Floating-Point Language-Independent Type for YAML™ Version 1.1

Working Draft 2005-01-18

Oren Ben-Kiki <oren@ben-kiki.org>
Clark Evans <cce+yaml@clarkevans.com>
Brian Ingerson <ingy@ttul.org>

Copyright © 2001-2005 Oren Ben-Kiki, Clark Evans, Brian Ingerson
This document may be freely copied provided it is not modified.

Status

This specification is a draft reflecting consensus reached by members of the yaml-core mailing list [http://lists.sourceforge.net/lists/listinfo/yaml-core]. Any questions regarding this draft should be raised on this list.

URI: tag:yaml.org,2002:float

Shorthand: !!float

Kind: Scalar.

Canonical:

\n0
\n|-?\.[0-9]*[1-9])?e[-+]0[1-9][0-9]+ (scientific)
\n|\-?.inf (infinity)
\n|\.nan (not a number)

Regexp:

[-+]?(0?9.[0-9_]*?[0-9_]*([eE][-+]0[0-9]+)* (base 10)
\n|[-+]?(0?9_[0-9_]*([0-5]?[0-9])?\.[0-9_]* (base 60)
\n|[-+]0\.(inf|Inf|INF) # (infinity)
\n|\.(nan|NaN|NAN) # (not a number)

Definition: Floating-point approximation to real numbers.

Floating-point numbers are approximations to real numbers, including three special values (positive and negative infinity and “not a number”). Using “:” allows expressing the integer part in base 60, which is convenient for time and angle values (the fractional part is always in base 10). Any “_” characters in the number are ignored, allowing a readable representation of large values.

This should be loaded to some native float data type. The processor may choose from a range of such native data types according to the size and accuracy of the floating-point value. Note that not all floating-point values can be represented exactly when stored in any native float type, and hence a float value may change by “a small amount” when round-tripped through a native type. The valid range and accuracy depends on the implementation, though 32 bit IEEE floats should be safe. Since YAML does not specify a particular accuracy, using floating-point mapping keys requires great care and is not recommended.
**Example 1. **float Examples

- canonical: 6.8523015e+5
- exponential: 685.230_15e+03
- fixed: 685_230.15
- sexagesimal: 190:20:30.15
- negative infinity: -inf
- not a number: .NaN