# The numodel-coach package

Export numodel models to CMA Coach 7

Paul Zuurbier
mail@paulzuurbier.nl

v0.14.0 (2026/11/28)

**Abstract**

numodel-coach writes a model declared with numodel as a modelling activity for CMA Coach 7 (a `.cma7` file), in Coach's text mode. Pupils open the file in Coach and get exactly the model of the text, without retyping it from the PDF. The model rules and initial values are numodel's Coachtaal rendering (`numodel.plaintext`); the file can also be embedded in the PDF.

## 1 Usage

Load the package after numodel (LuaLaTeX only):

```
\usepackage[syntax=NL]{numodel}
\usepackage{numodel-coach}
```

After the model is declared, `\coachmodel` writes it:

```
\newmodelprefix{ball}
\mvar{T}{t}{0}{\s}{2}{systeem}
\mvar{Dt}{dt}{0.05}{\s}{2}{systeem}
\mvar{V}{v}{0}{\m\per\s}{3}{voorraad}
\mvar{Y}{y}{100}{\m}{3}{voorraad}
\mvar{G}{g}{-9.81}{\m\per\s\squared}{3}{constante}
\mrule{V}{\ballV + \ballG * \ballDt}
\mrule{Y}{\ballY + \ballV * \ballDt}
\mrule{T}{\ballT + \ballDt}
\mstop{\ballY <= 2}
\textmodel
\coachmodel{vrije-val}
```

This writes `vrije-val.cma7` with the model rules

```
v := v + g*dt
y := y + v*dt
t := t + dt
Als y <= 2 Dan Stop EindAls
```

and the initial values

```
t := 0      's
dt := 0,05  's
v := 0      'm/s
y := 100    'm
g := -9,81  'm/s^2
```

All variables are listed in Coach's variable list (for tables and graphs) with their units.

`\coachmodel` `\coachmodel[⟨options⟩]{⟨name⟩}` writes the model with the current prefix to ⟨*name*⟩`.cma7` (no extension is added when ⟨*name*⟩ has one). Options: `prefix=`⟨*prefix*⟩ for another model, and `dir`, `attach`, `switch`, `mode` and `blanks` as below.

`\coachsetup` `\coachsetup{`⟨*options*⟩`}` sets defaults for all following `\coachmodel` calls:

**`dir=`**⟨***directory***⟩ Directory for the files, relative to the working directory; created when missing. Default: the working directory.

**attach=true/false** Also embed the file in the PDF (with embedfile). Default `false`. Whether readers can open embedded files depends on the PDF viewer.

**blanks=open/filled** (`\coachmodel` only) Write the pupil's version, in which every rule, stop condition and start value declared with numodel's `blank` key becomes a comment `' F_z := ...` to complete in Coach, or the answer version in full. Default: numodel's `blanks` setting (`open`).

**mode=text/graphic** Write the model as a text model (default) or as a graphical model, in which pupils can switch to the text view; see Section 2.

**graphic-style=edu/tight** The look of a graphical model, as numodel's `diagram-style edu` and `tight`: `edu` (default) draws every variable as an object of its own, `tight` lets a flow that is a single variable carry its name and formula; see Section 2.

**switch=true/false/auto** Whether pupils may switch between Coach's text and graphical view. Default `auto`: on for a graphical model, off for a text model. Coach derives the text view from the graphical model, which a text model leaves empty, so switching would lose a text model.

## 2 Graphical model

With `\coachmodel[mode=graphic]{...}` the file holds Coach's graphical model: the stock-and-flow diagram, with a formula in every object. Coach opens it in the graphical view, and pupils can switch to the text view, which Coach derives from the diagram (in its own order of the rules). The model is translated as follows:

- The time rule `t := t + dt` becomes Coach's time; the start value of $t$ and $\Delta t$ go to the model settings, together with the stop condition.
- A stock rule $X := X + \ldots$ in which every other term has a factor $\Delta t$ becomes a stock (*voorraadgrootheid*) with one flow per term; the factor $\Delta t$ goes, as Coach multiplies by it itself. A term $-Q\,\Delta t$ in one stock and $+Q\,\Delta t$ in another is one flow between them.
- With `graphic-style=edu` (the default) the flows are called `Stroom_1`, …; a flow $+a\,\Delta t$ has the formula $a$, and $a$ stays an object of its own with an arrow to the flow, as `\graphicmodel` draws it in its `edu` style. With `graphic-style=tight` a flow that is a single variable used nowhere else – an auxiliary variable, also a conditional one, or a constant – *is* that variable: it gets its name (shown in the diagram), unit and formula or value, as `\graphicmodel` draws it as the valve in its default `tight` style. Other flows are still `Stroom_1`, …
- Other rules become auxiliary variables; a conditional rule `c ? a : b` uses Coach's condition. Variables without a rule are constants. A start value of an auxiliary variable becomes its initial value.
- Every variable a formula uses gets an arrow (relation) to it.
- The positions follow `\graphicmodel`'s layout in the matching style; Coach lets pupils move everything. A flow that closes a loop of stocks runs back over the row above, with right angles as Coach draws its flows: from the inner top corner of one stock, along its valve, to the inner top corner of the other.

A model that cannot be drawn this way – a stock rule with a term without $\Delta t$ (`n := n + 1`), a conditional rule using $\Delta t$, two rules for one variable – is written as a text model, with a warning naming the reason. In a graphical model Coach computes the auxiliary variables from the stocks of the same step; a text model in which an auxiliary rule comes *after* the stock it uses computes it from the new value, so the two can differ slightly.

## 3 Instruction text

`modelinstruction` (*env.*) The instruction text belongs to numodel: the environment `modelinstruction[`⟨*prefix*=⟨*prefix*⟩⟩`]` defines the instruction text of a model (by default the current one). It typesets nothing. The former names `coachinstruction` and `\coachinstructiontext` remain as aliases.

`\modelinstructiontext[`⟨*prefix*=⟨*prefix*⟩⟩`]` typesets that text in the PDF, wherever you want it

and as often as you like (or not at all), and `\coachmodel` writes it to Coach's instruction window. The PDF and Coach thus share one source:

```
\begin{modelinstruction}
Een bal valt vanuit rust van \qty{100}{\m}. Neem
$g = \qty{9.81}{\m\per\s\squared}$.

Opdrachten:
\begin{enumerate}
  \item Bereken de \textbf{eindsnelheid} $v_{\text{eind}}$.
\end{enumerate}
\end{modelinstruction}

\modelinstructiontext   % the text in the PDF
\coachmodel{vrije-val}  % the same text in Coach
```

A model has one instruction text: a second `modelinstruction` for the same model replaces the first, with a warning. The text is typeset where `\modelinstructiontext` stands, so anything in it that depends on its position (a counter, say) takes its value there.

Coach shows HTML in its instruction window; the body is converted from the following LaTeX:

- paragraphs (blank lines), `\\`, `\newline`;
- `\textbf`, `\emph`, `\textit`, `\underline`, `\textsuperscript`, `\textsubscript`; `\text` and similar wrappers keep their text;
- `itemize` and `enumerate` with `\item`;
- inline math (`$...$`, `\(...\)`): letters in italics, _ and ^ as sub- and superscript, Greek letters, `\frac`, `\sqrt`, `\cdot`, relations such as `\leq`;
- `\qty`, `\num`, `\unit` (and `\SI`, `\si`): numbers with a decimal comma, units in Coach's notation;
- `~`, `--`, `---`, quotes, and a handful of symbols (`\ldots`, `\times`, `\degree`, …).

Any other command gives a warning; its braced argument is kept as text. Display math, tables and pictures are not converted.

# 4 What is exported

- *Model rules* in the order of `\textmodel`: `\mrule` as an assignment, conditional rules as `Als … Dan … Anders … EindAls` (on several lines for `\mrule*`), `\mstop` as `Als … Dan Stop EindAls`. A `\mruletext` row becomes a comment, with a warning. The display keys `alias`, `aliasleft` and `aliasright` do not apply: Coach gets the calculation.
- *Initial values* of all variables with a start value, with the unit as a comment.
- *Variable list*: every variable, with its unit, so it can be put in Coach's table and graphs. The axis range Coach uses for a new graph or meter is the range of the variable's graph in the document: the axis limits `\diagrammodel` drew (the union, when the variable appears in several diagrams). A variable that is not drawn gets the range a diagram of its computed series would have (the same rounding to whole ticks); only a model that was never computed keeps Coach's default 0 to 10. Call `\coachmodel` after `\computemodel` and the diagrams. Coach shows two decimals.
- *Names*: the display name without formatting (`F_{\text{res}}` becomes `F_res`, `\Delta t` becomes `Δt`, `\lambda` `λ`). Names Coach does not accept bare (with a space or a slash), and its reserved words, are put in brackets: `[t_1/2]`, `[Max]`.
- *Units*: siunitx units in Coach's notation: `\m\per\s\squared` becomes `m/s^2`, `\mole\per\litre\per\second` becomes `mol/(L*s)`. A unit of your own (`\DeclareSIUnit{\animals}{animals}`) is exported as its symbol: `\animals\per\day` becomes `animals/d`.

- *Angle unit*: Coach's "hoekeenheid" follows numodel's `angles` setting (`\numodelsetup{angles=degrees}` in the preamble; default radians), and so do the trigonometric functions: with degrees `sind(\alpha)` is exported as `Sin(alfa)`, `sin(x)` as `Sin(180/Pi*x)`, `asind(x)` as `Arcsin(x)` and `asin(x)` as `Pi/180*Arcsin(x)` (Coach's inverse functions answer in the angle unit too). `\textmodel` shows the same.
- *Iterations*: the number of iterations in Coach's model settings is numodel's `maxiter` (`\numodelsetup{maxiter=...}`, default 20000). The `Stop` rule ends the run; `maxiter` is only the upper limit.

Expressions are translated with care for the differences between l3fp and Coachtaal: Coach gives unary minus and `^` the same priority, so `-x^2` is exported as `-(x^2)`; comparisons combined with `EN`/`OF` get the parentheses Coachtaal requires. Whatever cannot be exported exactly (a two-argument `atan`, a nested conditional, an unknown unit macro) is reported as a package warning naming the model and the file.

# 5 The file format

CMA does not document the `.cma7` format. numodel-coach starts from an empty modelling activity saved by Coach 7.0 in text mode (stored as Lua data in `numodel-coach-template.lua`) and fills in the model rules, initial values, variable list and iteration count. Files written this way were tested with Coach 7 on Chromebooks. A later Coach version may change the format; please report problems.