

Variable Training Schemas

Tempo ramps, progressions, controlled randomness and heart-rate references

Serega Photoskiff

2026

Contents

Why variability matters	2
A minimal syntax introduction	2
Heart-rate references and repeatable work	3
Random blocks and seed	3
Core variable-load patterns	4
1. Smooth entry	4
2. Smooth exit	4
3. Tempo wave	4
4. Step ladder	4
5. Pyramid	4
6. Recovery compression	5
7. Period shift	5
8. Finishing kick	5
9. Turbulence	5
10. Budget AMRAP	5
11. Adaptive tempo	6
Sport-specific examples	6
Kettlebell snatch: smooth entry	6
Kettlebell snatch: continuous wave	6
Kettlebell snatch: negative split	6
High swing: acceleration in every minute	6
Kettlebell swing: ten-minute wave	7
Pull-ups: automatic ladder to eight	7
Running: cadence wave	7
Running: six smooth accelerations	7
Cycling: cadence pyramid	7
Cycling: eight spin-up sprints	7
Rowing ergometer: stroke-rate build	8
Boxing: round plus finishing combination	8
Sailing: gust simulation	8
Working with heart rate	8
Repeat one Schema	8
Compare model and measurement	8
Compare tactics	8
Suggested library tags	9
Practical design recommendations	9

Do not begin at maximum	9
Vary one parameter first	9
Use seed for tests	10
Preserve a heart-rate reference	10
Design record-session tactics	10

Summary	10
----------------	-----------

Why variability matters

An ordinary interval timer mainly answers: **when should I work and when should I recover?**

The Schema language can describe a richer object: **how load changes over time**. Load may be discrete - hard work, easy work, rest - but it may also be smooth, wave-shaped, progressive, randomly turbulent or adaptive.

Real training and record attempts are rarely perfect slices of identical intervals. They include entry into pace, acceleration, fatigue management, rhythm changes, finishing kicks, recovery, game-like bursts, technical phases and an individual physiological response.

Schema records those scenarios compactly, reproducibly and in a form suitable for analysis.

It can prescribe:

- duration, tempo, target repetitions and intensity;
- smooth tempo ramps within one work interval;
- stepwise progressions;
- local time or repetition budgets;
- random sessions with controlled turbulence;
- comparison of measured heart rate with a model or an earlier performance.

For a recreational athlete, this provides a controlled way to train by the **shape of load** rather than by guesswork. For an advanced athlete, it provides a language of tactics: entry pace, cruise pace, negative split, finishing kick, drift control and repeat-pass comparison.

A minimal syntax introduction

```
30/30@8*5#
```

- 30/30 - 30 seconds work, 30 seconds recovery;
- @8 - work intensity 8;
- *5 - repeat five times;
- # - finish after the Schema.

Tempo:

```
60/0@t20*10#
```

Ten minutes of continuous work at a tempo of 20 movements per minute.

Target repetitions:

```
30/20@r10*6#
```

Six blocks: perform ten repetitions within 30 seconds, then recover for 20 seconds.

Smooth tempo ramp:

```
120/0@t[14->24]#
```

Two minutes in which tempo rises smoothly from 14 to 24 per minute.

Progression:

```
30/30@7->[to:30/10@9,delta:30/-5@+0.5]#
```

Work remains 30 seconds, recovery decreases step by step and intensity rises until the target is reached.

Heart-rate references and repeatable work

The timer can compare measured heart rate with a reference curve. The reference may be:

1. a theoretical curve produced by the individual model; or
2. a measured curve from an earlier performance of the same or a similar Schema.

Repeated passes can answer questions that the final score alone cannot:

- Is heart rate lower at the same load?
- Has drift decreased?
- Is recovery faster after hard segments?
- Does heart rate enter severe zones later?
- Is the finishing ramp physiologically cheaper?
- Can tempo increase without increasing heart-rate cost?

Example:

The same Schema, tempo and kettlebell weight are repeated one month later. Mean heart rate is five beats lower, the peak is seven beats lower, and the finishing ramp no longer produces an abrupt move into Z5. This is measurable progress, not merely “it felt easier”.

Random blocks and seed

`random(...)` is useful when controlled unpredictability is part of the task: variable-pace running, games, boxing, sailing, agility and reaction work.

```
random(
  30/0@6t165
  20/0@8t190
  10/0@9t215
  turb:8 dur:600 min:10
)#title=broken pace 10 minutes
```

- the block contains several permitted atoms;
- `dur:600` gives the random block a ten-minute duration;
- `turb:8` sets high turbulence;
- `min:10` sets a minimum hold time;
- the runtime materialises one concrete load pattern.

Without seed, each materialisation may be new. Add a seed when the pattern must be repeated:

```
random(
  30/0@6t165
  20/0@8t190
  10/0@9t215
  turb:8 dur:600 min:10 seed:42
)#title=broken pace 10 minutes, seed 42
```

A seeded random Schema can act as a control course for repeated testing, athlete-to-athlete comparison or re-testing after a training block.

Core variable-load patterns

1. Smooth entry

```
[easy=12 work=24]
(120/0@t[easy->work] 0/20)*4#title=smooth entry
```

Tempo rises from 12 to 24 over two minutes, followed by a 20-second pause; repeat four times. Useful for kettlebell lifting, swings, running, rowing, cycling and record-session preparation.

General form:

```
[easy=A work=B] (WORK/0@t[easy->work] REST/0)*N#
```

2. Smooth exit

```
[work=24 easy=12]
(90/0@t[work->easy] 0/30)*4#title=smooth exit
```

Tempo decreases smoothly before the pause. This suits cooldown, technical recovery and control of breathing and movement under fatigue.

3. Tempo wave

```
[easy=12 hard=24]
(90/0@t[easy->hard] 90/0@t[hard->easy])*4#title=tempo wave
```

Each three-minute wave accelerates and decelerates without stopping. This trains pace control and makes the lag between load and heart-rate response visible.

4. Step ladder

```
60/0@7t18
->[to:60/0@9t26,delta:60/0@+0.5t+2]#
title=stepwise tempo build
```

Every step keeps 60 seconds of work, raises intensity by 0.5 and raises tempo by 2 until tempo 26 and intensity 9 are reached.

5. Pyramid

```
60/0@6t18
->[to:60/0@9t26,delta:60/0@+1t+2]
->[to:60/0@6t18,delta:60/0@-1t-2]#
title=tempo pyramid
```

Tempo and intensity rise, then return. The peak is in the middle rather than at the finish.

6. Recovery compression

```
30/60@8
->[to:30/15@9,delta:30/-15@+0.25]#
title=recovery compression
```

Work remains 30 seconds; recovery falls 60 -> 45 -> 30 -> 15; intensity rises slightly. Useful for strength endurance and dense competition formats.

7. Period shift

```
30/0@4r7
->[to:15/0@9r7,delta:-3/0@+1r+0]
->[to:30/0@4r7,delta:+3/0@-1r+0]#
title=period shift at constant volume
```

The target remains seven repetitions while the work interval contracts from 30 to 15 seconds, then expands again. The same volume is produced at changing density.

8. Finishing kick

```
[base=18 race=22 kick=26]
120/0@t[base->race]
360/0@t22
120/0@t[race->kick]#
title=negative split, 10 minutes
```

Two minutes of entry, six minutes at cruise pace and two minutes of acceleration. Useful for record attempts, rowing, cycling, running and kettlebell sport.

9. Turbulence

```
random(
  30/0@6t165
  20/0@8t190
  10/0@9t215
  turb:8 dur:600 min:10
)#title=turbulence, 10 minutes
```

A ten-minute variable load assembled from three permitted segment types. Add seed when an identical virtual course must be repeated.

10. Budget AMRAP

```
dur:600(~/0@r~*)#title=Recorder, 10 minutes
```

The block has a ten-minute budget; actual work segments, repetitions and sets are recorded during execution.

A repetition-budget version:

```
reps:200(~/0@r~*)#title=Recorder, 200 repetitions
```

11. Adaptive tempo

```
300/0@t180^#title=adaptive cadence, 5 minutes
```

The atom has a theoretical tempo plan, but the runtime may alter the realised plan in response to the athlete or interface. Intended and realised forms remain available for comparison.

Sport-specific examples

Kettlebell snatch: smooth entry

```
[low=14 work=22]
(120/0@t[low->work] 0/20)*4#
title=snatch smooth entry, 8 minutes
```

Four two-minute work blocks. Each rises from tempo 14 to 22, followed by 20 seconds of recovery. The aim is to enter working pace without disrupting breathing or technique.

Kettlebell snatch: continuous wave

```
[low=14 high=24]
(90/0@t[low->high] 90/0@t[high->low])*4#
title=snatch wave, 12 minutes
```

Four continuous waves. Use the descending half to observe how quickly heart rate stabilises as tempo falls.

Kettlebell snatch: negative split

```
[base=18 race=22 kick=26]
120/0@t[base->race]
360/0@t22
120/0@t[race->kick]#
title=snatch negative split, 10 minutes
```

Watch whether the first rise is too expensive, whether the middle develops uncontrolled drift, and what the final kick costs.

High swing: acceleration in every minute

```
58/2@t[18->26]*10`#
title=accelerating high swing, 10 minutes
```

Ten near-minute work blocks, each accelerating from 18 to 26. The two-second boundary permits a brief transition, and the backtick removes the final recovery.

Kettlebell swing: ten-minute wave

```
[lo=24 hi=36]
(60/0@t[lo->hi] 60/0@t[hi->lo])*5#
title=swing tempo wave, 10 minutes
```

Continuous strength-aerobic work with breathing and technique control at changing tempo.

Pull-ups: automatic ladder to eight

```
[dw=2 dr=2 N=8]
(dw/dr@r1
->[to:auto/auto@rN,delta:+dw/+dr@r+1]
->[to:auto/auto@r1,delta:-dw/-dr@r-1])*1`#
title=pull-up ladder to 8
```

Repetitions rise from one to eight and return to one while work and recovery scale with the target.

Running: cadence wave

```
[easy=165 fast=190]
(180/0@6t[easy->fast] 180/0@6t[fast->easy])*3#
title=running cadence wave, 18 minutes
```

Three six-minute waves. Observe heart-rate lag, recovery on the descending side and drift across repeated waves.

Running: six smooth accelerations

```
(20/70@9t[180->215])*6#
title=six smooth running accelerations
```

Twenty seconds of progressive acceleration followed by 70 seconds of easy recovery. This supports stride mechanics without an abrupt sprint start.

Cycling: cadence pyramid

```
60/30@6t80
->[to:60/30@9t110,delta:60/30@+0.5t+5]
->[to:60/30@6t80,delta:60/30@-0.5t-5]#
title=cycling cadence pyramid
```

Cadence rises from 80 to 110 and returns, with 60-second work periods and 30-second recoveries.

Cycling: eight spin-up sprints

```
(15/75@9t[95->125])*8#
title=eight cycling spin-up sprints
```

Fifteen seconds accelerating from cadence 95 to 125, then 75 seconds of recovery. Compare heart-rate recovery after successive sprints.

Rowing ergometer: stroke-rate build

```
[lo=22 hi=32]
(240/120@7t[lo->hi])*4#
title=rowing 4 x 4-minute stroke-rate build
```

Each four-minute interval rises from 22 to 32 strokes per minute, followed by two minutes of recovery.

Boxing: round plus finishing combination

```
(150/30@6t120 30/60@9t[180->260])*4#
title=boxing round plus finishing combination
```

Two and a half minutes of controlled work, a 30-second accelerating finish and 60 seconds of recovery, repeated for four rounds.

Sailing: gust simulation

```
random(
  20!1/10@7
  12!2/18@9
  30!3/20@6
  turb:8 dur:900 min:10 seed:31
)#title=hiking gust simulation
1=hike hold 2=gust burst 3=active recovery
```

Fifteen minutes of repeatable variable work combining a sustained hold, a short gust and active recovery.

Working with heart rate

Repeat one Schema

The simplest strong design is:

1. perform a Schema;
2. save measured heart rate;
3. use it as a reference;
4. repeat the same Schema after one week or one month;
5. compare both outcome and physiological cost.

Compare model and measurement

When no earlier pass exists, use the theoretical model as an initial reference. It may reveal an overly aggressive entry, insufficient recovery, an expensive finishing kick or a mismatch in model parameters.

Compare tactics

For the same ten-minute snatch task, compare:

Constant pace:

```
60/0@t22*10#title=snatch constant tempo 22
```

Negative split:

```
[base=18 race=22 kick=26]
120/0@t[base->race] 360/0@t22 120/0@t[race->kick]#
title=snatch negative split
```

Wave plus finish:

```
[low=18 high=26]
(90/0@t[low->high] 90/0@t[high->low])*3
60/0@t24#title=snatch wave plus finish
```

Compare total volume, mean and peak heart rate, time in zones, drift, technical stability and the cost of the finish.

Suggested library tags

ramp-in	smooth entry
ramp-out	smooth exit
wave	rise and fall
ladder	stepwise ladder
pyramid	ascent and descent
negative-split	controlled start, strong finish
density	recovery compression
period-shift	changing work-period length
finish-kick	finishing acceleration
random-turb	variable turbulence
seeded-random	repeatable random pattern
budget-amrap	dur:/reps: with deferred values
adaptive-tempo	live tempo with ^
hr-reference	comparison with a heart-rate reference
hr-replay	comparison with a previous pass

Sport tags may include kettlebell, running, cycling, rowing, boxing, swimming, calisthenics, climbing, sailing, agility and sking.

Practical design recommendations

Do not begin at maximum

A controlled entry is often preferable:

```
120/0@t[18->22]
```

rather than an abrupt:

```
120/0@t22
```

This is particularly relevant for older athletes, heavy implements, running and technically demanding movements.

Vary one parameter first

A good first experiment changes tempo, recovery, duration **or** repetitions. If everything changes at once, the session is difficult to interpret.

Use seed for tests

If a random Schema is a test, give it a seed. If it is a game-like session whose purpose is novelty, omit the seed.

Preserve a heart-rate reference

Repeat a well-defined Schema every one to four weeks. Compare both performance and physiological cost.

Design record-session tactics

A record attempt should answer:

- how to enter pace;
- where to hold cruise pace;
- when to begin the finish;
- what the finish costs;
- whether drift begins too early.

Summary

Sport is not only work and recovery. It is load taking shape over time: entry, acceleration, wave, hold, disruption, recovery, finishing effort, random gust and physiological response.

Schema makes that shape compact and executable. Coupled with heart rate, it becomes an analytical tool:

- compare measurement with a theoretical model;
- use an earlier performance as a reference;
- measure progress at the same load;
- test alternative tactics;
- create repeatable random tests through seed.

Schema describes not only a workout, but a reproducible experiment on the shape of load and the athlete's response.