

Nix simplified grammar

expressions

```
e ::= x | c
    | e.a | e.a or e
    | p:e | e e
    | { e = e; ...; e = e; }
    | with e; e
    | if e then e else e
    | let x = e; ...; x = e; in e
    | Cons(e, e) | nil
    | o

o ::= attrNames(e) | listToAttrs(e)
    | addAttr(e, e, e) | removeAttr(e, e)
    | deepSeq(e, e) | seq(e, e)
    | functionArgs(e)

a ::= b. ... .b

b ::= #{e} | l

c ::= s | i | ...
    | typeOf
    | ...

p ::= q | q@x | x

q ::= { f, ..., f } | { f, ..., f, ... }
    | Cons(x, x) | nil
    | c

f ::= x | x ? e

values

v ::= c
    | p:e
    | { v = e; ...; v = e; }
    | Cons(e, e) | nil
```