Release notes

0.7.0-X

Breaking Changes

- predicates:
  - range check or unification to a *variable* of a *predicate*'s *truth value* requieres an `=` character
- terms:
  - *regexp* is no longer an *atom*
  - *escaper* behavior have changed

Changes

- elementals:
  - FZZCFFBNetwork:
    - new `datafile` property to save network to a binary file
    - support for *list* and *data* terms
  - MRKCBFSolver:
    - new property `reply.on` and `cascade`, `cascade.tmo`
  - FZZCCollector
    - speed-up
    - modified behavior of property `tmo` to be the time-out from the last received replies
    - added property `ttl` to specify the time-to-live value for the query
  - MRKCCSVStore
    - property `arity` to specify the arity of the statements (if the number of columns is greater than the arity, the extra will be grouped into a list as the last term)
  - EV3CSENLEGOGyros
    - property `inverted`
  - EV3CBEVSonar
    - support to *peek* at the current reading
- primitives:
  - `lst.sort` now accept as 3rd term a list of indexes to be used for sorting lists (+1 index will be used when the lists' terms are equal)
  - revisited the way the `sim` primitive compute the similarity between two numbers
- samples:
  - updated `irl2asm.fizz`
- terms:
  - a *frame*'s label can be any atom (and not just a *symbol*)
- console:
  - `/spy` output contains the timestamp
  - use `verbose` property to silence output
  - ignore *variables* with name starting with an upper case

Additions

- samples:
  - iris2, iris3
  - nlu
  - movies
  - ml
  - db
  - fuzzy
  - fun, eval, exec, sexp, lstrnd
  - funx, funx2, funx3
  - tasc (based on Hector Levesque's book "Thinking as Computation" (ISBN: 978-0-262-01699-5))
- console:
  - /trace (see section 4.4 on page 35)
- elementals:
  - MRKCStopper (see section 6 on page 103)
  - FZZCFUNRunner (see section 6 on page 103)
  - EV3CSRVMapping (see section 7 on page 124)
- predicates:
  - `?` prefix for *predicate* (see section 2.3 on page 4)
- terms:
  - *data* (see section 3.3 on page 12)
  - *quirk* (see section 3.11 on page 18)
- volatiles ((see section 3.10 on page 18):
  - `sym.8`
  - `sym.6`
  - `now.ms`
- constraints:
  - `fun.label`
- constants:
  - `pi`
- prototypes
  - support for alternate fuzzy and-or evaluation (see section 2.4 on page 7)
- primitives:
  - `rnd.sint` (see section 5.12 on page 81)
  - `qrk.head` (see section 5.11 on page 79)
  - `qrk.tail` (see section 5.11 on page 80)
  - `qrk.make` (see section 5.11 on page 79)
  - `is.quirk` (see section 5.17 on page 95)
  - `lst.any` (see section 5.7 on page 64)
  - `lst.all` (see section 5.7 on page 63)
  - `qat.euler` (see section 5.18 on page 97)

- `qat.apply` (see section 5.18 on page 97)
- `vec.lenght` (see section 5.18 on page 99)
- `vec.dist` (see section 5.18 on page 99)
- `vec.angle` (see section 5.18 on page 98)
- `vec.angle.signed` (see section 5.18 on page 99)
- `vec.norm` (see section 5.18 on page 100)
- `mat.make` (see section 5.18 on page 97)
- `mat.apply` (see section 5.18 on page 96)
- `min` (see section 5.1 on page 39)
- `max` (see section 5.1 on page 39)
- `cache` (see section 5.2 on page 42)
- `rng.not` (see section 5.13 on page 83)
- `rng.real` (see section 5.13 on page 85)
- `frm.swap` (see section 5.5 on page 62)
- `pull` (see section 5.2 on page 50)
- `push` (see section 5.2 on page 50)
- `drop` (see section 5.2 on page 47)
- `lst.split` (see section 5.7 on page 71)
- `lst.knit` (see section 5.7 on page 67)
- `mao.sin` (see section 5.9 on page 77)
- `mao.cos` (see section 5.9 on page 74)
- `mao.atan2` (see section 5.9 on page 73)
- `mao.d2r` (see section 5.9 on page 74)
- `spawn` (see section 5.2 on page 52)
- `cease` (see section 5.2 on page 43)
- `shoot` (see section 5.2 on page 52)
- `is.data` (see section 5.17 on page 93)
- `prune` (see section 5.2 on page 50)
- `daa.make` (see section 5.4 on page 57)
- `daa.length` (see section 5.4 on page 57)
- `daa.member` (see section 5.4 on page 58)
- `daa.format` (see section 5.4 on page 57)
- `daa.item` (see section 5.4 on page 57)
- `fzz.labels` (see section 5.10 on page 77)
- `is.primitive` (see section 5.17 on page 95)
- `exec` (see section 5.2 on page 47)
- `var.capture` (see section 5.10 on page 79)
- `var.release` (see section 5.10 on page 79)
- `var.collect` (see section 5.10 on page 79)
- `cpy` (see section 5.2 on page 45)
- `uny` (see section 5.2 on page 54)
- `nab` (see section 5.2 on page 48)
- `lst.combi` (see section 5.7 on page 64)
- `hush.if` (see section 5.2 on page 48)

- `hush.if.not` (see section 5.2 on page 48)
- `cut.if` (see section 5.2 on page 45)
- `cut.if.not` (see section 5.2 on page 45)
- `lst.min` (see section 5.7 on page 69)
- `lst.max` (see section 5.7 on page 68)
- `lst.avg` (see section 5.7 on page 64)
- `vec.add` (see section 5.18 on page 98)
- `vec.sub` (see section 5.18 on page 100)
- `vec.mul` (see section 5.18 on page 100)
- `vec.div` (see section 5.18 on page 99)
- `rng.norm` (see section 5.13 on page 83)
- `daa.find` (see section 5.4 on page 56)
- `daa.min` (see section 5.4 on page 58)
- `daa.max` (see section 5.4 on page 58)
- `daa.avg` (see section 5.4 on page 56)
- `qat.add` (see section 5.18 on page 97)
- `qat.sub` (see section 5.18 on page 98)
- `qat.length` (see section 5.18 on page 98)
- `lst.it` (see section 5.7 on page 66)
- `any` (see section 5.2 on page 41)

Bug Fixes

- issue with *variables* in the `define` *primitive*
- issue with `fun.terms` not unifying to a `split-list` (as 2nd term)
- issue with property `clone` not finding the *elemental* to clone from
- issue with `rnd.sint` and `rnd.uint` crashing with `"Floating point exception (core dumped)"` when the range was given using the same value
- issue with `then` *primitive* confusing minutes and seconds
- issue with `mao.abs` *primitive* returning `0` when the first term was a negative floating point value
- issue with `frm.make` *primitive* failing when an empty *list* was used as one of the *term*
- issue in `FZZCCLUGateway` leading to long delay in further transmission after a large one