version have been moved into the .msi installer database. Separate downloads of these prerequisites are therefore no longer necessary.
 Deployment changes

- The Model Edit API function `AimmsMeNodeExists` has been added (see the Language Reference for details).
 Improvements
- The new option `MINLP_probing` controls whether the Nonlinear Presolver should attempt to apply (very basic) probing to constraints with binary variables.
- The function `Spreadsheet::ClearRange` (which was previously known as `ExcelClearRange`) has been extended with an optional argument, called `IncludeCellFormatting`. When set to 1, the cell formatting is cleared as well.

- It is now possible to use a **Scalar** object to let the user enter a password. The entered text will be displayed using a sequence of '*' characters.
- The **Network** object now has support for OpenStreetMap when using a GIS object as its background.
- All the **Excel**-related functions have been renamed from Excel... to Spreadsheet::.... For example, the function ExcelAssignTable is now called Spreadsheet::AssignTable. The reason for this is the addition of **Open Office Calc** support. Although the old function names can still be used, we encourage you to use the new naming.
- When an attribute window was resized (larger in both vertical and horizontal direction), sometimes the contents were not updated correctly.

- **Case Types** and **Data Categories** that were setup using references to sections in the model tree (as opposed to explicit references to individual identifiers) did not work correctly.
- A library or a main model is now read-only if its associated name change file is read-only.
- In some situations, the appearance of the error window could cause AIMMS to become unresponsive for quite some time.
- When specifying an unknown printer in the ActivePrinter argument of the function Spreadsheet::Print (previously called ExcelPrint), AIMMS now gives an error message.
- When using the Excel functions from AIMMS, an EXCEL.EXE process remained persistent in the task manager upon closing the AIMMS project.
- When using dual monitors with the secondary monitor positioned at the left of the primary monitor, the right mouse menu in an attribute window popped up on the wrong screen.

Issues ...

- When entering an element name in a page object (e.g. a **Table** object), leading and trailing spaces were removed before matching them against existing element names. Now, the name is first matched exactly as typed, and if no match is found a second attempt is made without leading and trailing spaces.
- If the list of actions for a **Button** object or a **Menu Item** only contained one item, that action could not be deleted.
- When creating a new **Gantt Chart** object for data that is linked to a calendar in the model, the information in that model calendar is now used to initialize the (x-axis) calendar of the Gantt chart.
- In the **Gantt chart** object, when bars overlap each other, selecting a bar with the mouse has been improved.
- When the **Title** property of a page was specified by an expression containing more than 100 characters, this specification was silently removed after saving and re-opening the page. If the expression contained newline characters, the page could not be read back at all.

...GUI

- When using the **Pivot Table** object, reverse links on the identifier index did not work for identifiers that had their 'Title' property set. *... Pivot Table*
 - Changing the option 'show subheaders' in a **Pivot Table** object now also triggers a refresh of the Pivot Table.
 - The **Pivot Table** copy command (CTRL + C) now also copies unit information of the headers (if present).
 - **Pivot Table** objects that use an implicit set of identifiers, could trigger a fatal application error.
 - The initial width for **Pivot Table** objects that contained linebreak characters in the elements of the set that is shown as the last index in the row tree, and that do not have an explicit width property specified for this index, was not computed correctly.
-
- In rare cases, AIMMS could crash while deleting solver sessions when closing the project. *... Solving*
 - An out of memory error during mathematical program generation could lead to severe internal errors.
 - AIMMS could raise a severe internal error, when generating the robust counterpart of a mathematical program with violation penalties.
 - Affine coefficients were not saved in cases (affine coefficients are "declared" in the dependency attribute of an adjustable variable, and referenced via the infix `.adjustable`).
 - The postsolve objective value shown in the progress window was not always correct.
 - The procedure `GMP::SolverSession::AsynchronousExecute` isn't supported any longer by solvers that use the Proxy Stub interface.
 - Retrieving the constraints solution for a large model from **GUROBI** could result in a crash.
 - **GUROBI** no longer attempts to write a solution file if no solution exists.
 - Incumbents of a maximization MIP/MIQP/MIQCP problem were not always passed when using **CPLEX**.
-
- The `SyntaxEditor` could not correctly determine the validity of an entry inside a so-called 'context-sensitive token list' (e.g. argument lists or option lists). *... Compiler and Execution*
 - There was a regression on the case ensure subset consistency check; this check might be applied too early in the presence of reference cases, leading to superfluous error messages.
 - Using vector valued functions inside for loops could sometimes lead to a severe internal error.
 - Reading the set `someSet` from a file "someFile.txt" didn't compile when `someSet` was a defined set. Now it is accepted when `someSet` is a subset of `AllIdentifiers`.
 - Element names longer than 255 characters are now correctly flagged as an error (as the limit on element names is 255 characters).

- A severe internal error could occur when specifying the quantity of a unit parameter that already had a default, but that default was not commensurate with the given quantity.