
AIMMS Function Reference - Deprecated

This file contains only one chapter of the book. For a free download of the complete book in pdf format, please visit www.aimms.com

Part VI

Deprecated

Deprecated

The current implementation of AIMMS supports the following deprecated features, but it may cease to do so in a future implementation. The current implementation does so to support converted GAMS and AIMMS 2 applications.

Deprecated keywords

The keywords for which direct replacements are available are documented in Table 59.1.

Deprecated	Modern equivalent
clean	CleanDependents
CumulativeDistribution	DistributionCumulative
eps	zero
evaluate	update
FailureCount	FailCount
InverseCumulativeDistribution	DistributionInverseCumulative
maximise	maximize
maximising	maximize
maximizing	maximize
minimise	minimize
minimising	minimize
minimizing	minimize
net_inflow	netinflow
net_outflow	netoutflow
puttl	puthd

Table 59.1: AIMMS deprecated keywords and their modern equivalents

The deprecated keyword `abort`

The keyword `abort` is a GAMS keyword that can be followed by a condition and a list of identifiers to be displayed. The execution run is interrupted after

executing this statement. Suggested rewrite: use a `display` statement followed by a `halt` statement or a `raise error` statement. See also

- `display` See Section 29.3,
- `halt` See Section 8.3.6, and
- `raise error` See Section 8.4.2.

The deprecated keywords `yes` and `no`

The keywords `yes` and `no` are GAMS keywords that can be used in assignments to sets in order to add or remove elements. Suggested rewrite: use the AIMMS set syntax. For instance, replace

```
s1(i) $ cond1(i) := yes ;
s2(i) $ cond2(i) := no ;
```

by the following code:

```
s1 += { i | cond1(i) } ;
s2 -= { i | cond2(i) } ;
```

The deprecated keyword `system`

The GAMS keyword `system` is followed by a suffix. The AIMMS language supports the following equivalent code for selected `system` suffixes as documented in Table 59.2.

Deprecated	Modern equivalent
<code>.date</code>	<code>CurrentToString("%Am AllAbbrMonths %d, %C%y")</code>
<code>.time</code>	<code>CurrentToString("%H:%M:%S")</code>
<code>.version</code>	<code>AimmsRevisionString(string parameter, 4);</code>
<code>.page</code>	<code>currentOutputFile.PageNumber</code>

Table 59.2: The keyword `system` and selected suffixes with their modern counterparts

The `system` suffixes `.ifile`, `.ofile`, `.rdate`, `.rfile`, `.runtime`, `.sfile`, and `.title` are pointless within the AIMMS environment.

Deprecated intrinsic procedures and functions

The mapping of the matrix manipulation procedures to GMP procedures and functions is documented in Table 21.6 of the Language Reference.

The following intrinsic functions are deprecated, but can be replaced by an equivalent call to an existing intrinsic procedure or function:

- FindRString(SearchString, Key, CaseSensitive, WordOnly, IgnoreWhite) can be replaced by a call to FindString(SearchString, Key, -1, CaseSensitive, WordOnly, IgnoreWhite) where -1 indicates that searching should be done right to left, see also [FindString](#).
- One may replace SQLDirect with DirectSQL
- One may replace StringToLabel with StringToElement

The deprecated iterative operators are documented in Table 59.3.

Deprecated	Modern equivalent
smax	max
smin	min
arg	nth

Table 59.3: AIMMS deprecated iterative operators and their modern equivalents

Deprecated suffixes

Most deprecated suffices can be directly translated into their modern equivalents, as documented in Table 59.4.

The following suffixes deserve some more consideration:

- .ap The append mode of a file, 0: replace contents when opening the file, 1: append to file. This functionality is now covered by the mode attribute of that file, see Section 29.1.
- .m The marginal value of a variable or constraint. For a constraint the suffix .m should be replaced by the suffix .ShadowPrice. For a variable the suffix .m should be replaced by the suffix .ReducedCost.
- .modelstat This suffix of a mathematical program is numeric, it should be replaced by the element valued suffix .ProgramStatus. Note that Element(AllSolutionStates, mp.solvestat+1) = mp.ProgramStatus. See also Table 15.4 and [AllSolutionStates](#).
- .solvestat or .solverstat These suffixes of a mathematical program are numeric, they should be replaced by the element valued suffix .SolverStatus. Note that Element(AllSolutionStates, mp.solvestat+15) = mp.SolverStatus. See also Table 15.4 and [AllSolutionStates](#).
- .dim This should be replaced by a call to [IdentifierDimension](#).
- .txt This should be replaced by a call to [IdentifierText](#).
- .type This should be replaced by a call to [IdentifierType](#).

Deprecated	Modern equivalent
Variables	
.l	.level
.lo	.lower
.up	.upper
.freeze	.nonvar
.prior	.priority
Files	
.bm	.BottomMargin
.cc	.BodyCurrentColumn
.cr	.BodyCurrentRow
.ftcc	.FooterCurrentColumn
.ftcr	.FooterCurrentRow
.ftll	.HeaderSize
.hdcc	.HeaderCurrentColumn
.hdcr	.HeaderCurrentRow
.hdll	.FooterSize
.lm	.LeftMargin
.lp .pn	.PageNumber
.pc	.PageMode
.ps	.PageSize
.pw	.PageWidth
.tm	.TopMargin
Mathematical programs	
.bestest .object	.LinearObjective
.iterusd	.iterations
.nodusd	.nodes
.number	.SolverCalls
.numequ	.NumberOfConstraints
.numinfes	.NumberOfInfeasibilities
.numintvar	.NumberOfIntegerVariables
.numnlequ	.NumberOfNonlinearConstraints
.numnlins	.NumberOfNonlinearInstructions
.numnlhz .numnlz	.NumberOfNonlinearNonzeros
.numnlvar	.NumberOfNonlinearVariables
.numnz	.NumberOfNonzeros
.numSOS1	.NumberOfSOS1Constraints
.numSOS2	.NumberOfSOS2Constraints
.numvar	.NumberOfVariables
.objval	.Objective
.resgen	.GenTime
.resusd	.SolutionTime
.suminfes	.SumOfInfeasibilities

Table 59.4: AIMMS deprecated suffices and their modern equivalents