

AIMMS 3.13

Release Notes for Win32 Build 3.13 PR - SU5

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

Paragon Decision Technology

November 19, 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	10
2.3.1 Personal and machine nodelocks	11
2.3.2 Installing an AIMMS license	14
2.3.3 Managing AIMMS licenses	16
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 3.13	25
AIMMS 3.13 PR - SU5, release date 2014-11-19	30
AIMMS 3.13 PR - SU4, release date 2014-07-18	31
AIMMS 3.13 PR - SU3, release date 2014-01-29	31
AIMMS 3.13 PR - SU2, release date 2013-10-16	33
AIMMS 3.13 PR - SU1, release date 2013-07-12	35
AIMMS 3.13 PR, release date 2013-02-13	38
AIMMS 3.13 FR1, release date 2012-10-10	40

Chapter 1

System Requirements

This chapter discusses the system requirements necessary to run the various components of your Win32 AIMMS 3.13 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 3.13 is designed to run under

Supported Windows versions

- Windows Server 2003,
- Windows Vista,
- Windows 7,
- Windows 8,
- Windows Server 2008, and
- Windows Server 2012.

The Win32 version of AIMMS 3.13 Unicode is supported under

- Windows XP,
- Windows Server 2003,
- Windows Vista,
- Windows 7,
- Windows 8,
- Windows Server 2008, and
- Windows Server 2012.

Running the Win32 version of AIMMS 3.13 versions under the 32-bit Windows versions 95/98/ME/2000/NT 4.0/XP 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 3.13, 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, CPLEX 12.2 or higher, and CPOPT 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	11	√	√	√	-	-
CBC	2.7.5	√	√	√	√	√
CBC	2.8.0	√	√	√	√	√
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	√	√	√	√	√
	12.5.1	√	√	√	√	√
	12.6	√	√	√	√	√
CPOPT	12.2	-	√	√	√	√
	12.4	√	√	√	√	√
	12.5	√	√	√	√	√
	12.6	√	√	√	√	√
GUROBI	4.0	-	√	√	√	√
	4.5	-	√	√	√	√
	4.6	-	√	√	√	√
	5.0	-	√	√	√	√
	5.1	√	√	√	-	√
	5.5	√	√	√	-	√
	5.6	√	√	√	-	√
IPOPT	3.10.1	√	√	√	√	√
KNITRO	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 3.13 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 your computer. The purple Sentinel dongle must be connected to a USB port of your computer. The AIMMS 3.13 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>.

AIMMS dongles

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:

Two types of nodelocks

- personal nodelocks, and
- machine 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.

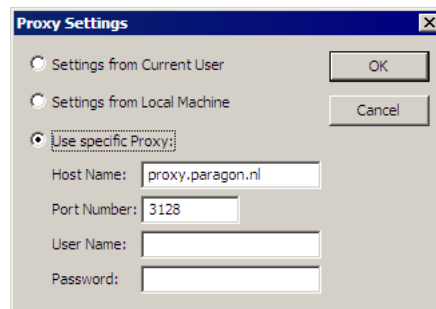


Figure 2.1: The AIMMS **Proxy Configuration** dialog box

2.3.2 Installing an AIMMS license

When you start up AIMMS 3.13 for the first time after installation, AIMMS will open the **License Configuration** dialog box illustrated in Figure 2.2. Through

Managing your AIMMS licenses

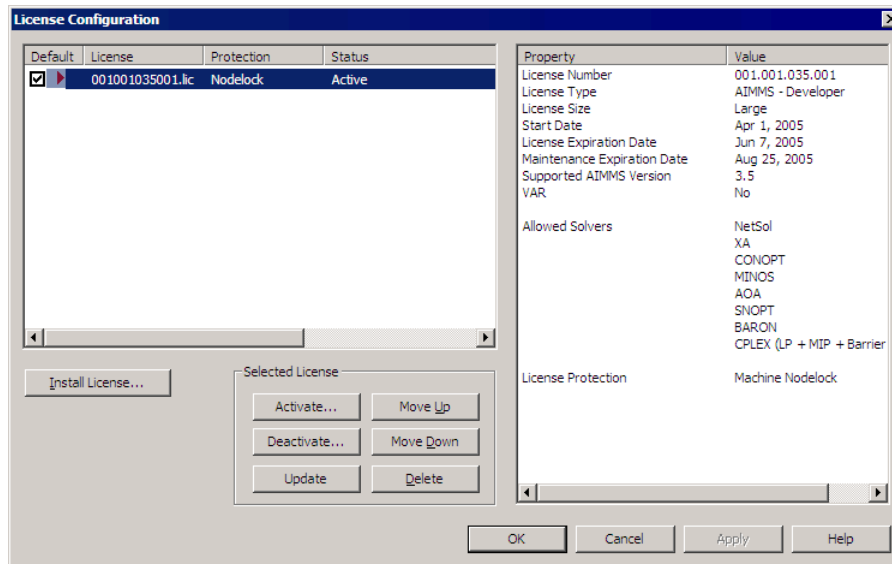


Figure 2.2: The **License Configuration** dialog box

this dialog box you can install new AIMMS licenses and manage all AIMMS licenses that already have been installed on your computer.

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 3.12 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 up to 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

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 3.13 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.

SSLey 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 adhered 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 data-base-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 AIMMS to continuously improve the quality and reliability of AIMMS. This service is automatically called when a severe/fatal error appears. You can read why you should report these errors at [Why Report Errors](#). You can read what information is collected by the AIMMS Error Report Service and what we do with this information at [Data Collection Policy](#).

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 3.13. In addition, it contains the release notes of all AIMMS 3.13 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 3.13
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 3.13

This documentation reflects the state of the first AIMMS 3.13 Feature Release. Compared to the AIMMS 3.12 Production Release, the following major new and extended functionalities have been added to the system:

For **AIMMS 3.13 FR1**, the new features/extended functionalities are:

- New IDE
- Constraint Programming
- 64-bit ActiveX objects
- Column and Row Referencing in GMP Library
- Change the Edit size of resizable Pages

*What's new in
AIMMS 3.13
FR1?*

- 64-bit Excel Add-In
- More Simple AIMMS Installers

AIMMS comes with a completely new IDE (Integrated Development Environment), presenting a different *look* and a slightly different *feel*. The new default *look* uses a display theme which is rather different than previous AIMMS versions, but there are various other themes to choose from. The new IDE has some new features. The most interesting ones are:

New IDE

- Docked pages can now be made *floating*, which, for instance, allows you to put the Model Explorer outside the main AIMMS window (and on to your dual monitor!)
- Pages can be made docked (or floating) as well, which allows you to create a more *cockpit-like* layout of your pages.
- When you restart your application, AIMMS can (optionally) restore all docked and floating windows to the position where they were when you exited the application.
- The new IDE shows the same behavior on 32-bit Windows as on 64-bit Windows operating systems.

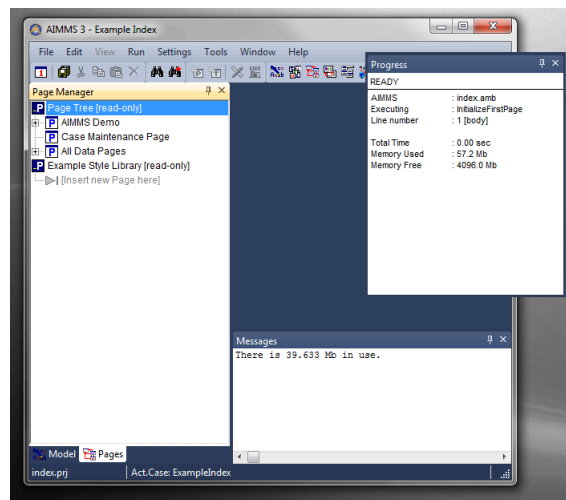


Figure 4.1: The New AIMMS IDE

Constraint Programming (CP) is a relatively new paradigm used to model and solve combinatorial optimization problems; in many cases these are problems that are hard or impossible to solve using classical math programming techniques such as MIP.

Constraint Programming

CP has proven to be most effective on highly combinatorial problem domains, such as timetabling, sequencing and resource constrained scheduling. Typical

problems that can be formulated using CP are Nurse Scheduling, Workforce Optimization, Job Shop Scheduling, Bridge Building, Sport Scheduling, Peg-board Planning, Allocation Problems etc.

In order to leverage its power, we extended the AIMMS language in this AIMMS 3.13 release. Now, you are able to formulate both constraint optimization problems (COP) and constraint satisfaction problems (CSP) as constraint programming models in aimms high level modeling language. AIMMS uses the well-known and respected constraint programming solver **CP Optimizer** from IBM ILOG to actually solve the problems (new solver add-on).

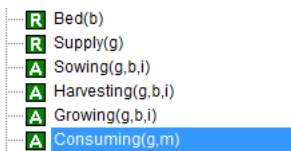


Figure 4.2: Some new Constraint Programming identifier types

With AIMMS 3.13, we now support the ActiveX objects (such as 2D Chart, 3D Chart, Calendar selection, sliders) on both 32-bit Windows and 64-bit Windows. This makes that your Graphical User Interface, built on a 32-bit Windows platform, is now fully portable to 64-bit Windows users.

64-bit ActiveX objects



Figure 4.3: ActiveX objects are now also available in 64-bit AIMMS

The GMP library has been extended to make it easier for you to build your own generic algorithm. When AIMMS generates a mathematical program it translates multi-dimensional variables and constraints into one-dimensional vectors of columns and rows (which are parsed to the solver). With the new GMP functions `GMP::Instance::GetColumnNumbers` and `GMP::Instance::GetRowNumbers`, it is possible to retrieve a set of columns or rows that is generated for some set of variables or constraints. GMP functions now accept a column or row number as input for a column or row argument. This makes it possible, for example, to collect all coefficients in the coefficient matrix corresponding to the integer variables in a generic way:

Column and Row Referencing in GMP Library

```
ColNrs := GMP::Instance::GetColumnNumbers(myGMP, AllIntegerVariables);
RowNrs := GMP::Instance::GetRowNumbers(myGMP, AllConstraints);
```

```

for (r, c) do
  coef(r, c) := GMP::Coefficient::Get(myGMP, r, c);
endfor;

```

Here ColNrs and RowNrs are subsets of the set Integers with index c and r respectively.

Other GMP functions that have been added are:

- GMP::Instance::GetObjectiveColumnNumber
- GMP::Instance::GetObjectiveRowNumber
- GMP::Solution::SetColumnValue
- GMP::Solution::SetRowValue

Via the properties of a page you can now change the *edit size* of a page. For existing applications, especially those that were created in a time when monitors were much smaller, pages with resize lines are sometimes still very small when put in edit mode. This new feature allows you to resize the edit size of such a page, using the resizing algorithm itself to reposition all page objects.

Change the Edit size of resizable Pages

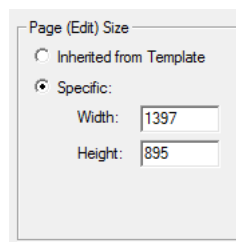


Figure 4.4: Specifying a new page edit size

In addition to the 64-bit Windows support of the full IDE features and ActiveX objects, AIMMS 3.13 FR1 also supports 64-bit Microsoft Excel through the AIMMS Excel Add-In. This means that both 32-bit and 64-bit Microsoft Excel can communicate with 32-bit and 64-bit AIMMS versions through the Excel Add-in (before, this was limited to 32-bit Excel only). This compatibility is available for both the ASCII and the Unicode AIMMS version, offering you easier deployment and more integration possibilities.

64-bit Excel Add-In

The AIMMS .msi installer and the AIMMS prerequisites are now packaged into one (.exe) installer, that will ensure that the prerequisites are installed before installing AIMMS itself. Because the prerequisites are (under the hood) installed as a separate component, they remain eligible for updates separately, i.e. through Windows Update. In addition, we improved the graphical interface that guides you through the setup. The installer has been redesigned to the most minimalistic approach needed for an installer: just three clicks and

More Simple AIMMS Installers

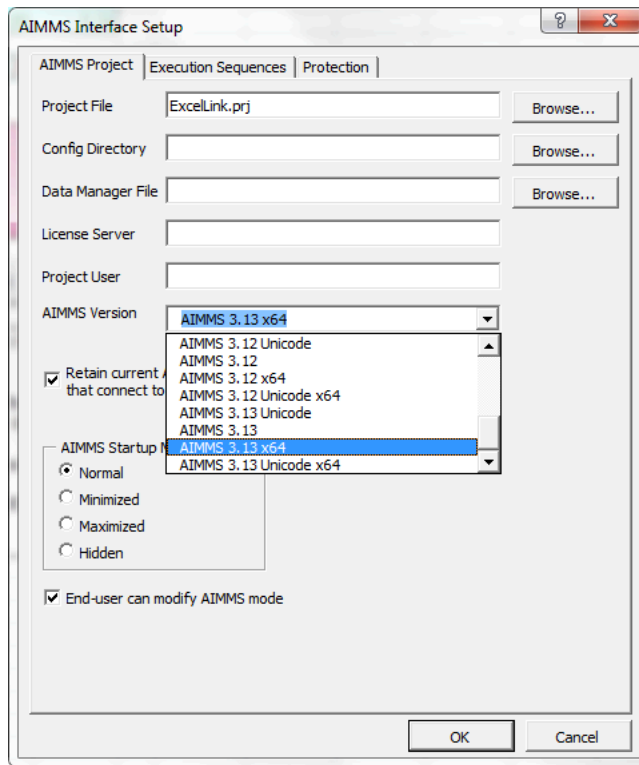


Figure 4.5: The Excel Add-in is now fully 64-bit compatible

you're done. In case of installation issues, a log file can easily be produced that can be used to receive and provide better support.

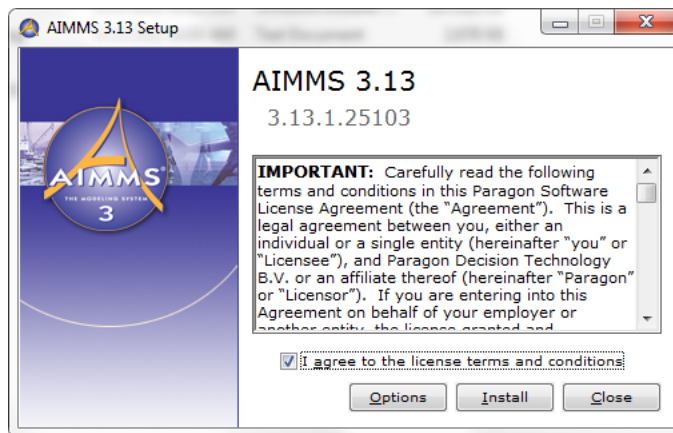


Figure 4.6: A simpler installation process

A new solver version has been added to AIMMS 3.13 FR1:

Solver Update

- **BARON 11**

BARON 11 comes with a wealth of new branching, relaxation, convexity exploitation, local search, and range reduction techniques. You will see improvements in many problems, including convex ones. In addition, **BARON 11** offers systematic treatment of missing bounds through safe LP relaxations. Users will see far fewer messages about missing bounds from **BARON** from now on, while many more problems will be guaranteed global.

AIMMS 3.13 PR - SU5, release date 2014-11-19

In this fifth Software Update of the AIMMS 3.13 Production Release, the following issues have been fixed or improved.

*3.13 PR - SU5
2014-11-19*

- **CPLEX 12.6** and **CP Optimizer 12.6** have both been upgraded to version 12.6.0.1.

Improvement

- Unfortunately, **Yahoo** has stopped offering their Yahoo Maps service. For the **GIS Network** object in AIMMS, this means that you can not select maps from Yahoo Maps anymore.
- If two different pages had the same user defined menubar, but a different toolbar, the keyboard shortcuts in the menubar could disappear.
- The model editing function `me::ImportLibrary` corrupted the list of regular libraries in the project. This could lead to AIMMS ending up in an infinite loop.
- If the **Network License Server** exited unexpectedly, AIMMS only tried to reconnect to the same License Server and not to any other License Servers in the license configuration.
- `Spreadsheet::RetrieveTable` sometimes did not work properly when its arguments involved local sets.
- Committing a local numeric parameter with a non-atomic unit after initial compilation could result in an erroneous scaling of the initial value.
- AIMMS could produce a severe internal error after the error "The limit of 32700 Identifiers is exceeded." was issued.
- When two or more defined parameters and sets were referencing each other, AIMMS could go into an infinite loop determining whether or not these definitions were constant.
- There was a performance problem with inline variables with a complicated index domain expanded in high-dimensional variables. This could lead to an unnecessary re-evaluation of the index domain while generating the high-dimensional constraint.
- An expression like `FormatString("5.0n", 1.5)` resulted in a rounding warning.

Issues ...

- A statement like `Write anIndexedSet to file filtering i in someSubSet;` would ignore the filtering altogether.
- Empty index sets could lead the function `DisAggregate` with a locus set into an infinite amount of memory allocated.

AIMMS 3.13 PR - SU4, release date 2014-07-18

In this fourth Software Update of the AIMMS 3.13 Production Release, the following issues have been fixed or improved.

*3.13 PR - SU4
2014-07-18*

- **Gurobi 5.6** has been upgraded to version 5.6.3.
- Expressions of the form `i IN someIndexedSet(someElemPar(i))` could return erroneous results.
- `CaseCompareIdentifier` did not handle the comparison of special values correctly.
- AIMMS could issue a superfluous warning on rounding ambiguity when the units of the arguments involve parametric units.
- For **CPLEX 12.5.1** (and higher), setting `aggressive_node_probing` was missing for the option `MIP_node_presolve`.
- Consider `A(i,j)`, `i` in `S`, `is` in `subS` subset of `S`, and the statement `EMPTY A(is,js)`. AIMMS could forget to update the definition of `subS` before executing `EMPTY A(is,js)`.
- Deleting at runtime, identifiers of type indexed set could lead to severe internal errors.
- The upper bound for the option `warning_round_tolerance` is lowered from 0.5 to 0.1 in order to avoid considering the value 0.5 to be the integer 1 during the writing of this option value to the option file.
- After `maximal_number_of_execution_errors_reported` errors reported, AIMMS did not get into an `On Error` block. Now this option only applies to the global error handler.
- Scrolling the **Gantt Chart** calendar could end up in an endless loop.

Improvement

Issues ...

AIMMS 3.13 PR - SU3, release date 2014-01-29

In this AIMMS 3.13 Software Update, the following issues have been fixed or improved.

*3.13 PR - SU3
2014-01-29*

- The option `auxiliary_root_threads` has been added for **CPLEX 12.5** and higher.
- **CPLEX 12.6** has been added. **CPLEX 12.6** can find global solutions for non-convex quadratic programs (QP and MIQP).
- There is a new version of **GUROBI**, i.e. **GUROBI 5.6**.

Improvements

- **CPOptimizer 12.6** has been added.
- Using the Microsoft Text ODBC driver for reading .csv files, raised an error with newer versions of this driver. *Issues ...*
- User authentication properties (username and password) have been added to the GIS background of the **Network Object** for Yahoo! Maps, Bing Maps and OpenStreetMap. *... GUI*
- Using the up or down buttons on the **Indices** tab of the **Identifier Selection** wizard of a **Table** object caused a crash in the 64-bit version of AIMMS.
- 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.
- It was possible to open/close empty branches in a **Pivot Table** object, resulting in a misalignment between the header and the grid area.
- Several math program suffices were not updated after a solve. *... Solving*
- The procedure `GMP::Tuning::TuneMultipleMPS` can now also be used to tune SAV files.
- Incorrect column or row numbers were sometimes passed by the following GMP routines:
`GMP::Column::GetName`,
`GMP::Row::GetName` and
`GMP::Solution::SetColumnValue`.
- The AIMMS presolver could generate an incorrect constraint definition if it removed variables from a constraint containing the `/ $` operator.
- Identifier types no longer publicly supported, but present in upgraded models, will now be added to the predeclared set `AllIdentifierTypes` during compilation. *... Compiler and Execution*
- The scalar iterative `Min` operator, with no condition and iterating over an empty set, would return 0 instead of `inf`.
- Emptying the definition attribute of a variable would not remove that variable from the set `AllConstraints`.
- Set definitions were evaluated during a `CleanDependents` statement, instead of before it, resulting in inconsistent data.
- Consider a condition such as $EP(i) \lt e$; AIMMS will translate such a condition into `REL_EP(i,e)` 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.
- Calling the function `ExitAIMMS` from within the startup procedure did only result in closing the project, and not in exiting AIMMS entirely.
- Deleting a unit parameter and subsequently creating a unit parameter with the same name may lead to superfluous error messages.

AIMMS 3.13 PR - SU2, release date 2013-10-16

In this second AIMMS 3.13 Software Update, the following issues have been fixed or improved.

*3.13 PR - SU2
2013-10-16*

- There is a new version of **CBC**, i.e. **CBC 2.8.0**. This new version fixes several issues including an issue where **CBC 2.7.5** returned 'infeasible' for a feasible MIP model.
- There is a new version of **CPLEX**, i.e. **CPLEX 12.5.1**.

Improvements

- AIMMS could become unresponsive while writing a case, when all subsets of the set Integers have the property NoSave (there should be at least one subset of Integers in the model). If only the subset of Integers with the smallest integer has a NoSave property, loading cases could result in erroneous data.
- A resolve in the **Math Program Inspector** would result in a severe internal error after the solver was deleted.
- 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.
- In some situations, the database mapping wizard did not allow you to select element parameters.

Issues ...

- Docked pages which are open during the startup of AIMMS could interfere with the startup sequence, and sometimes even lead to a crash.
- An early call to PageClose in the startup procedure could lead to a crash.
- The **Gantt Chart** object's navigation keys did not work when the chart was scrolled down.

... GUI

- The **Take Column Header Area Into Account** option (when 'Resetting the Column Width' for a **Pivot Table** object), now takes all rows in the column tree into account when determining the maximum size. Previously, it only considered the leaf cells in the column tree, but in some situations the text displayed at a parent level could be wider than the text displayed in a leaf cell.
- When determining the maximum column width in a **Pivot Table** object, any unit string present on an intermediate level in the column tree is now also taken into account.
- Using an indexed color parameter to specify the text color of an explicit index in the **Pivot Table** object, did not work.
- In a **Pivot Table** object on a data page for a parameter with a binary range, the values were not represented via check boxes, which was the case in earlier AIMMS versions.

... Pivot Table

- Consider the declaration:

```
SET:
  identifier : S
  index      : i
  order by   : user
  initial data: data { b, a } ;
```

and the assignment

```
S := sort( i, i );
```

In the above statement, the ordering of S would be used erroneously by the Sort operator, instead of the appropriate lexicographic order.

- Interrupting an AIMMS session executing a statement stmt within

```
Block
  stmt
OnError Err Do
  HndErrStmt
EndBlock
```

would position the error message on the HndErrStmt instead of on the actual (long running) stmt.

- Consider the following declarations:

```
SET:
  identifier : S
  index      : i ;
SET:
  identifier : T
  parameter : t0
  index      : j ;
ELEMENT PARAMETER:
  identifier : eP
  index domain : i
  range      : T ;
ELEMENT VARIABLE:
  identifier : eV
  range      : S
  property   : emptyElementAllowed ;
CONSTRAINT:
  identifier : selectedTshouldBe
  definition : eP(eV) = t0 ;
```

With the above declarations, AIMMS would base its element numbering for the set S on a possible presence of the empty element, but did not do so for the set T. It should have done so, however, because an empty element for eV results in eP(eV) also being the empty element.

- The Prod operator over an empty condition could erroneously return 0 instead of 1.
- Recomputing a calendar in which the new end date contains a component that is out of range, say, a month number of 32, might result in a severe internal error.

AIMMS 3.13 PR - SU1, release date 2013-07-12

In this first AIMMS 3.13 Software Update, the following issues have been fixed or improved.

*3.13 PR - SU1
2013-07-12*

- A new function has been introduced to obtain all spreadsheet names in a workbook: `Spreadsheet::GetAllSheets`.
- The concurrent optimizers of **CPLEX 12.5** and **Gurobi 5.5** are now supported by AIMMS. The concurrent optimizer can apply multiple algorithms to a single LP problem. The concurrent optimizer can be enabled by setting the **CPLEX** option `LP_method` or the **Gurobi** option `method`.
- There is a new version of **GUROBI**, i.e. **GUROBI 5.5**. **GUROBI 5.5** includes a new automatic option tuning tool. **GUROBI 5.5** introduces the concurrent MIP solver which performs several independent MIP solves in parallel, reporting the best results obtained by any of them. **GUROBI 5.5** can use one or more Gurobi Compute Servers for its computations. A Gurobi Compute Server license is required. This functionality is only available for the 'link-only' Gurobi license.
- There is a new version of **CP Optimizer**, i.e. **CP Optimizer 12.5**.
- The option `MIP_kappa` has been added for **CPLEX 12.5**.

Improvements

- The **XML Schema Mapping** tool stopped working because of the upgrade to MSXML 6 in earlier versions of AIMMS.
- Conversion of pages from unicode to ascii (or vice versa) was not implemented correctly. Some characters could stop the page read sequence, leaving the page in an invalid state.
- 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 lead to a corruption of 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.
- AIMMS could become unresponsive while writing a case, when all subsets of the set `Integers` have the property `NoSave` (there should be at least one subset of `Integers` in the model). If only the subset of `Integers` with the smallest integer has a `NoSave` property, loading cases could result in erroneous data.

Issues ...

- Sometimes the GIS background image of a network object appeared as background of other areas of the page. *... GUI*
- A **Curve** object with the property **Remove Defaults**, did not render correctly if the indices in the **Y identifier** and **X identifier** were not in the same order.
- Property dialogs that display properties in a tree format, like the ones used in the **ActiveX** objects or the **Pivot Table** object, would not always correctly display multiplicity properties for which the only entry was a double-click to create new entry.

- The **Pivot Table** object will not display a unit anymore when the cell is empty (e.g. because the cell value is at its default). *... Pivot Table*
- When running under AIMMS PRO, tooltips in a **Pivot Table** object could cause a crash.
- In a **Pivot Table** object, switching to another outer index for cases in which the element description (shown in the dropdown) was identical for distinct elements, would not update the grid area correctly.
- The multi-element filter in a **Pivot Table** object did not allow you to select the first identifier in your model.

- Producing a solution listing could, in rare circumstances, result in a severe internal error. *... 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.
- Solving an LP and regenerating and solving it two times more could lead to a severe internal error, if a constraint exists with the following properties:
 - there are multiple terms referencing the same variable and
 - some term would generate non-zero coefficients, but these coefficients cancel out resulting in no row generated for that constraint
- The postsolve step used by **CBC** and **Gurobi** could return incorrect variable values.
- Procedure `GMP::Coefficient::Set` could result in a severe internal error if the row was not generated before.
- **GMP-AOA** could crash if **MOSEK** was used as NLP solver.

- The `FileCopy` function did not check whether the destination file was already opened by some other process on Linux. *... Compiler and Execution*
- Executing a function with a three or higher dimensional result multiple times, could result in memory corruption and severe internal errors later on.
- Quantity based scaling, introduced as part of Constraint Programming in AIMMS 3.13, could cause a severe internal error when a parameter has a unit, say PSI. If this unit is commensurate with another unit, say PA, and PA is defined over three other units, say kg, m, and s, and a **Convention** is

- used to convert kg, m, or s, this error would occur.
- 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 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.
 - $P(n-i+m)$, where n, m are integer valued expressions and i is an index over a subset of the integers, would be handled as if parsed as $P(n-(i+m))$ instead of $P((n-i)+m)$.
 - 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.
 - Specifying a unit for an optional argument of a procedure or function and subsequently calling that procedure or function without the optional argument, would lead to a superfluous warning.
 - The intrinsic function `OptionGetDefaultString` is now implemented.
 - When using a construct like $Q(c, i, j, k) \mid c = \text{TheCase}$ (where c is an index in `AllCases` and `TheCase` is an element parameter in `AllCases`), with a definition of $c.P(i, j)$, AIMMS might use the incorrect case for retrieving the value of the first element of $P(i, j)$.
 - Iterating over the elements in a set stored in a case as in the following statement:


```
for i in epSelectedCase.someSet do
  ! do something with i
endfor ;
```

 whereby the elements in `epSelectedCase.someSet` are not in the root set of `someSet`, might lead to superfluous error messages.
 - Consider the expression: `mean(i, otherExpr)`. In case `otherExpr` uses an iterative operator this could lead to a superfluous error message.
 - AIMMS could produce erroneous results for expressions like:


```
sum( (i,j,k) | k in anIndexedSet(i,j-1) )
```

 Here, k is the last running index and a lag/lead is used in the referencing of the indexed set `anIndexedSet`.
 - 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.
- Editing values in the **Identifier View** window could result in a crash.
 - A comment-only **Body call** attribute of an external function could lead to a severe internal error.
 - In the **CaseLoad** or **CaseSave** dialog box, when using data management

Model Editor

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.

- Data entry for indexed sets, whereby a domain set or the range of that indexed set is a compound set, is now supported.
- Removing the **Schedule domain** attribute from an **Activity** could lead to a severe internal error when restarting the project.

AIMMS 3.13 PR, release date 2013-02-13

In this AIMMS 3.13 Production Release, the following issues have been fixed or improved.

*Build 3.13 PR
2013-02-13*

- There is a new write mode for writing AIMMS data to a database table: the write in insert mode. This mode uses only SQL INSERT statements under the hood for writing (as opposed to a combination of SQL INSERT and SQL UPDATE statements in the other write modes). In case this creates duplicate keys, AIMMS will issue an error. In practice, there are situations where you will know upfront that you will only introduce new rows in a table. Using the new insert mode for this, should show a significant performance improvement if your table already contains (lots of) data.
- **BARON 11** has been upgraded to version 11.8.0. The default of option `Maximal_variable_bound` has been changed to `inf`. Value `automatic` has been added for option `NLP_solver` and has become the default.
- There is a new version of **CPLEX: CPLEX 12.5**.
- There is a new version of **Gurobi: Gurobi 5.1**. **Gurobi 5.1** is not supported on 32-bit Linux machines.
- The performance of the procedure `GMP::Instance::CalculateSubGradient` with **CPLEX** has been improved.
- AIMMS is now able to read and write case files which exceed the former 2Gb size limit. This holds for both 32-bit and 64-bit versions of AIMMS.
- The OpenStreetMap GIS source has been extended with an optional URL property. Furthermore, the `GeoFindCoordinates` function has been extended with an optional `Url` argument.
- Setting the priority attribute or priority suffix of continuous variables, except for complementary variables, is now allowed. The **BARON** solver deploys these priorities.
- AIMMS now updates the solver configuration after a license upgrade. Newly licensed solvers will automatically be added.

Improvements

- The profiling information regarding the evaluation of the index domain condition of variables and constraints could be presented as a duplicate of the generation time of variables and constraints, when the index domain condition is not separately computed. *Issues ...*
 - Some command line options in the <projectName>.cmdargs file were not applied correctly.
 - Running a 32-bit and a 64-bit version of AIMMS next to each other on the same locally installed license, resulted in incorrect error messages. You should be able to have two AIMMS versions open, but they cannot do a solve at the same time.
 - **File - Open Project** now again has the shortcut CTRL+O.
 - Some attribute wizards (for which the attribute value is specified by an identifier reference) did not correctly show the currently selected identifier upon opening of the identifier selection dialog. For example, the **Priority** attribute of a variable.
-
- When using a non-default value for the **X-Axis tick spacing** property of the **2D Chart** ActiveX object, the chart was shifted one position to the right. *... GUI*
 - The **DateTimePicker** object behaved strangely upon changes in its **First Date Allowed** and **Last Date Allowed** properties.
 - When switching from a page to another newly opened page, the new page was very shortly drawn with a white background, which could be described as a short 'flash'.
 - If a dialog page is derived from a template page, then when opening the dialog, the toolbar of the dialog page could become visible. For dialog pages a toolbar or menubar is not applicable and thus should not be shown either.
 - Resizable dialog boxes or dialog pages did not show the small AIMMS icon in the top left corner.
-
- A **Pivot Table** object with some 'Implicit Identifiers' specified, could get messed up after a data change, because of properties of explicit identifiers being mapped onto the wrong identifiers. *... Pivot Table*
 - All subheaders in the **Pivot Table** now have a sort button.
 - **Pivot Table** properties that were defined by an identifier that was declared over a subset of `AllIdentifiers`, did not work for identifiers that were declared inside a library.
-
- AIMMS did not check for lazy constraints when solving a CP model, which could result in a crash instead of an error message. *... Solving*
 - The implementation of the procedure `GMP::Coefficient::Set` had become inefficient.
 - Using the procedure `GMP::Column::SetAsObjective`, could, under rare circumstances, result in a crash.

- AIMMS didn't recognize the ambiguity in DENSE MERGE mode for reading/writing. Now it flags this as an error.
- A read from file statement filtering on an index over a defined set, could issue superfluous error messages of elements outside the set.
- Increasing the dimension of an element parameter could confuse the AIMMS compiler to generate superfluous error messages at the place where this element parameter is now correctly used.
- No error message was raised when the ElementCast or StringToElement intrinsic functions added an element to a defined set.
- After deleting an element from a set and a call to CleanDependents, a **Pivot Table** that was already open, could crash.
- Saving an identifier containing only values in the range [-1e-13, 1e-13] in a case, could be read back as 1's.
- Computing scaling factors for unit scaling could involve taking the reciprocal of the original scaling multiple times, resulting in small but numeric significant differences. This, in turn, could result in a failed check whether a number is integral, and subsequently, a superfluous error message.
- AIMMS could issue a superfluous error message when, in a variable $X(i)$ with range $[0, p(i)]$, an indexed parameter $P(j)$ is referenced using an index i whereby i has the same root set as j , but i and j have different ranges.
- When the second or third argument of TimeSlotToMoment were not in the used calendar, AIMMS did not always issue an error message.
- Data Management 'Using disk files and folders' now updates the status line during lengthy Case I/O operations.

... Compiler and Execution

AIMMS 3.13 FR1, release date 2012-10-10

In this AIMMS 3.13 Feature Release, the following issues have been fixed or improved.

Build 3.13 FR1
2012-10-10

- The ActiveX **2D Chart** and **3D Chart** objects are now also available in the 64-bit versions of AIMMS.
- The AIMMS Excel Add-In now also runs under the 64-bit version of Excel.
- The AOA algorithm for convex MINLP problems (which is implemented as an open box algorithm in the system module GMPOuterApproximation using GMP functions) has been improved. The new implementation uses the lazy constraint callback of **CPLEX** 12.3 (and higher) and **GUROBI** 5.0 and has several benefits: (1) a simplified and more natural implementation, (2) improved performance, (3) it can also be used by **GUROBI**, and (4) it can also be used for problems with general integer variables (as opposed to the previous version, that could only be used for problems with binary variables). Also, the algorithm now starts with a preprocessing

Improvements

- step (using the nonlinear presolver) by default.
- **CPLEX** version 12.4 has been upgraded to **12.4.0.1**.
 - MINLP probing as part of the nonlinear presolver has been improved. It can linearize more nonlinear constraints. Probing can now also apply coefficient reduction which tightens the NLP relaxation and can decrease the solution time. The setting 'on' of the option `MINLP_probing` has been renamed to 'limited' and two new option settings that activate coefficient reduction have been added. See the AIMMS help for more information.
 - This release introduces the function `GMP::Solution::ConstraintListing` that outputs a description of a GMP and its contents. For further information, see the documentation of this function in the Function Reference.
 - There is now a possibility to override the location of the Code Snippets folder for the syntax editor, by providing a configurable option in the Syntax Editor Settings 'Misc' tab.
 - This release introduces the function `GMP::Instance::CreateFeasibility` that creates a feasibility problem for an infeasible GMP. For further information, see the documentation of this function in the Function Reference.
 - The **User/Edit** button, used for switching between user mode and edit mode of a page, has been replaced by 2 separate buttons.
-
- When starting AIMMS with the `--ide-style-95` command line option, the **Edit** menu in the **Math Program Inspector** did not show the 'Find' and 'Repeat Find' menu entries. *Issues ...*
 - Reading and writing cases over a network could be rather slow. The files are now cached on the local disk.
 - Added support for the use of AIMMS in combination with Adobe Acrobat X (i.e. Adobe Acrobat 10 (or higher)).
 - If for some reason a database read statement would result in an unexpected read state, a crash could occur afterwards.
 - Profiler data got lost when executing multiple procedures as an action of a **Button** object.
 - It is no longer allowed to set a breakpoint on an Assertion.
 - The option `visible_project_window`, together with the **Project Window** itself, has been removed.
 - The **New Project** dialog box no longer by default selects a folder in which another AIMMS project exists. In addition, if you try to create a new project within a folder that already contains an existing .prj file, a dialog pops up for confirmation.
 - Warnings issued by selected intrinsic functions were issued to the **Messages** window, but not to the **Errors/Warnings** window when the return value of that intrinsic procedure was not checked and the option `intrinsic_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 is used in the unit attribute of another parameter, variable or constraint, 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.
-
- 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. *... GUI*
 - The 'edit size' of resizable pages can now be changed via the properties dialog. Alternatively, you can use the 'size' property of a page to define the (initial) size of a dialog page or a page that is used as an inner page of an Indexed Page object.
 - For GIS feature layers (e.g. GML or ESRI), setting the point style and size had no effect.
 - In the **Calendar** object, the property 'Month Background' has been removed. It is now the same as the property 'Background'.
 - Using the `ExitAimms` function could sometimes result in AIMMS not being properly closed, or even in a crash.
 - The **DocumentViewer** object has a new 'Allow Navigation' property. If it is set to 'No', the end user cannot browse to any links that are contained in the web page represented by the given URL property.
 - The **DocumentViewer** object has a new 'Enable Browser Context Menu' property. If this option is set to 'Yes', the right-mouse context menu shows a default menu from the underlying Internet Explorer browser control.
 - 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.
-
- A **Pivot Table** object with auxiliary indices specified by an element parameter would crash after a data change (e.g. a case load). *... Pivot Table*
 - 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).
 - In the **Pivot Table** object, the first visible row and column were not always restored correctly (after a data change).
 - The **Pivot Table** object showed value bars for cells that did not contain a value.
 - The **Pivot Table** object did not show value bars for numerical suffices of identifiers that did not have a numerical level suffix.
 - 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.

- 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).
 - The property 'Location in Initial View' for the suffix index of a **Pivot Table** object was not defined as an option with keywords.
 - State files got bigger and bigger, even if the information in the state of a **Pivot Table** itself was shrinking.
 - The **Pivot Table** object did not always correctly restore the position of artificial indices from the state file.
-
- 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*
 - Special numbers (like UNDF, ZERO and NA) could be parsed as keywords, which caused expressions that referenced those special numbers to be incorrect, resulting in a red or wrong syntax highlighting.
 - Tooltip messages on AIMMS keywords have been enabled.
 - An editor window could crash when trying to collapse a statement-block without a terminator.
 - The syntax coloring of formal argument names, using positional arguments, didn't function correctly.
-
- An 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.
 - Replacing the uncertainty attribute with a region specification in an uncertain parameter, could previously have resulted in using both the restrictions from the uncertainty attribute and the region specifications.
 - AIMMS will now preprocess the uncertain mathematical program before constructing the robust counterpart. This is controlled by the new option `preprocess_uncertain_model`.
 - Solving a robust optimization problem containing adjustable variables with an empty dependency, could lead to incorrect results.
 - The `.violation` suffix of constraints and variables was not always filled if multiple threads were used by the solver.
 - Using the `GMP::Column::SetAsObjective` function, followed by a solve with **GUROBI** 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, a crash could be the result.

- AIMMS did not check the data type (numeric, element, or string) of references, that are used in the attributes **Range**, **Stage**, **Owner**, **Datasource** and **Server**. *... Stochastic Programming*

- 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. *... Units of Measurement*
- 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.

- The AIMMS compiler allowed `Halt`, `Return`, `Skip` and `Break` statements between `OnError` and `EndBlock`. Now, an error message is raised in such cases. In addition, a `Halt` statement is now ignored while executing statements in an `OnError` clause (and an accompanying warning is issued). *... Compiler and Execution*
- The function `ActiveCard` produced wrong results on element parameters when the value of an element parameter referred to an element that was no longer in its range.
- A non-empty comment-only definition of an ordered set could cause the ordering of the set to be computed incorrectly.
- Some additional checks are added before executing a procedure scheduled using `ScheduleAt`.
- The `%s` modifier in `FormatString`'s first argument now also accepts expressions of data type element, numeric and unit.
- 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.
- Reading a section, using the intrinsic Model Edit Function `me : ImportNode`, where the section contained procedures without a body, could lead to a severe internal error.